

**PENERAPAN *CUSTOMER RELATIONSHIP MANAGEMENT* (CRM)
PADA SISTEM INFORMASI PENJUALAN DI CV DUTA SQUARE
BANDAR LAMPUNG**

(SKRIPSI)

Oleh

FIKRI AHMAD MAULANA



**JURUSAN ILMU KOMPUTER
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG**

2019

ABSTRACT

APPLICATION OF CUSTOMER RELATIONSHIP MANAGEMENT (CRM) ON SALES INFORMATION SYSTEM IN CV DUTA SQUARE BANDAR LAMPUNG

By

FIKRI AHMAD MAULANA

To compete and develop business better, businesses must continue to innovate in the implementation of business processes. One of the innovations that can be done by business actors is implementing technopreneurship in their business. Application of technology and utilization of data that occurs in the business conducted will have many advantages for business people, therefore Customer Relationship Management (CRM) is made in the micro business in the retail field CV. Duta Square, located in Bandar Lampung, Lampung. It is expected that this system can assist the owner in handling the problem of data collection of goods, data collection of sales transactions, and can find out the stock of goods in the warehouse. SCRUM will be used in the development phase and will implement CRM which focuses on point-of-sales, data collection, and member transaction storage. Implementation of the company's needs will be discussed with business owners to determine the effectiveness of the system that has been made.

Keywords: Customer Relationship Management, point-of-sales, technopreneurship.

ABSTRAK

PENERAPAN *CUSTOMER RELATIONSHIP MANAGEMENT* (CRM) PADA SISTEM INFORMASI PENJUALAN DI CV DUTA SQUARE BANDAR LAMPUNG

Oleh

FIKRI AHMAD MAULANA

Untuk bersaing dan mengembangkan bisnis lebih baik lagi, pelaku usaha harus terus berinovasi dalam pelaksanaan proses bisnisnya. Salah satu inovasi yang dapat dilakukan oleh pelaku usaha adalah mengimplementasikan *technopreneurship* dalam bisnis yang dilakukannya. Penerapan teknologi dan pemanfaatan data yang terjadi pada bisnis yang dilakukan akan memiliki banyak keuntungan untuk pelaku usaha, oleh karena itu dibuat *Customer Relationship Management* (CRM) pada usaha mikro bidang ritel CV. Duta Square yang berlokasi di Bandar Lampung, Lampung. Diharapkan sistem ini dapat membantu pemilik dalam menangani masalah pendataan barang, pendataan transaksi penjualan, dan dapat mengetahui stok barang yang ada di gudang. SCRUM akan digunakan dalam tahap pengembangan dan akan mengimplementasikan CRM yang berfokus pada *point-of-sales*, pendataan barang, dan penyimpanan transaksi member. Implementasi kebutuhan perusahaan akan didiskusikan kepada pemilik usaha untuk menentukan keefektifan dari sistem yang telah dibuat.

Kata kunci: *Customer Relationship Management, point-of-sales, technopreneurship.*

**PENERAPAN *CUSTOMER RELATIONSHIP MANAGEMENT* (CRM)
PADA SISTEM INFORMASI PENJUALAN DI CV DUTA SQUARE
BANDAR LAMPUNG**

Oleh

Fikri Ahmad Maulana

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

: **PENERAPAN CUSTOMER RELATIONSHIP
MANAGEMENT (CRM) PADA SISTEM
INFORMASI PENJUALAN DI CV DUTA SQUARE
BANDAR LAMPUNG**

Nama Mahasiswa

: **Fikri Ahmad Maulana**

No. Pokok Mahasiswa : 1517051218

Jurusan

: Ilmu Komputer

Fakultas

: Matematika dan Ilmu Pengetahuan Alam



1. Komisi Pembimbing

Astria Hijriani, S.Kom., M.Kom.

NIP. 19810308 200812 2 002

2. Mengetahui

Ketua Jurusan Ilmu Komputer
FMIPA Universitas Lampung

Dr. Ir. Kurnia Muludi, M.S.Sc.

NIP. 19640616 198902 1 001

MENGESAHKAN

1. Tim Penguji

Ketua : Astria Hijriani, S.Kom., M.Kom.

Penguji I

Bukan Pembimbing : Drs. Rd. Irwan Adi Pribadi, M.Kom.

Penguji II

Bukan Pembimbing : Rico Andrian, S.Si., M.Kom.



Drs. Sarafman, M.Sc.

19640604 199003 1 002



Tanggal Lulus Ujian Skripsi : 20 November 2019

PERNYATAAN

Saya yang bertanda tangan di bawah ini, menyatakan bahwa skripsi saya yang berjudul "**Penerapan *Customer Relationship Management (CRM)* Pada Sistem Informasi Penjualan 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 di kemudian 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, 20 November 2019



Fikri Ahmad Maulana
NPM. 1517051218

RIWAYAT HIDUP



Penulis dilahirkan pada tanggal 14 April 1997 di Bandar Lampung, sebagai anak pertama dari empat bersaudara dengan Ayah bernama Adrisman dan Ibu Yumilia Harianti.

Penulis menyelesaikan pendidikan formal pertama kali di Taman Kanak-Kanak Kartini I pada tahun 2003.

Pendidikan dasar di SDN 2 Palapa dan selesai pada tahun 2009. Pendidikan menengah pertama di SMPN 9 Bandar Lampung yang diselesaikan pada tahun 2012, kemudian melanjutkan ke pendidikan menengah kejuruan di SMKN 4 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 Badan Khusus Himpunan Mahasiswa Jurusan Ilmu Komputer pada periode 2016/2017.

3. Menjadi Asisten Laboratorium dan Asisten Dosen Jurusan Ilmu Komputer pada periode 2016/2017 sampai dengan 2017/2018.
4. Pada bulan Januari 2018 penulis melaksanakan Kerja Praktik di Badan Pusat Statistik (BPS) Kota Bandar Lampung.
5. Pada bulan Juli 2018 penulis melaksanakan Kuliah Kerja Nyata (KKN) di Desa Mulyo Jadi, Kecamatan Gunung Terang, Kabupaten Tulang Bawang Barat.

PERSEMBAHAN

Segala puji dan syukur kupersembahkan kepada ALLAH Subhanallahu Wa Ta'ala yang maha pengasih lagi maha penyayang atas berkat, nikmat, anugerah, rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan skripsi ini dengan baik.

Serta Rasulullah Nabi Muhammad Shallallahu 'Alaihi Wa Sallam yang telah membawa umat dari zaman kebodohan menuju zaman ilmu pengetahuan.

Kedua Orang Tua Tercinta Ayahku Adrisman dan Umiku Yumilia Harianti

Orang tua yang telah membesarkan, merawat, memberikan ilmu agama dan dunia, pengalaman yang berharga serta memberikan dukungan materil maupun moril selama diriku masih kecil hingga sekarang. Terima kasih atas semua doa dan harapan yang telah dipanjatkan kepadaku selama ini.

Saudara Kandung dan Seluruh Keluarga besar

Seluruh keluarga yang telah mendukung dan memberikan ilmu serta pelajaran yang berharga untuk hidupku selama ini. Terima kasih telah menjadi inspirasi dan motivasi bagiku untuk menjadi orang yang lebih baik.

Teman dan Sahabat Tersayang

Teman dan sahabat yang selalu berbagi cerita dan kisah selama hidupku, canda tawa, suka duka pengalaman, dan kenangan. Terima kasih atas dukungan, saran, dan masukan yang membuatku menjadi orang yang lebih baik.

Serta

Keluarga Ilmu Komputer 2015 dan Almamater yang kubanggakan.

UNIVERSITAS LAMPUNG

MOTTO

“Maka sesungguhnya bersama kesulitan ada kemudahan, sesungguhnya bersama kesulitan ada kemudahan”

(Q.S Al-Insyirah: 5-6)

“Berlomba-lombalah dalam kebaikan”

(Q.S Al-Baqarah: 148)

“Sebaik-baik manusia adalah yang paling bermanfaat bagi manusia”

(HR. Ahmad, ath-Thabrani, ad-Daruqutni)

“Mencari ilmu itu hukumnya wajib bagi muslimin dan muslimat”

(HR. Ibnu Abdil Bari)

SANWACANA

Assalamualaikum Wr. Wb,

Segala puji dan syukur penulis haturkan kehadiran ALLAH S.W.T yang Maha Pengasih lagi Maha Penyayang karena berkat, nikmat, anugerah, dan rahmat-Nya sehingga penulis dapat menyelesaikan skripsi ini yang berjudul “Penerapan *Customer Relationship Management* (CRM) Pada Sistem Informasi Penjualan di CV Duta Square Bandar Lampung”.

Dengan keyakinan, ketabahan, dan kemauan yang keras serta bimbingan dan ridho dari ALLAH S.W.T, juga melalui dukungan dan bantuan dari berbagai pihak penulis dapat menyelesaikan karya ini. Penulis menyadari bahwa masih terdapat banyak kekurangan dalam pembuatan karya ini dikarenakan kurangnya pengetahuan yang penulis miliki. Melalui kesempatan ini penulis hendak mengucapkan terima kasih yang tidak terhingga kepada berbagai pihak mulai dari keluarga, dosen dan teman-teman yang telah membantu penulis memberikan dukungan moril maupun spiritual.

Dengan diiringi salam dan doa kepada ALLAH S.W.T serta ucapan terima kasih yang tidak terhingga penulis ucapkan kepada:

1. Kedua orangtuaku, Ayah dan Umi tercinta Bapak Adrisman dan Ibu Yumilia Harianti yang menjadi inspirasiku dan terus mengorbankan waktu, tenaga, dan kasih sayang yang tak ternilai. Terima kasih atas segala kasih sayang, doa, semangat, dan segala dukungan yang telah diberikan kepadaku selama ini. Aku akan terus berjuang untuk menjadi kebanggaan dan membahagiakan kalian. Semoga ALLAH S.W.T terus memberikan kesehatan dan limpahan rahmat kepada kedua orangtuaku.
2. Untuk ketiga adik-adikku, Putri Ayu Nabilah, Luthfiyah Adriyanti Safitri, dan Hafiz Ghani Muzaffar yang telah mendukungku, semoga aku bisa menjadi inspirasi dan contoh yang baik kepada kalian semua dan kita dapat membanggakan kedua orangtua yang kita cintai. Semoga ALLAH S.W.T terus memberikan kesehatan dan limpahan rahmat kepada kita semua.
3. Keluarga besarku tersayang yang selalu mendukung dari berbagai hal, menyemangati, dan terus mendidik diriku menjadi orang yang lebih baik dunia akhirat, serta membuatku mendapat pengalaman yang berharga.
4. Ibu Astria Hijriani S.Kom., M.Kom. selaku Pembimbing Utama, yang telah bersedia meluangkan waktu, pikiran, dan tenaga, serta memberikan semangat agar terus berusaha yang terbaik. Terima kasih atas bimbingan, arahan, dan masukan yang membuat skripsi ini menjadi lebih baik.
5. Bapak Drs. Rd. Irwan Adi Pribadi, M.Kom. selaku dosen Pembahas I yang telah memberikan kritik, saran, dan masukan yang sangat membantu penulis dalam memperbaiki dan membuat skripsi ini menjadi lebih baik.

6. Bapak Rico Andrian, S.Si., M.Kom. selaku dosen Pembahas II yang telah memberikan kritik, saran, dan masukan berharga yang sangat membantu penulis dalam memperbaiki dan membuat skripsi ini menjadi lebih baik.
7. Bapak Aristoteles, S.Si., M.Si. selaku Pembimbing Akademik yang telah meluangkan waktu, tenaga, dan memberikan motivasi serta membimbing penulis selama melaksanakan perkuliahan.
8. Bapak Drs. Suratman, M.Sc. selaku Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.
9. Bapak Dr. Ir. Kurnia Muludi, M.S.Sc. selaku Ketua Jurusan Ilmu Komputer FMIPA Universitas Lampung.
10. Bapak Didik Kurniawan, S.Si., M.T. selaku Sekretaris Jurusan Ilmu Komputer.
11. Bapak dan Ibu Dosen Jurusan Ilmu Komputer yang telah memberikan banyak ilmu, pengalaman, dan motivasi terhadap penulis selama menjadi mahasiswa. Semoga ilmu yang diberikan dapat bermanfaat dan menjadi amal Jariah yang bermanfaat untuk seluruh masyarakat serta menjadi bekal yang berharga untuk penulis kedepannya.
12. Ibu Ade Nora Maela yang telah membantu segala administrasi dan membantu penulis melaksanakan kegiatan skripsi.
13. Bang Zainudin, A.Md. dan keluarga Laboratorium Ilmu Komputer yang telah berbagi ilmu, pengalaman, dan canda tawa selama ini.
14. Mas Ardi Nofalian yang telah memberikan izin tempat untuk melaksanakan seluruh seminar dan tempat mengerjakan skripsi.

15. Elin Aprilliana sahabat terbaik semenjak SMP yang selalu mendukungku dalam keadaan senang maupun susah, yang telah menemani perjuangan selama ini hingga menyelesaikan skripsi dan seluruh prosesnya. Semoga kita dapat mencapai keinginan dan cita-cita serta membanggakan dan membahagiakan orang yang kita cintai.
16. *Team CV*. Duta Square, Elin Aprilliana, Jannati Asri safitri, dan Dina Munjiati yang telah menyelesaikan skripsi bersama dalam sebuah *team* dan melalui banyak proses sulit hingga menghasilkan karya yang bermanfaat dan semoga dapat kita banggakan.
17. Sahabat-sahabat Normal Elin, Shendy, Dara, Ghani, Dito, Jaka, Ibrahim, Pido, Rexy, Wawai, Steffy, Rozzak, Putri, Ardian, dan Fanny, yang telah menjadi sahabat yang berbagi kebahagiaan, canda, tawa, dan pengalaman mulai dari pertama menjadi mahasiswa hingga kini, semoga kita menjadi orang yang sukses dan berhasil ke depannya.
18. Teman-teman Ilmu Komputer dan teman seperjuangan skripsi yang telah berbagi cerita, ilmu, dan canda tawa mulai dari perkuliahan hingga akhir perkuliahan ini, semoga kita dapat menggapai cita-cita kita.

Penulis menyadari bahwa skripsi ini masih jauh dari kata sempurna, semoga skripsi ini membawa manfaat dan keberkahan bagi semua civitas Ilmu Komputer Universitas Lampung.

Bandar Lampung, 20 November 2019

Fikri Ahmad Maulana
NPM. 1517051218

DAFTAR ISI

	Halaman
DAFTAR ISI	xviii
DAFTAR TABEL	xix
DAFTAR GAMBAR	xxi
I. PENDAHULUAN	26
A. Latar Belakang	26
B. Rumusan Masalah	29
C. Batasan Masalah.....	29
D. Tujuan	30
E. Manfaat	31
II. TINJAUAN PUSTAKA	32
A. CV. Duta Square	32
B. POS (<i>Point of Sale</i>)	33
C. CRM (<i>Customer Relationship Management</i>)	33
D. PHP	34
E. Laravel	34
F. CSS	36
G. Bootstrap	36
H. <i>Database</i>	37
I. MySQL	37
J. phpMyAdmin	37
K. <i>Unified Modeling Language</i> (UML)	38
L. Metodologi Pengembangan Sistem	41

III. METODE PENELITIAN	44
A. Waktu dan Tempat	44
B. Alat Penelitian	44
C. Metodologi Penelitian	46
1. <i>User stories</i>	50
2. <i>Product backlog</i>	51
D. Metodologi Pengumpulan Data	53
1. Observasi	54
2. Wawancara	54
E. Analisis Sistem	54
1. Analisis Masalah	54
2. Analisis Kebutuhan Sistem	55
F. Perancangan Sistem	57
1. <i>Use case diagram</i>	57
2. <i>Activity diagram</i>	59
3. <i>Sequence diagram</i>	76
4. <i>Class diagram</i>	91
5. Desain antarmuka atau <i>interface</i>	93
G. Pengujian Sistem	101
IV. HASIL DAN PEMBAHASAN	115
A. Hasil	115
B. Pembahasan	117
C. Hasil Pengujian Sistem	158
V. SIMPULAN DAN SARAN	174
A. Simpulan	174
B. Saran	175
DAFTAR PUSTAKA	176

DAFTAR TABEL

	Halaman
Tabel 1. Simbol-simbol <i>Use Case Diagram</i>	38
Tabel 2. Simbol-simbol <i>Activity Diagram</i>	39
Tabel 3. Simbol-simbol <i>Sequence Diagram</i>	40
Tabel 4. Simbol-simbol <i>Class Diagram</i>	41
Tabel 5. Rencana Implementasi CRM	57
Tabel 6. <i>Test Scenario Black-Box</i> Super Admin	101
Tabel 7. <i>Test Scenario Black-Box</i> pada Sistem Untuk Kasir	103
Tabel 8. <i>Test Scenario Black-Box</i> pada Sistem Untuk Admin Penjualan	106
Tabel 9. <i>Test Scenario Black-Box</i> pada Sistem Untuk <i>Customer</i>	114
Tabel 10. Hasil Analisis Numerik.....	116
Tabel 11. Daftar <i>Model</i> Sistem Informasi Penjualan CRM	117
Tabel 12. Daftar <i>View</i> Sistem Informasi Penjualan CRM.....	119
Tabel 13. Daftar <i>Controller</i> Sistem Informasi Penjualan CRM	122
Tabel 14. Daftar Jadwal Pengerjaan Sistem	125
Tabel 15. <i>Sprint Planning</i>	126
Tabel 16. Progress <i>Daily Scrum Sprint</i> ke-1	131
Tabel 17. Progress <i>Daily Scrum Sprint</i> ke-2	133
Tabel 18. Progress <i>Daily Scrum Sprint</i> ke-3	139

Tabel 19. Progress <i>Daily Scrum Sprint</i> ke-4	144
Tabel 20. Progress <i>Daily Scrum Sprint</i> ke-5	148
Tabel 21. Progress <i>Daily Scrum Sprint</i> ke-6	153
Tabel 22. Hasil <i>Test Scenario Black-Box</i> Super Admin	158
Tabel 23. Hasil <i>Test Scenario Black-Box</i> Kasir	161
Tabel 24. Hasil <i>Test Scenario Black-Box</i> Customer	164
Tabel 25. Hasil <i>Test Scenario Black-Box</i> Admin Penjualan	164

DAFTAR GAMBAR

	Halaman
Gambar 1. Logo Laravel.....	35
Gambar 2. Tahapan Penggunaan Metode Scrum.	42
Gambar 3. Alur Metode Scrum.	47
Gambar 4. Hubungan Antar Sistem.	48
Gambar 5. <i>Product Backlog</i> Masing-masing Sistem.	52
Gambar 6. <i>Usecase Diagram</i> Sistem CRM.	58
Gambar 7. <i>Activity Diagram</i> Manambah User.	59
Gambar 8. <i>Activity Diagram</i> Mengedit Password User.....	60
Gambar 9. <i>Activity Diagram</i> Menghapus User.....	61
Gambar 10. <i>Activity Diagram</i> Melihat Aktivitas User.....	61
Gambar 11. <i>Activity Diagram</i> Menambah Data <i>Department</i>	62
Gambar 12. <i>Activity Diagram</i> Mengedit Data <i>Department</i>	63
Gambar 13. <i>Activity Diagram</i> Menghapus Data <i>Department</i>	63
Gambar 14. <i>Activity Diagram</i> Melihat Detail <i>Department</i>	64
Gambar 15. <i>Activity Diagram</i> Menambah Identitas Kasir.	65
Gambar 16. <i>Activity Diagram</i> Melihat Data Identitas Kasir.	65
Gambar 17. <i>Activity Diagram</i> Mengedit Data Identitas Kasir.....	66
Gambar 18. <i>Activity Diagram</i> Menghapus Data Kasir.....	67

Gambar 19. <i>Activity Diagram</i> Mulai Transaksi.	67
Gambar 20. <i>Activity Diagram</i> Akhiri Transaksi.	68
Gambar 21. <i>Activity Diagram</i> Melihat Detail Sesi Penjualan.	68
Gambar 22. <i>Activity Diagram</i> Melihat Detail Penjualan.	69
Gambar 23. <i>Activity Diagram</i> Menambah Member.	70
Gambar 24. <i>Activity Diagram</i> Melihat Detail Member.	70
Gambar 25. <i>Activity Diagram</i> Mengedit Data Member.	71
Gambar 26. <i>Activity Diagram</i> Menghapus Data Member.	72
Gambar 27. <i>Activity Diagram</i> Melihat Detail Master Barang.	72
Gambar 28. <i>Activity Diagram</i> Mencetak Barcode Barang.	73
Gambar 29. <i>Activity Diagram</i> Melihat Display Toko.	73
Gambar 30. <i>Activity Diagram</i> Melihat Detail Kategori Barang.	74
Gambar 31. <i>Activity Diagram</i> Kasir Mengelola Point of Sales.	75
Gambar 32. <i>Activity Diagram</i> Kasir Melihat Detail Supplier.	75
Gambar 33. <i>Activity Diagram</i> Mencari Informasi Barang.	76
Gambar 34. <i>Sequence Diagram</i> Menambah User.	77
Gambar 35. <i>Sequence Diagram</i> Mengedit Password User.	78
Gambar 36. <i>Sequence Diagram</i> Menghapus User.	78
Gambar 37. <i>Sequence Diagram</i> Aktivitas User.	79
Gambar 38. <i>Sequence Diagram</i> Menambah Data Department.	80
Gambar 39. <i>Sequence Diagram</i> Mengedit Data Department.	80
Gambar 40. <i>Sequence Diagram</i> Menghapus Data Department.	81
Gambar 41. <i>Sequence Diagram</i> Melihat Data Department.	81
Gambar 42. <i>Sequence Diagram</i> Melihat Detail Kasir.	82

Gambar 43. <i>Sequence Diagram</i> Mulai Transaksi Penjualan.	83
Gambar 44. <i>Sequence Diagram</i> Akhiri Transaksi Penjualan.	83
Gambar 45. <i>Sequence Diagram</i> Melihat Detail Sesi Penjualan.....	84
Gambar 46. <i>Sequence Diagram</i> Melihat Detail Data Penjualan.....	85
Gambar 47. <i>Sequence Diagram</i> Menambah Data <i>Member</i>	85
Gambar 48. <i>Sequence Diagram</i> Melihat Data <i>Member</i>	86
Gambar 49. <i>Sequence Diagram</i> Mengedit Data <i>Member</i>	87
Gambar 50. <i>Sequence Diagram</i> Manghapus <i>Member</i>	87
Gambar 51. <i>Sequence Diagram</i> Melihat Detail Barang.	88
Gambar 52. <i>Sequence Diagram</i> Mencetak <i>Barcode</i> Barang.	88
Gambar 53. <i>Sequence Diagram</i> Melihat <i>Display</i> Toko.	89
Gambar 54. <i>Sequence Diagram</i> Melihat Kategori Barang.	89
Gambar 55. <i>Sequence Diagram</i> Mengelola Menu <i>Point of Sales</i>	90
Gambar 56. <i>Sequence Diagram</i> Melihat Detail <i>Supplier</i>	90
Gambar 57. <i>Sequence Diagram</i> Mencari Informasi Barang.....	91
Gambar 58. <i>Class Diagram</i> <i>Customer Relationship Management</i>	92
Gambar 59. Rancangan <i>Interface</i> <i>Login</i>	93
Gambar 60. Rancangan <i>Interface</i> <i>Point of Sales</i>	94
Gambar 61. Rancangan <i>Interface</i> Detail Penjualan.	94
Gambar 62. Rancangan <i>Interface</i> Mencari Informasi Barang.	95
Gambar 63. Rancangan <i>Interface</i> Data <i>Supplier</i>	96
Gambar 64. Rancangan <i>Interface</i> Detail <i>Supplier</i>	96
Gambar 65. Rancangan <i>Interface</i> Master Barang.	97
Gambar 66. Rancangan <i>Interface</i> Detail Barang.	98

Gambar 67. Rancangan <i>Interface Display Toko</i>	98
Gambar 68. Rancangan <i>Interface Detail Display toko</i>	99
Gambar 69. Rancangan <i>Interface Menambah Data Member</i>	100
Gambar 70. Rancangan <i>Interface Data Member</i>	100
Gambar 71. Tampilan Halaman <i>Department</i>	136
Gambar 72. Tampilan Halaman Tambah <i>Department</i>	136
Gambar 73. Tampilan Halaman <i>Edit Department</i>	137
Gambar 74. Tampilan Halaman Hapus <i>Department</i>	137
Gambar 75. Tampilan Halaman Identitas Kasir.....	138
Gambar 76. Tampilan Halaman <i>Edit Identitas Kasir</i>	138
Gambar 77. Tampilan Halaman Hapus Identitas Kasir.	138
Gambar 78. Tampilan Halaman <i>Member</i>	142
Gambar 79. Tampilan Halaman Tambah <i>Member</i>	142
Gambar 80. Tampilan Halaman Detail Data <i>Member</i>	142
Gambar 81. Tampilan Halaman <i>Edit Data Member</i>	143
Gambar 82. Tampilan Halaman Hapus Data <i>Member</i>	143
Gambar 83. Tampilan Halaman Master Barang.	147
Gambar 84. Tampilan Halaman Kategori Barang.....	147
Gambar 85. Tampilan Halaman Penjualan.	150
Gambar 86. Tampilan Halaman Detail Penjualan.....	150
Gambar 87. Tampilan Halaman <i>Point of Sales</i>	151
Gambar 88. Tampilan Halaman Jumlah Barang <i>Point of Sales</i>	151
Gambar 89. Tampilan Halaman Jumlah Pembayaran <i>Point of Sales</i>	152
Gambar 90. Tampilan Halaman Notifikasi <i>Point of Sales</i>	152

Gambar 91. Tampilan Halaman Sesi Penjualan.....	155
Gambar 92. Tampilan Halaman <i>Supplier</i>	155
Gambar 93. Tampilan Halaman <i>Display</i> Toko.	156
Gambar 94. Tampilan Halaman Pencarian Barang Berdasarkan <i>Barcode</i>	156
Gambar 95. Tampilan Halaman Hasil Pencarian Barang.	157
Gambar 96. Tampilan Halaman Pencarian Berdasarkan <i>Merk</i>	157
Gambar 97. Tampilan Halaman Pencarian Berdasarkan Nama Barang.....	158

I. PENDAHULUAN

A. Latar Belakang

Kegiatan yang dilakukan dari hasil usaha mikro, kecil, dan menengah (UMKM) memberikan dampak yang signifikan terhadap tingkat kewirausahaan di Indonesia, berdasarkan data dari BPS Nomor: 82/Press/SM.3.4/III/2017 tingkat kewirausahaan meningkat dari 1,67% pada tahun 2013/2014 menjadi 3,31% pada tahun 2017, hal ini melebihi target nasional sebesar 2%. Banyaknya wirausaha di Indonesia memberikan dampak yang sangat baik untuk kemajuan suatu negara dan dapat meningkatkan kualitas hidup setiap individu (Depkop, 2017).

Dengan kemajuan teknologi saat ini pada sebuah usaha industri telah diperkenalkan *technopreneurship*. *Technopreneurship* memiliki arti wirausaha yang menggunakan teknologi komunikasi dan informasi dalam pelaksanaannya. Hal ini membuat sebuah wirausaha dapat mempermudah proses bisnis yang dijalankannya. Proses dalam menentukan langkah produksi, pendataan barang, serta mengatur pemodalan dapat dijalankan dengan memanfaatkan teknologi. Kemudahan dalam mengolah data juga keamanan dalam melaksanakan sebuah proses bisnis menjadi hal yang dapat dipertimbangkan (Nurseto, 2010).

CV. Duta Square merupakan perusahaan yang bergerak dalam penjualan pakaian muslim. CV. Duta Square juga menyediakan berbagai alat, sandang, dan kebutuhan sehari-hari. CV. Duta Square memiliki 2 cabang, yang pertama di jalan Teuku Umar dan yang kedua di Simpur Center. Dalam proses transaksi, pendataan barang, dan proses bisnis lainnya perusahaan ini masih menggunakan sistem manual (tradisional).

Selama kurang lebih 5 tahun CV. Duta Square berjalan dengan sistem tradisional, pemilik merasa proses yang berjalan terlalu banyak menghabiskan waktu dan tenaga. Rata-rata proses transaksi tiap bulan adalah 1500 transaksi. Pada sistem yang berjalan saat ini, perusahaan melakukan proses transaksi menggunakan kasir konvensional dan omzet harian dihitung secara manual. Barang yang tersimpan di gudang juga tidak memiliki pendataan sehingga informasi jumlah dan ketersediaan barang tidak dimiliki perusahaan. Hal ini menyebabkan kesulitan dari pihak CV. Duta Square untuk melakukan pendataan barang dan mencatat proses transaksi yang terjadi.

Untuk menangani masalah yang dimiliki CV. Duta Square, maka akan dibuat sistem berbasis POS (*Point of Sales*). POS merupakan sistem dimana terjadinya transaksi jual beli. Proses yang terjadi pada sistem berbasis POS adalah penjual akan menghitung jumlah transaksi yang dilakukan oleh pembeli, dan pembeli akan melakukan pembayaran sesuai jumlah yang telah dihitung. Proses transaksi akan tersimpan pada *database* sistem, dan pembeli akan menerima tanda bukti transaksi.

Dalam implementasinya, sistem informasi *Point of Sales* ini sudah banyak pengembangnya. Sistem Acosys, yaitu sistem POS berbasis desktop merupakan salah satu sistem yang menawarkan fitur transaksi yang cukup baik. Namun menurut pemilik CV. Duta Square, sistem Acosys masih belum memenuhi kebutuhan perusahaan. CV. Duta Square membutuhkan sistem POS yang memiliki data untuk diolah lebih lanjut. Perusahaan juga membutuhkan fitur untuk meningkatkan pelayanan terhadap pembeli.

Pada sistem yang akan dibuat penulis akan menggunakan sistem berbasis web dan akan memasukkan fitur *Customer Relationship Management (CRM)*. *Customer Relationship Management (CRM)* merupakan strategi yang biasa digunakan oleh penjual untuk dijadikan sebagai alat pendekatan terhadap calon pembeli (Dyantina, dkk., 2012).

Pada implementasinya sistem akan dibuatkan fitur untuk memudahkan pembeli dalam pengecekan barang yang tersedia. Dalam implementasi CRM, fitur yang diberikan tersebut akan memberikan kenyamanan untuk pembeli dalam memperoleh informasi produk yang diinginkannya. Sistem juga akan diberikan fitur *barcode scanner* agar dalam pelaksanaan proses *Point of Sales* menjadi lebih cepat dan meningkatkan kenyamanan pembeli.

Selain itu, sistem ini akan saling terintegrasi dengan sistem lainnya sehingga data yang dimiliki dapat diolah dan dimanfaatkan dengan baik oleh sistem lain. Data transaksi yang didapat pada sistem ini akan diberikan kepada Sistem Informasi Akuntansi dan Sistem *Bussiness Intelegence Dashboard* untuk diolah lebih lanjut sebagai sarana perhitungan dan pengambilan keputusan. Data barang yang diberikan pemasok diterima oleh Sistem *Supplier* dan Stok Gudang.

B. Rumusan Masalah

Berdasarkan latar belakang yang penulis paparkan, rumusan masalah yang dapat dikemukakan adalah “Bagaimana merancang sistem *Customer Relationship Management* (CRM) yang memenuhi kebutuhan proses bisnis CV. Duta Square, meningkatkan relasi perusahaan dengan pembeli, dan membuat data penjualan dapat dimanfaatkan lebih lanjut”.

C. Batasan Masalah

Dalam Pengembangan *Customer Relationship Management* (CRM) Pada Sistem Informasi Penjualan di CV. Duta Square Bandar Lampung ini memiliki batasan masalah sebagai berikut:

- a. sistem informasi ini merupakan sistem informasi berbasis web,
- b. terdapat fitur *Point of Sales* (POS) untuk menangani transaksi penjualan yang terjadi pada CV. Duta Square,

- c. terdapat *data center* untuk memudahkan pembeli melihat informasi barang yang tersedia,
- d. terdapat data *member* untuk melihat informasi *customer*,
- e. sistem dibangun dengan *framework* Laravel,
- f. sistem dapat diakses secara *online*,
- g. sistem dibangun menggunakan MySQL sebagai *database*,
- h. sistem ini terhubung dengan sistem lainnya menggunakan API (*Application Programming Interface*),
- i. data barang untuk gudang didapat dari Sistem *Supplier* dan Stok Gudang, dan
- j. untuk mengakses data yang disediakan *web service* dibutuhkan koneksi menuju sistem ini.

D. Tujuan

Tujuan dari penelitian ini adalah membantu pemilik usaha dalam menangani proses bisnis yang berjalan dan mengembangkan Sistem *Customer Relationship Management* (CRM) di CV. Duta Square Bandar Lampung dengan menerapkan *framework* Laravel.

E. Manfaat

Manfaat dari penelitian ini adalah:

1. Bagi *Customer* CV. Duta Square Bandar Lampung
 - a. mempermudah *customer* dalam pencarian stok barang.
2. Bagi pengembang
 - a. hasil dari penelitian ini dapat menambah pengetahuan baru bagi penulis,
 - b. menambah wawasan mengenai CRM (*Customer Relationship Management*),
 - c. dapat dijadikan sebagai acuan untuk mengembangkan atau membuat penelitian yang sama,
 - d. mengembangkan dan menerapkan ilmu yang didapat selama di perkuliahan, dan
 - e. menambah pengetahuan tentang analisis sistem dalam pemrograman web.
3. Bagi instansi terkait (CV. Duta Square Bandar Lampung)
 - a. CV. Duta Square dapat melakukan pendataan pada setiap transaksi yang terjadi,
 - b. CV. Duta Square dapat melihat kualitas relasi terhadap konsumen,
 - c. CV. Duta Square dapat melihat berapa jumlah stok barang yang ada di gudang, dan
 - d. data pada sistem yang dibuat dapat dimanfaatkan oleh sistem lainnya.

II. TINJAUAN PUSTAKA

A. CV. Duta Square

Duta Square didirikan pada tahun 2004 oleh wirausahawan bernama Adrisman. Awal mulanya beliau mendirikan *brand* Duta dengan menjual perhiasan dan busana muslim di toko pertamanya yang berada di Simpur Center. Kemudian melihat pangsa pasar yang cukup besar dalam penjualan Pakaian Muslim, beliau lebih memfokuskan dalam bidang ini.

Pada tahun 2010 *owner* mencoba untuk membuka cabang di Bambu Kuning, dan tahun 2011 membuka cabang lagi di jalan Wolter Monginsidi. Melihat respon pelanggan yang cukup baik dalam pembukaan toko di luar mal, maka *owner* terus mengembangkan bisnisnya. Kemudian tahun 2013 *owner* mencoba untuk membangun toko di daerah Teuku Umar dengan *brand* Duta Square.

Pada tahun 2016 *owner* mendaftarkan *brand* Duta Square sebagai Perseroan Komanditor atau Commanditaire Vennootschap dengan nama CV. Duta Square. Sekarang Duta Square menyediakan pakaian muslim, kebutuhan *fashion*, dan beberapa alat kebutuhan. Dalam pengoperasiannya CV. Duta Square memiliki 48 karyawan dan berlokasi di jalan Teuku Umar.

B. POS (*Point of Sale*)

Point of Sale adalah sistem dimana terjadinya transaksi ritel di antara penjual dan pembeli. Dalam POS penjual akan melakukan pertukaran untuk barang atau jasa yang telah disediakan. Proses yang berlangsung penjual akan menghitung jumlah transaksi oleh pelanggan dan memberikan pilihan kepada pelanggan untuk melakukan pembayaran. Sistem POS juga akan mengeluarkan tanda terima terhadap transaksi yang terjadi (Hidayat, 2014).

C. CRM (*Customer Relationship Management*)

CRM (*Customer Relationship Management*) merupakan sebuah sebutan dalam industri TI yang bertujuan untuk mengelola dan meningkatkan hubungan kepada pelanggan. CRM sendiri dapat diimplementasikan untuk membuat suatu metodologi, strategi, dan perangkat lunak.

Menggunakan CRM perusahaan berkonsentrasi untuk memuaskan pelanggan sehingga pelanggan tidak beralih ke pesaing. Untuk menentukan hal tersebut perusahaan akan mencatat interaksi yang dilakukan oleh pelanggan mulai dari transaksi dan interaksi terhadap *staff marketing* (Hamidin, 2008).

CRM pada dasarnya mengandung 3 aspek yang saling berkaitan. Masing-masing aspek dapat dijelaskan yaitu:

- a. *Customer*, merupakan pengguna layanan barang dan jasa suatu organisasi yang memberikan profit atau manfaat bagi organisasi tersebut,

- b. *Relationship*, merupakan interaksi antara *customer* dan organisasi yang dapat terjadi dalam jangka pendek, jangka panjang, secara terus-menerus, atau hanya sekali. Biasanya *customer* akan melakukan interaksi secara berulang jika telah memiliki rasa percaya atau puas terhadap pelayanan suatu organisasi, dan
- c. *Management*, informasi atau data yang didapat dari *customer* ketika interaksi terjadi. Dengan adanya informasi atau data tersebut organisasi dapat meningkatkan pelayanan yang dimiliki (Budiardjo, dkk., 2008).

D. PHP

PHP (singkatan dari PHP: *Hypertext Preprocessor*) adalah bahasa *scripting opensource* yang digunakan dalam pengembangan web dan dapat saling berintegrasi dengan HTML. Tidak seperti C atau Perl PHP memberikan fitur yang mudah untuk berintegrasi dengan HTML. Dengan kode “<?php dan ?>” pengembang dapat menggunakan script PHP secara fleksibel.

Kode PHP dijalankan dalam sisi-*server* sehingga hal ini yang membedakannya dengan bahasa JavaScript (Php, 2019).

E. Laravel

Laravel merupakan *framework opensource* berbasis MVC (*Model View Controller*) menggunakan bahasa pemrograman PHP untuk pembuatan web

aplikasi dengan sintaks yang terstruktur. Laravel dibuat agar pengembang dapat bereksplorasi dan mendapat pengalaman yang baik dalam tahap *developing* web aplikasi. Untuk mencapai tujuan tersebut laravel memberikan fitur yang bertujuan untuk mempermudah pengembang seperti otentikasi, *routing*, *session*, dan *caching*. Laravel mudah untuk diakses namun tetap *powerful*, *framework* ini memiliki *tools* dan *library* yang sangat luas dan banyak pengembangnya (Otwell, 2012). Logo Laravel dapat dilihat pada Gambar 1.



Gambar 1. Logo Laravel.

Menurut survei yang dilakukan *sitepoint*, *framework* PHP terbaik pada tahun 2014 adalah Laravel. Survei tersebut menghasilkan sebuah data yang berupa hasil persentase, Laravel menduduki posisi pertama dengan hasil persentasenya sebanyak (25.85%), lalu diurutan kedua adalah Phalcon dengan presentase sebanyak (16.73%) dan posisi ketiga adalah Symfony2 dengan presentase sebanyak (10.62%). Laravel menang dalam penilaian *reload* data dan tingkat keamanan yang baik. Dokumentasi dan komunitas yang dimiliki *framework* Laravel juga sangat baik (*sitepoint*, 2013).

F. CSS

CSS (singkatan dari *Cascading Style Sheet*) adalah *stylesheet language* yang digunakan untuk mendeskripsikan penyajian dari dokumen yang dibuat dalam *mark up language*. CSS merupakan sebuah dokumen yang berguna untuk melakukan pengaturan pada komponen halaman *web*, inti dari dokumen ini adalah memformat halaman *web* standar menjadi bentuk *web* yang memiliki kualitas yang lebih indah dan menarik. (Yusi, dkk, 2012).

G. Bootstrap

Bootstrap pada awalnya dibuat dan dikembangkan oleh *desainer* dan pengembang di Twitter. Bootstrap merupakan *framework front-end* yang populer dalam pengembangan proyek *open source*. *Framework* ini dibuat pada pertengahan tahun 2010 oleh @mdo dan @fat. Sebelum menjadi *framework*, Bootstrap dikenal sebagai Twitter Blueprint. Bootstrap menjadi *framework* yang memudahkan penggunanya membuat “*style*” CSS dengan mudah dan fleksibel.

Bootstrap telah melakukan dua puluh rilis, termaksud dua penulisan ulang utama. Pada perbaikan Bootstrap 2, pengembang menambah fungsionalitas responsif ke seluruh *framework* sebagai *stylesheet* optional. Pada perbaikan Bootstrap 3, pengembang menulis ulang *library* untuk membuat Bootstrap responsif secara *default* dengan mendukung fitur *mobile* (Bootstrap, 2014).

H. Database

Database (Basis Data) merupakan sekumpulan data yang tersimpan dalam penyimpanan komputer. Penggunaan *Database* berfungsi sebagai penyedia informasi dan tempat penyimpanan berbagai jenis data untuk pemakainya. Setiap data pada *database* saling terintegrasi antara satu dan lainnya, dan diproses oleh perangkat lunak tertentu (Speed & Engineering, 2009).

I. MySQL

MySQL merupakan sistem manajemen *database opensource* yang dikembangkan, didistribusikan, dan didukung oleh Oracle Corporation. *Server* MySQL dapat digunakan secara fleksibel, seperti berjalan di desktop atau laptop, disamping aplikasi lain, *server web*, dan lainnya. *Server* MySQL awalnya dikembangkan untuk menangani *database* berskala besar.

Server MySQL berfungsi di *client* atau *server* atau sistem *embedded*. *Database* MySQL terdiri dari *server* SQL multithread yang mendukung jalannya sistem secara *back-end*, beberapa sistem *client* atau *server* dengan bermacam-macam *library*, dan berbagai *interface* aplikasi berbasis API (Oracle, 2018).

J. phpMyAdmin

phpMyAdmin adalah *software open source* yang ditulis menggunakan pemrograman php, yang bertujuan untuk menangani administrasi dari MySQL

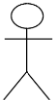




melalui web. *Software* phpMyAdmin juga menyediakan dukungan terhadap MariaDB. Pengoperasian *database* seperti mengelola basis data, tabel, kolom, relasi, indeks, perizinan, dan lain-lain dapat dilakukan melalui GUI *interface* namun *user* juga tetap dapat menggunakan *sintaks* SQL dalam penggunaannya (Phpmyadmin, 2018).

K. Unified Modeling Language (UML)

1. Use case diagram

Use case diagram merupakan sebuah interaksi atau hubungan antar aktor dengan sistem. Dalam pembuatan *use case diagram* dapat membantu pekerjaan kita dalam menyusun sebuah *requirement* pada sistem tertentu. Pada Tabel 1 menjelaskan tentang simbol-simbol dari *use case diagram*.


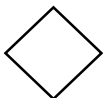


Tabel 1. Simbol-simbol *Use Case Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Mewakili peran orang, sistem yang lain atau alat ketika berkomunikasi dengan <i>use case diagram</i> .
2		<i>Use Case</i>	Abstraksi dan interaksi antara sistem dan juga <i>actor</i> .
3		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
4		<i>Include</i>	Menunjukkan bahwa suatu <i>use case diagram</i> seluruhnya merupakan fungsionalitas dari <i>use case diagram</i> lainnya.
5		<i>Extend</i>	Hubungan generalisasi antar <i>use case diagram</i> , hal itu menunjukkan bahwa <i>use case diagram</i> yang satu merupakan spesialisasi.

2. Activity diagram

Activity diagram merupakan alur dari aktivitas dalam sebuah sistem yang sedang dirancang, dalam *activity diagram* dijelaskan mulai dari tahapan awal sampai dengan tahapan akhir. *Activity diagram* juga dapat diartikan sebagai *state diagram* khusus, dimana sebagian besar dari *state* yaitu *action* dan sebagian besarnya lagi transisi di-*trigger* oleh selesainya *state* sebelumnya (*internal processing*). Simbol-simbol yang dapat digunakan dalam membuat *activity diagram* dapat dilihat pada Tabel 2.

Tabel 2. Simbol-simbol Activity Diagram

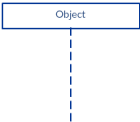
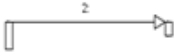


NO	GAMBAR	NAMA	KETERANGAN
1		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain.
2		<i>Decision</i>	<i>Decision</i> merupakan pengambilan sebuah keputusan untuk memilih pilihan yang lebih dari satu.
3		<i>Start Point</i>	Titik awal dari sebuah <i>activity diagram</i> .
4		<i>End Point</i>	Titik akhir dari sebuah <i>activity diagram</i> .

3. Sequence diagram

Sequence diagram yaitu hubungan atau interaksi antar objek yang ada di dalam dan sekitar sistem. Interaksi itu berupa pesan atau *message* yang

ditampilkan dalam kurun waktu tertentu. *Sequence diagram* dapat diartikan pula sebagai skenario yang menggambarkan rangkaian langkah-langkah yang nantinya akan menghasilkan *output* tertentu. Ada beberapa contoh simbol-simbol *sequence diagram* diantaranya dapat dilihat pada Tabel 3.

Tabel 3. Simbol-simbol *Sequence Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>LifeLine</i>	objek <i>entity</i> , antarmuka yang saling berinteraksi.
2		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi.
3		<i>Object Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi.
4		<i>Message to Self</i>	Menggambarkan pesan atau hubungan antar objek yang menunjukkan urutan kejadian yang terjadi.



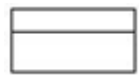
4. *Class diagram*

Class diagram merupakan spesifikasi yang nantinya akan menghasilkan suatu objek tertentu dan menggambarkan atribut atau properti dari sistem.

Class diagram dapat diartikan juga sebagai implementasi dari sebuah

interface, yaitu *class* abstrak yang hanya mempunyai metode. *Class diagram* menggambarkan struktur dan deskripsi *class*, *package* dan obyek beserta hubungan satu sama lain seperti *containment*, pewarisan, asosiasi, dan lain-lain (Mahdiana, 2011). Simbol-simbol yang dapat digunakan dalam pembuatan *class diagram* dapat dilihat pada Tabel 4.

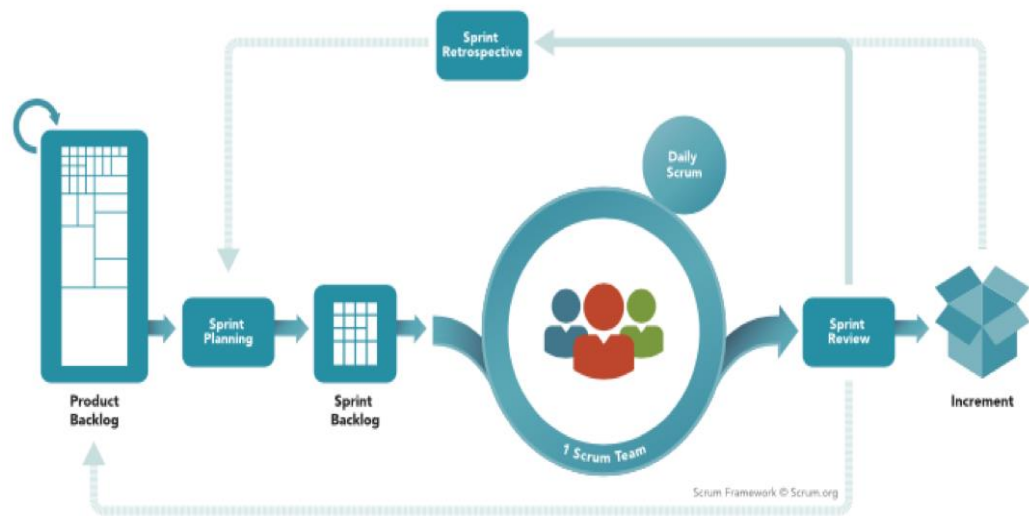
Tabel 4. Simbol-simbol *Class Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Generalization</i>	Hubungan antar <i>class</i> yang lebih umum dengan <i>class</i> spesifik, yang kemudian digabungkan menjadi <i>superclass</i> .
2		<i>Realization</i>	Kegiatan yang benar-benar terjadi atau dilakukan oleh suatu objek.
3		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.

L. Metodologi Pengembangan Sistem

Dalam proses perancangan sistem *Customer Relationship Management*, penulis memilih metodologi pengembangan sistem scrum. Metode tersebut dipilih dikarenakan metode ini dianggap sebagai metode yang paling sesuai saat mengerjakan secara berkelompok. Tahapan metode Scrum dapat dilihat pada Gambar 2.

SCRUM FRAMEWORK



Gambar 2. Tahapan Penggunaan Metode Scrum.

Sumber: <https://www.scrum.org/resources/what-is-scrum>

1. *Product Backlog*

Tahapan pertama adalah 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*

Pada tahap selanjutnya, ketika semua kebutuhan sudah dianalisis maka dilanjutkan dengan berdiskusi mengenai perencanaan dimana masing-masing dari tim menentukan bagian-bagian yang akan mereka kerjakan.

3. *Sprint Backlog*

Setelah melakukan diskusi mengenai tugas dari masing-masing tim. Tahapan ini akan menampilkan seluruh target yang akan dicapai, yang nantinya akan dipantau peningkatannya. Pada tahapan ini setidaknya ada satu peningkatan.

4. *Daily Scrum*

Merupakan proses yang dilakukan oleh tim pengembang saat *sprint* berlangsung. Setiap harinya tim pengembang akan membahas pekerjaan yang telah dilakukan dan kegiatan 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*.
- Hal yang akan menjadi hambatan untuk tim pengembang atau diri sendiri dalam penyelesaian *sprint goal*.

5. *Sprint Review*

Merupakan proses yang dilakukan untuk meninjau apa saja yang telah diselesaikan saat tahapan *sprint*. Berdasarkan hasil diskusi ditentukan proses yang akan dilakukan selanjutnya untuk meningkatkan nilai bisnis.

6. *Increment*

Merupakan hasil dari *Product Backlog* Item 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).

III. METODE PENELITIAN

A. Waktu dan Tempat

Penelitian ini dilaksanakan di Jurusan Ilmu Komputer Universitas Lampung dan CV. Duta Square Bandar Lampung. Waktu penelitian dimulai pada Semester Ganjil Tahun Ajaran 2018/2019 pada bulan September 2018 dan direncanakan selesai bulan Agustus 2019.

B. Alat Penelitian

Alat yang dibutuhkan dalam penelitian ini adalah:

1. Perangkat lunak (*software*)

Perangkat lunak yang digunakan penulis untuk membuat sistem informasi ini adalah.

a. Sistem Operasi: *Windows 10 Home Single Language* 64 bit.

b. Aplikasi:

- Corel Draw 2017 sebagai aplikasi pengolah grafis.
- XAMPP versi 7.1.10-0 sebagai aplikasi *local server* untuk melakukan pengembangan sistem MYSQL dan PHP.

- *Atom Text Editor* versi 1.25.1 sebagai aplikasi untuk menulis *code* program.
- *Web Browser Google Chrome* dan *Microsoft Edge* sebagai aplikasi untuk menguji dan menjalankan sistem.
- *Star UML* sebagai aplikasi untuk membuat diagram model sistem informasi.

c. Web Aplikasi

- *Application Programming Interface* sebagai pertukaran data antar sistem.
- *Draw.io* sebagai aplikasi untuk membuat diagram.

d. *Framework*

- *Laravel* sebagai *framework* dalam pembuatan *website* untuk mengatur logika dan kerja sistem dengan bahasa pemrograman PHP.
- *Bootstrap* sebagai *framework* berbasis CSS yang berguna untuk mengatur elemen html dan memperindah tampilan *website*.
- *Vue.js* sebagai *framework* berbasis javascript untuk membuat aplikasi menjadi dinamis dan interaktif.

2. Perangkat keras (*hardware*)

Perangkat keras yang digunakan penulis untuk membuat sistem informasi ini adalah:

a. Laptop Dell Inspiron 7460

- Intel® Core™ i7-7200U CPU @2.60GHz
- 8 GB RAM

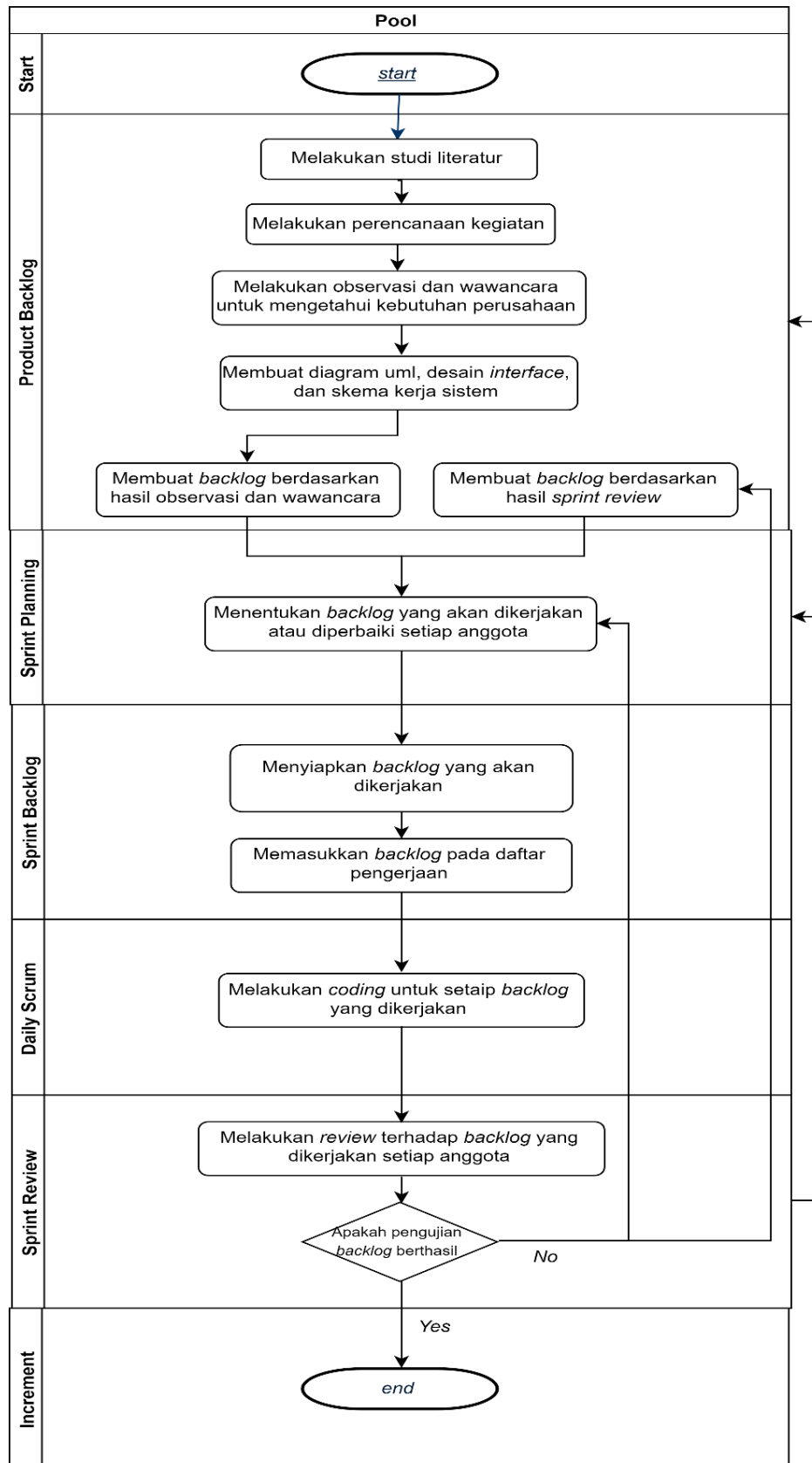
- 1 TB HDD
- b. Internet IndiHome 10Mb
- c. Komputer *Server* Dell
- Intel® Xeon E5-2609V4
 - 8 GB Ram
 - 1 TB HDD

C. Metodologi Penelitian

Scrum merupakan metodologi yang paling cocok digunakan secara berkelompok. Ada beberapa tahapan yang kami lakukan dalam mengerjakan sistem dengan menggunakan metode Scrum.

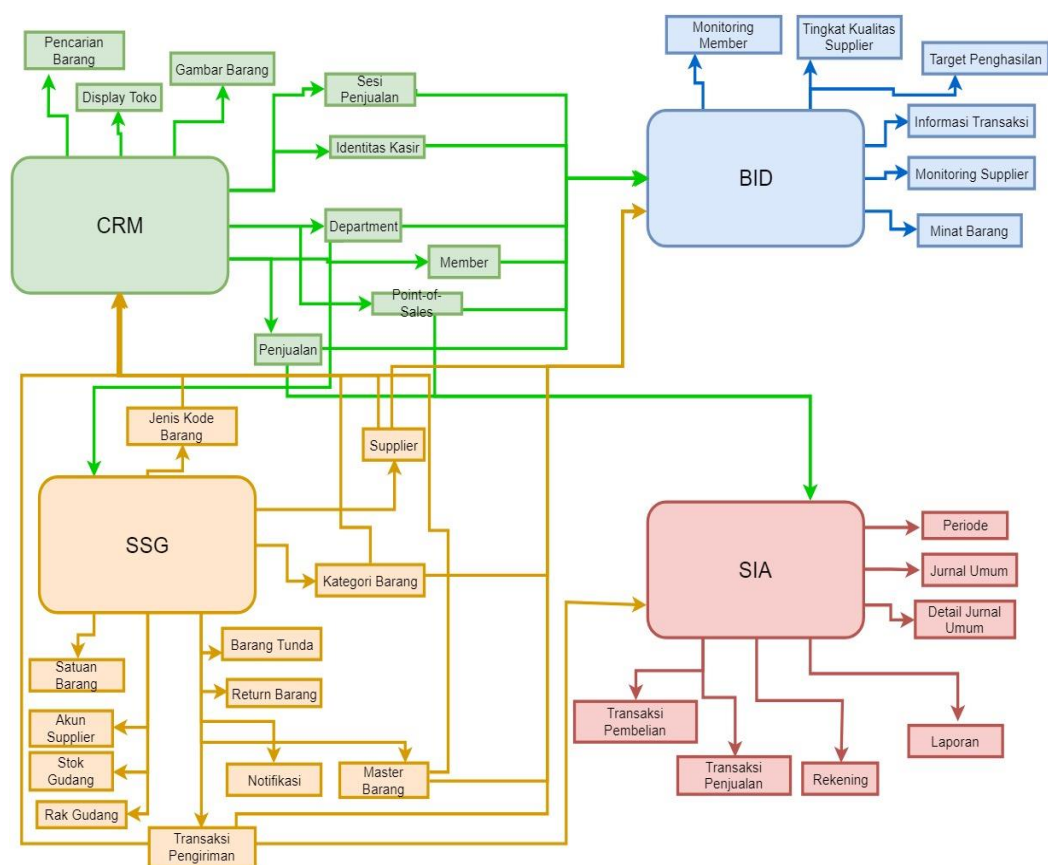
Tahapan atau alur metode Scrum dapat dilihat pada Gambar 3. Pada tahapan Scrum terdapat Scrum Tim, dimana Scrum Tim memiliki pembagian masing-masing tugas. Pembagian tugas Scrum Tim yaitu

<i>Product Owner</i>	: Yumilia Harianti
Scrum Master	: Fikri Ahmad Maulana
Team	: 1. Elin Aprilliana
	2. Jannati Asri Safitri
	3. Dina Munjiati



Gambar 3. Alur Metode Scrum.

Sistem CRM (*Customer Relationship Management*) dalam pengembangannya akan terhubung ke Sistem SSG (*Supplier dan Stok Gudang*), Sistem BID (*Bussiness Intelligence 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 4.



Gambar 4. Hubungan Antar Sistem.

Gambar 4 menunjukkan hubungan antar sistem yang saling mengirim dan menerima data dari satu sistem ke sistem lainnya. Data yang dikirim dan diterima di antaranya sebagai berikut:

1. Sistem Informasi *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*, 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.

1. *User stories*

User stories merupakan deskripsi mengenai kebutuhan sistem dalam bahasa natural yang mudah dipahami oleh *end user* yang tidak memiliki *background* TI. Dalam penggunaan metode Scrum hal yang harus ditentukan oleh *Product Owner* adalah *user stories*. *User stories* digunakan untuk membuat *backlog*. Salah satu hal yang paling penting dari *user stories* ialah kalaborasi antara orang teknis dan orang bisnis untuk mencapai harapan dari pengembangan *software* (Kurniawan & Sari, 2019).

User Stories Owner

- a. sebagai seorang *owner*, saya ingin *customer* dapat mencari barang sesuai dengan yang diinginkan, sehingga mempermudah *customer* dalam mengetahui stok barang yang sedang dicari,

- b. sebagai seorang *owner*, saya ingin data sesi penjualan dicatat secara detail mulai dari dibukanya sesi penjualan sampai ditutupnya sesi penjualan, sehingga saya dapat memantau langsung data sesi penjualan,
- c. sebagai seorang *owner*, saya ingin data penjualan dicatat mulai dari nama barang yang dibeli, jumlah barang yang dibeli, total harga barang, sampai dengan tercetaknya nota, sehingga memudahkan dalam proses penjualan,
- d. sebagai seorang *owner*, saya ingin data yang menampilkan nama barang dan stok barang yang sudah otomatis terpotong dari penjualan,
- e. sebagai seorang *owner*, saya ingin data *member* dicatat secara detail, sehingga saya dapat melihat langsung nama-nama *member* yang terdaftar,
- f. sebagai seorang *owner*, saya ingin mencetak *barcode* barang, sehingga memudahkan dalam pebelan barang.

2. Product backlog

Product backlog yang dimiliki Sistem CRM (*Customer Relationship Management*), Sistem SSG (*Supplier dan Stok Gudang*), Sistem BID (*Business Intelligence Dashboard*), dan Sistem Informasi Akuntansi (SIA) dapat dilihat pada Gambar 5.



Gambar 5. *Product Backlog* Masing-masing Sistem.

Penjelasan dari setiap langkah-langkah yang ada pada metode Scrum yaitu:

1. *Product Backlog*

Pada *product backlog* tim membuat daftar fitur yang diinginkan oleh *product owner*. Deskripsi *backlog* dapat dilihat pada Gambar 5.

2. *Sprint Planning*

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

3. *Sprint Backlog*

Sprint backlog dilakukan saat *sprint planning* telah dilaksanakan dan dipantau oleh *scrum master*. Pada tahapan ini setiap *backlog* yang ingin dilaksanakan dimasukkan pada kolom *to do*.

4. *Daily Scrum*

Pada tahapan ini tim akan melaksanakan *backlog* yang telah direncanakan dan tim akan lebih fokus untuk mendapatkan *sprint goal*.

5. *Sprint Review*

Pada tahapan *sprint review* ini tim akan meninjau hasil dari *sprint* yang dilakukan, tahapan 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.

D. Metodologi Pengumpulan Data

Metodologi pengumpulan data yang dilakukan penulis dalam proses pembuatan sistem informasi ini adalah:

1. Observasi

Observasi yaitu pengamatan data secara langsung dengan melakukan analisa pada proses transaksi jual beli yang dilakukan di CV. Duta Square Bandar Lampung. Observasi sudah dilaksanakan sebanyak 3 kali, yang pertama observasi tentang proses bisnis CV. Duta Square Bandar Lampung, yang kedua dan ketiga observasi tentang *backlog* sistem.

2. Wawancara

Wawancara yaitu metode yang digunakan untuk memperoleh informasi dengan cara berinteraksi dengan narasumber yang terkait pada proses bisnis dan memberikan pertanyaan-pertanyaan untuk mendapatkan informasi yang lebih detail. Wawancara sudah dilakukan sebanyak 4 kali, yang pertama wawancara mengenai sejarah CV. Duta Square, yang kedua wawancara tentang masalah yang terjadi di CV. Duta Square, kemudian yang ketiga tentang kebutuhan dari CV. Duta Square dan yang keempat memvalidasi kebutuhan tersebut.

E. Analisis Sistem

1. Analisis Masalah

CV. Duta Square merupakan salah satu perusahaan yang bergerak dalam penjualan pakaian muslim. Selama kurang lebih 5 tahun CV. Duta Square berjalan dengan sistem tradisional, pemilik toko merasa proses yang berjalan

terlalu banyak menghabiskan waktu dan tenaga. Rata-rata proses transaksi setiap bulannya bisa mencapai 1500 transaksi.

Setelah dilakukan wawancara dengan pemilik toko sebanyak 4 kali, pemilik toko menemukan beberapa masalah yang sering terjadi pada saat proses transaksi jual beli. Seiring berjalannya waktu penjualan pakaian muslim akan terus menerus bertambah. Dengan begitu masalah akan terus bertambah jika tidak diselesaikan satu persatu. Salah satu masalah yang terjadi pada saat proses transaksi jual beli berlangsung yaitu, tidak tercatatnya stok barang sehingga pemilik toko tidak mengetahui berapa sisa barang yang masih tersedia dan juga *customer* tidak bisa mencari stok barang yang sedang mereka cari.

2. Analisis Kebutuhan Sistem

Dalam upaya menyelesaikan masalah proses transaksi jual beli yang sudah dijelaskan, maka akan dibuat sistem yang dapat membantu menyelesaikan masalah-masalah yang terjadi. Sistem yang akan dibuat yaitu Sistem *Customer Relationship Management* (CRM) dimana sistem ini dapat membantu *customer* pada saat melakukan pencarian barang dan mencatat proses transaksi jual beli.

Kerangka komponen CRM diklasifikasikan menjadi tiga:

- Operasional dikenal sebagai *front office* perusahaan yang berperan dalam interaksi dengan pelanggan. CRM operasional mencakup proses

otomatisasi yang terintegrasi dari keseluruhan proses bisnis, seperti otomatisasi pemasaran, transaksi dan pelayanan. Salah satu penerapan CRM yang termasuk dalam kategori operasional CRM adalah dalam bentuk aplikasi web. Melalui web, suatu perusahaan dapat memberikan pelayanan kepada konsumen.

- Analitikal dikenal sebagai *back office* perusahaan yang berperan dalam memahami kebutuhan konsumen. CRM analitik berperan dalam melaksanakan analisis konsumen dan pasar, seperti analisis *trend* pasar dan analisis perilaku konsumen. Data yang digunakan pada CRM analitik adalah data yang berasal dari CRM operasional.
- Collaborative meliputi e-mail, *personalized publishing*, *e-communities*, dan sejenisnya yang dirancang untuk interaksi antara konsumen dan organisasi. Tujuan utamanya adalah menyemangati dan menyebarkan loyalitas konsumen ke konsumen lain yang masih belum berada di level kesetiaan pelanggan. Collaborative juga mencakup pemahaman atau kesadaran bahwa konsumen yang setia dapat menjadi magnet bagi konsumen lain (Kurniawan, 2009).

Ketiga komponen tersebut akan diimplementasikan ke dalam Sistem *Customer Relationship Management* (CRM). Rencana implementasi CRM dapat dilihat pada Tabel 5.

Tabel 5. Rencana Implementasi CRM

No	Komponen	Rencana Implementasi Pada Sistem
1	<i>Operational</i>	<ul style="list-style-type: none"> - Dalam melakukan transaksi penjualan <i>customer</i> melalui proses <i>point-of-sales</i>. - Pelanggan dapat mencari informasi barang berdasarkan merk, kategori, dan kode barang.
2	<i>Analytical</i>	<ul style="list-style-type: none"> - Pemilik usaha mendapatkan data jumlah pembelian setiap <i>customer</i>. - Pemilik usaha mendapatkan data total penjualan disetiap hari sesi penjualan. - Pemilik usaha mendapatkan data barang yang dibeli disetiap transaksi lengkap dengan waktu dan tanggal penjualan.
3	<i>Collaborative</i>	<ul style="list-style-type: none"> - Setiap <i>customer</i> belanja dan memiliki kartu <i>member</i> maka <i>customer</i> akan mendapatkan tambahan poin. - <i>Customer</i> dapat melihat informasi letak barang pada <i>display</i> dan gambar barang yang dicari.

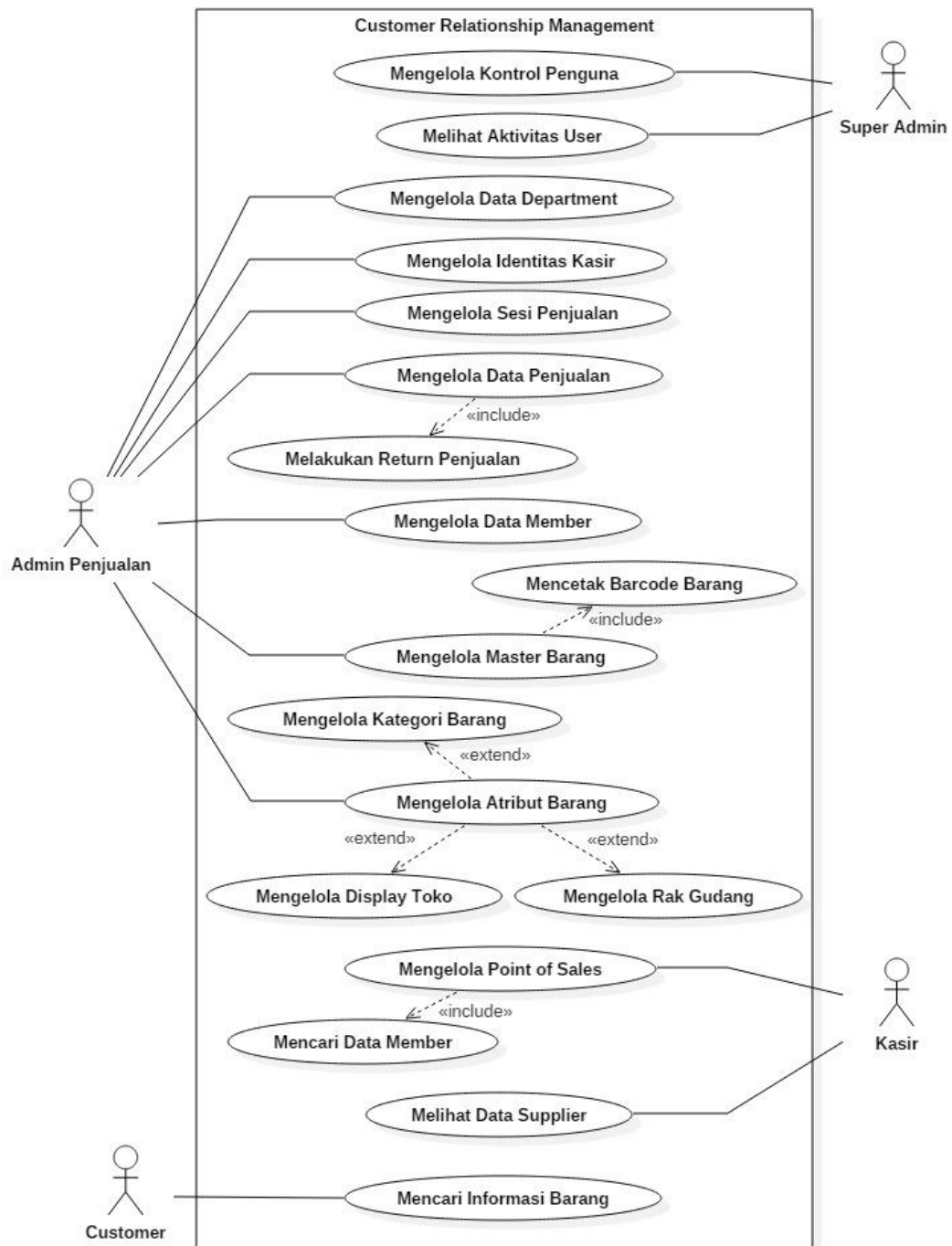
F. Perancangan Sistem

Tahap selanjutnya yang dilakukan pada proses pembuatan sistem informasi ini yaitu perancangan desain sistem. Perancangan ini meliputi; perancangan *use case diagram*, *activity diagram*, *class diagram*, dan *interface*.

1. *Use case diagram*

Use case diagram adalah suatu model yang sangat fungsional dalam sebuah sistem yang menggunakan aktor dan *use case*. Dalam pembuatan *use case diagram* dapat membantu pekerjaan kita dalam menyusun sebuah

requirement pada sistem tertentu. Use case diagram sistem Customer Relationship Management dapat dilihat pada Gambar 6.



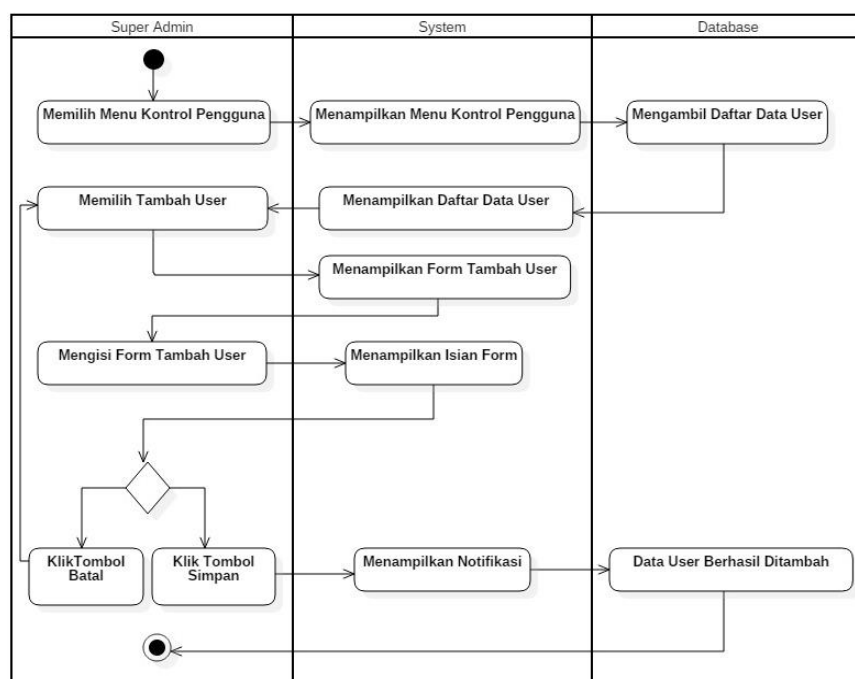
Gambar 6. Usecase Diagram Sistem CRM.

2. Activity diagram

Activity diagram adalah suatu diagram menggambarkan tentang aktifitas yang terjadi pada sistem. Dari pertama sampai akhir, diagram ini menunjukkan langkah-langkah dalam proses kerja sistem yang dikembangkan secara detail.

2.1. Activity Diagram Super Admin Manambah User

Pada *activity* menambah *user* menjelaskan proses penambahan *user*. Super Admin melakukan penambahan *user* dengan mengisi *form* yang telah disediakan. Ketika semua data sudah terisi maka data tervalidasi dan data tersimpan ke *database*. *Activity diagram* menambah *user* dapat dilihat pada Gambar 7.

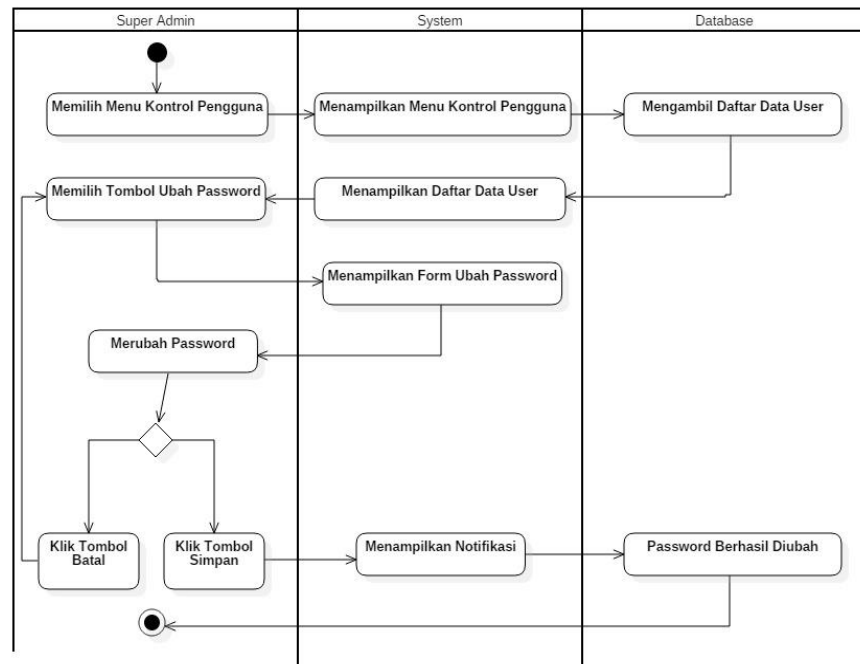


Gambar 7. Activity Diagram Manambah User.

2.2. Activity Diagram Super Admin Mengedit Password User

Pada *activity diagram* ini menjelaskan bagaimana cara merubah *password user*. Dalam proses ini jika ingin merubah *password user* pengguna harus memasukkan *password super admin* terlebih dahulu.

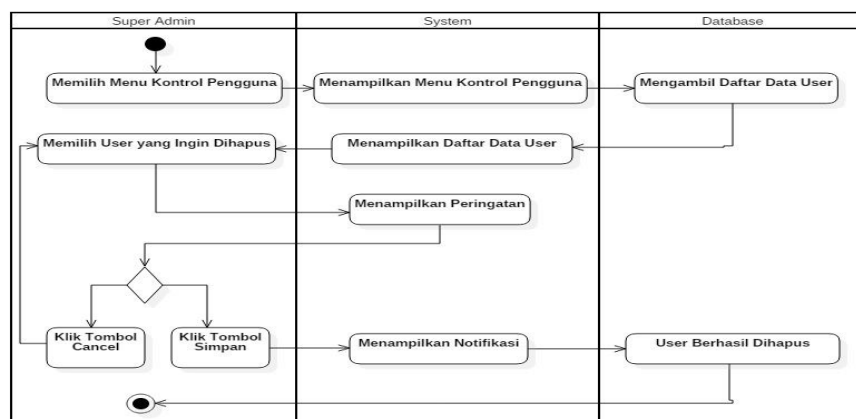
Activity diagram dapat dilihat pada Gambar 8.



Gambar 8. Activity Diagram Mengedit Password User.

2.3. Activity Diagram Super Admin Menghapus User

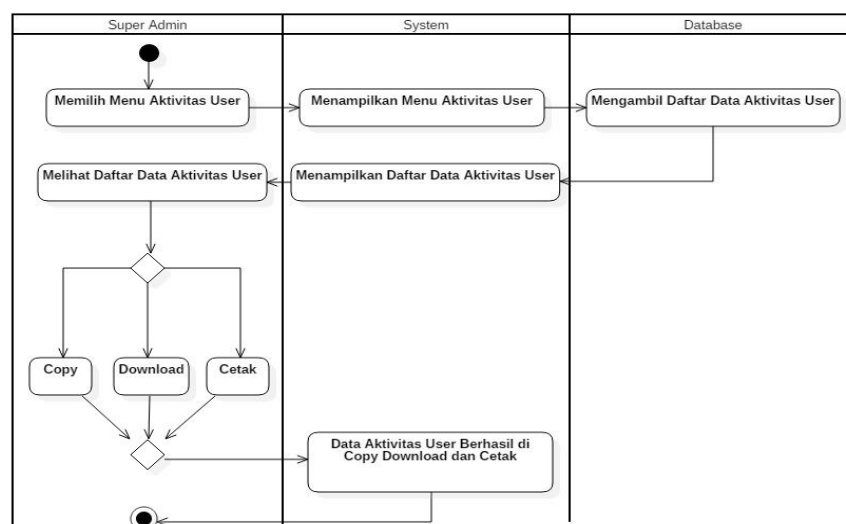
Pada *activity diagram* menghapus data *user* menjelaskan tahapan-tahapan menghapus *user* mulai dari membuka daftar *user* sampai dengan *user* berhasil dihapus. Pada Gambar 9 menjelaskan tentang *activity diagram* menghapus *user*.



Gambar 9. Activity Diagram Menghapus User.

2.4. Activity Diagram Super Admin Melihat Aktivitas User

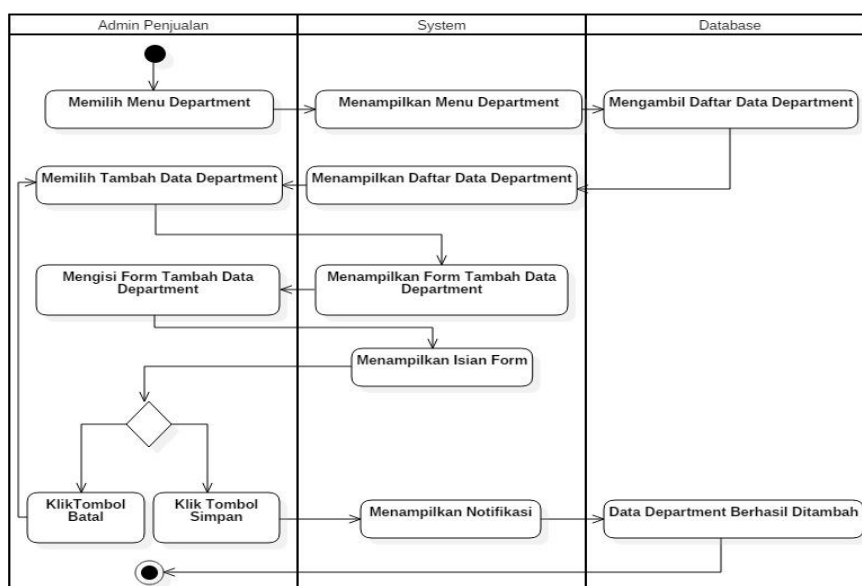
Pada *activity diagram* ini menjelaskan bagaimana proses melihat aktivitas *user*. Sistem akan menampilkan semua daftar aktivitas *user* yang dilakukan oleh semua pengguna. Daftar data aktivitas *user* dapat di-copy, di-download, dan di-print. Pada Gambar 10 menjelaskan tentang *activity diagram* melihat aktivitas *user*.



Gambar 10. Activity Diagram Melihat Aktivitas User.

2.5. Activity Diagram Admin Penjualan Menambah Data *Department*

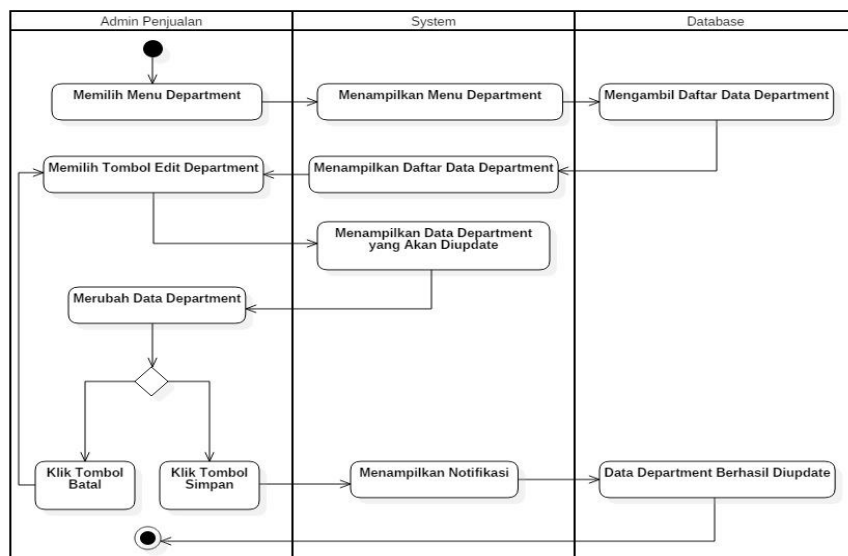
Pada *activity diagram* menambah data *department* menjelaskan proses penambahan data *department*. Admin Penjualan melakukan penambahan data *department* dengan mengisi *form* yang telah disediakan. Ketika semua data sudah terisi maka data tervalidasi dan data tersimpan ke *database*. *Activity diagram* menambah data *department* dapat dilihat pada Gambar 11.



Gambar 11. Activity Diagram Menambah Data *Department*.

2.6. Activity Diagram Admin Penjualan Mengedit Data *Department*

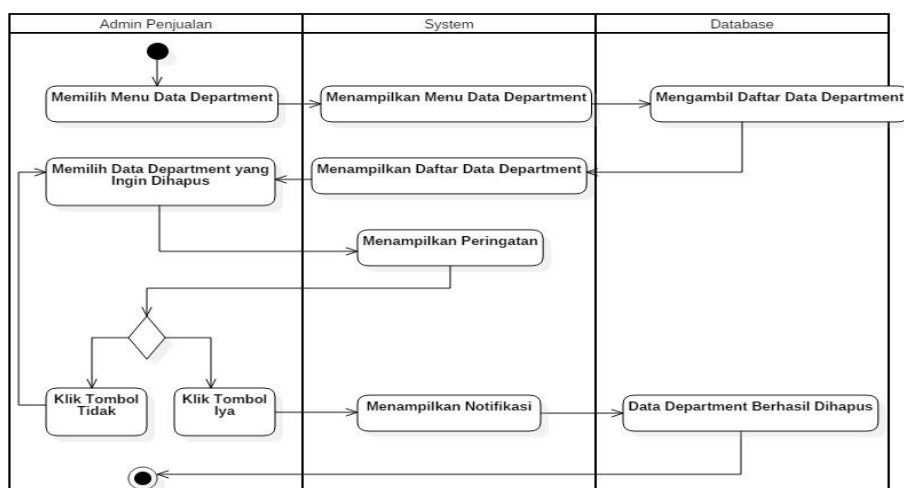
Pada *activity diagram* ini admin penjualan dapat mengubah data setiap *department* yang tersimpan pada *database*. *Activity diagram* dapat dilihat pada Gambar 12.



Gambar 12. Activity Diagram Mengedit Data Department.

2.7. Activity Diagram Admin Penjualan Menghapus Data Department

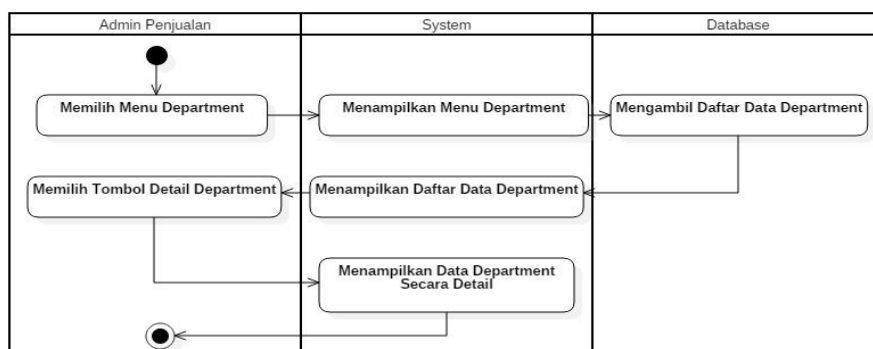
Pada *activity diagram* ini admin penjualan dapat menghapus data *department* yang tersimpan pada *database*. Activity diagram dapat dilihat pada Gambar 13.



Gambar 13. Activity Diagram Menghapus Data Department.

2.8. Activity Diagram Admin Penjualan Melihat Detail *Department*

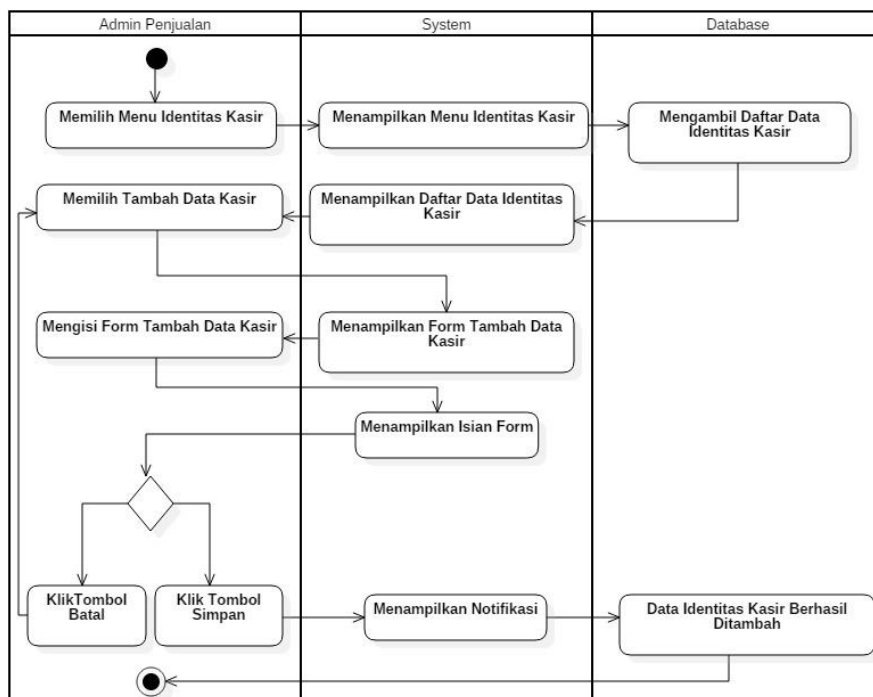
Pada *activity diagram* ini admin penjualan dapat melihat detail data dari setiap *department* yang tersimpan pada *database*. *Activity diagram* dapat dilihat pada Gambar 14.



Gambar 14. Activity Diagram Melihat Detail *Department*.

2.9. Activity Diagram Admin Penjualan Menambah Identitas Kasir

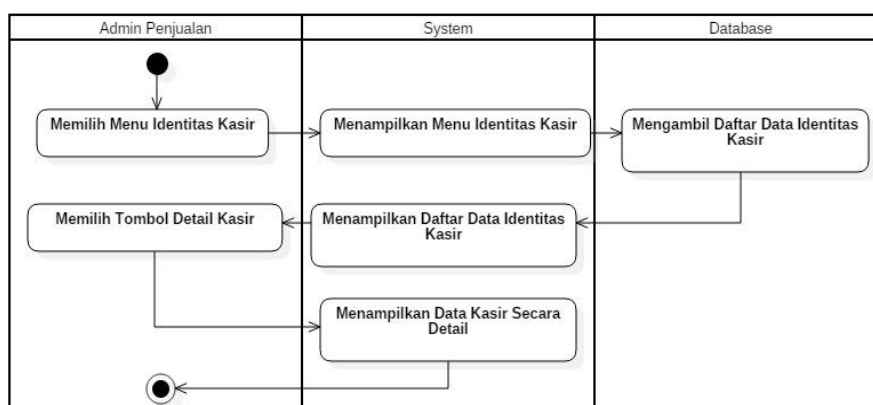
Pada *activity diagram* menambah data identitas kasir menjelaskan proses penambahan data identitas kasir. Admin Penjualan melakukan penambahan data identitas kasir dengan mengisi *form* yang telah disediakan. Ketika semua data sudah terisi maka data tervalidasi dan data tersimpan ke *database*. *Activity diagram* menambah data identitas kasir dapat dilihat pada Gambar 15.



Gambar 15. Activity Diagram Menambah Identitas Kasir.

2.10. Activity Diagram Admin Penjualan Melihat Data Identitas Kasir

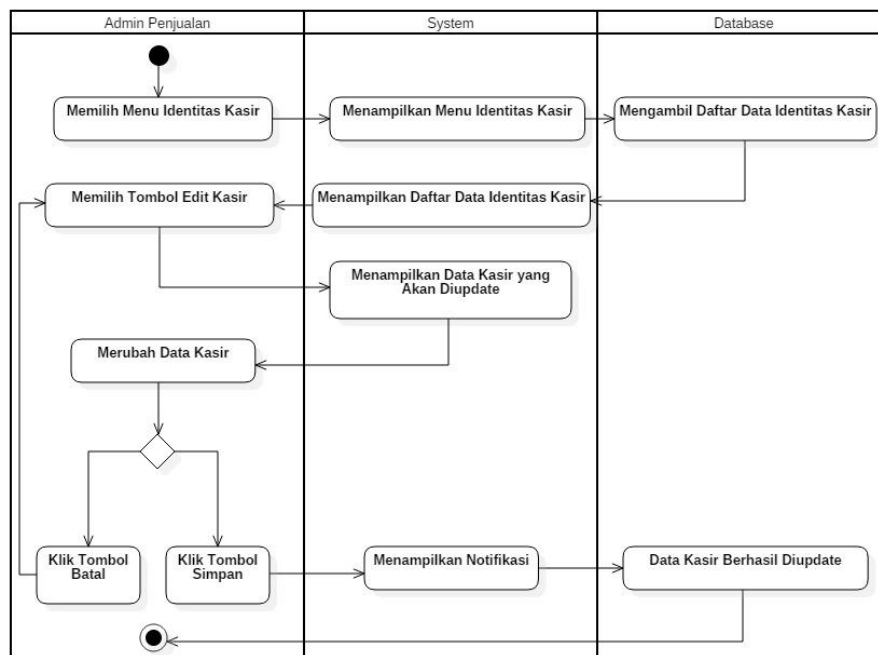
Pada *activity diagram* ini admin penjualan dapat melihat data identitas kasir secara detail yang tersimpan pada *database*. *Activity diagram* dapat dilihat pada Gambar 16.



Gambar 16. Activity Diagram Melihat Data Identitas Kasir.

2.11. Activity Diagram Admin Penjualan Mengedit Data Identitas Kasir

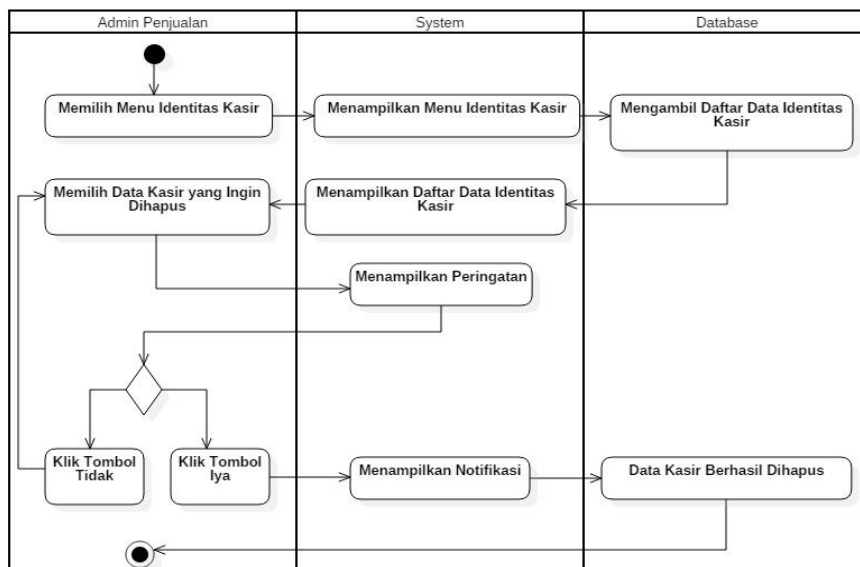
Pada *activity diagram* ini admin penjualan dapat mengubah data identitas kasir yang tersimpan pada *database*. *Activity diagram* dapat dilihat pada Gambar 17.



Gambar 17. Activity Diagram Mengedit Data Identitas Kasir.

2.12. Activity Diagram Admin Penjualan Menghapus Data Kasir

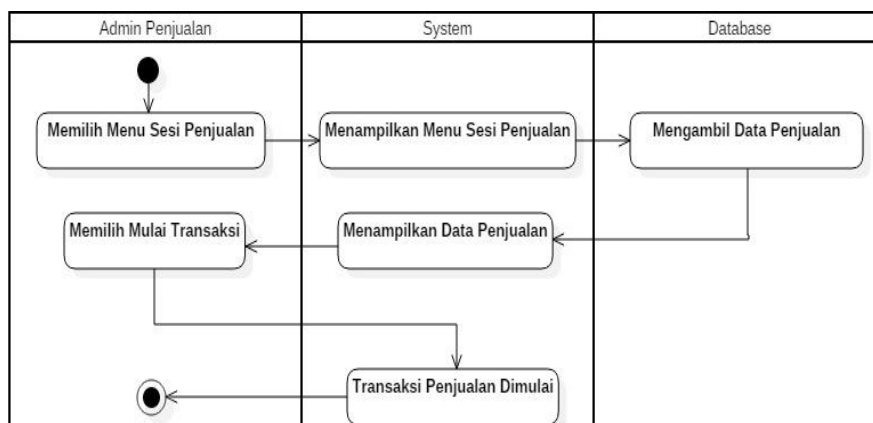
Pada *activity diagram* ini admin penjualan dapat menghapus data kasir yang tersimpan pada *database*. *Activity diagram* dapat dilihat pada Gambar 18.



Gambar 18. Activity Diagram Menghapus Data Kasir.

2.13. Activity Diagram Admin Penjualan Mulai Transaksi

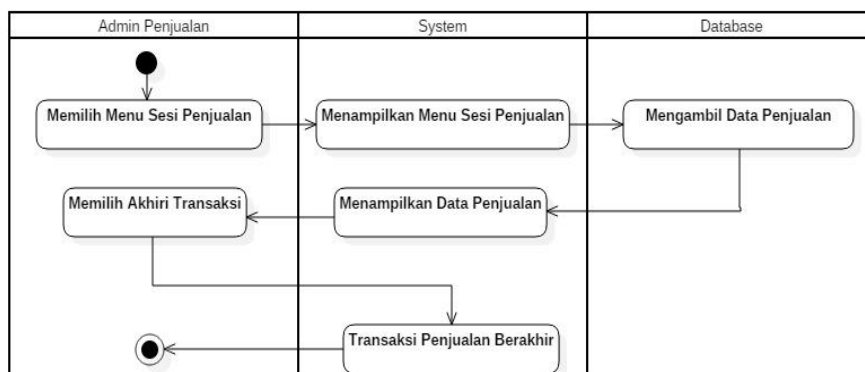
Pada *activity diagram* mulai transaksi penjualan ini menjelaskan bagaimana proses memulai transaksi penjualan. Ketika menu sesi penjualan sudah dibuka oleh Admin Penjualan, maka menu *point of sales* sudah dapat diakses. *Activity diagram* mulai transaksi penjualan dapat dilihat pada Gambar 19.



Gambar 19. Activity Diagram Mulai Transaksi.

2.14. Activity Diagram Admin Penjualan Akhiri Transaksi

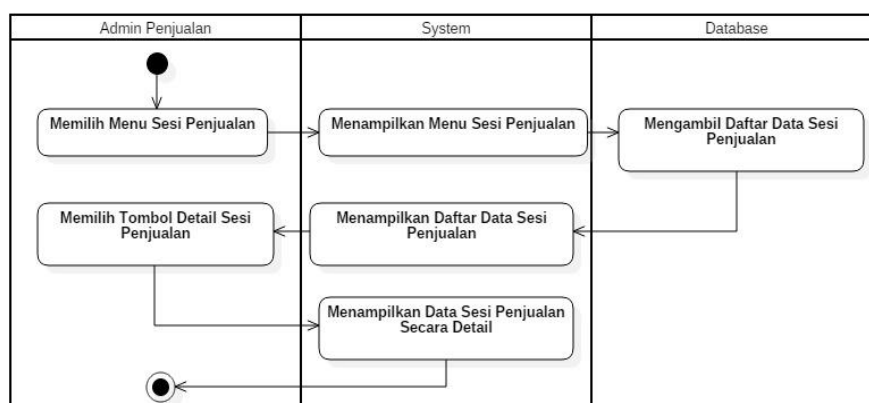
Pada *activity diagram* ini menjelaskan bagaimana proses mengakhiri transaksi penjualan. Ketika menu sesi penjualan sudah diakhiri oleh Admin Penjualan, maka menu *point of sales* tidak dapat diakses. *Activity diagram* akhiri transaksi dapat dilihat pada Gambar 20.



Gambar 20. Activity Diagram Akhiri Transaksi.

2.15. Activity Diagram Admin Penjualan Melihat Detail Sesi Penjualan

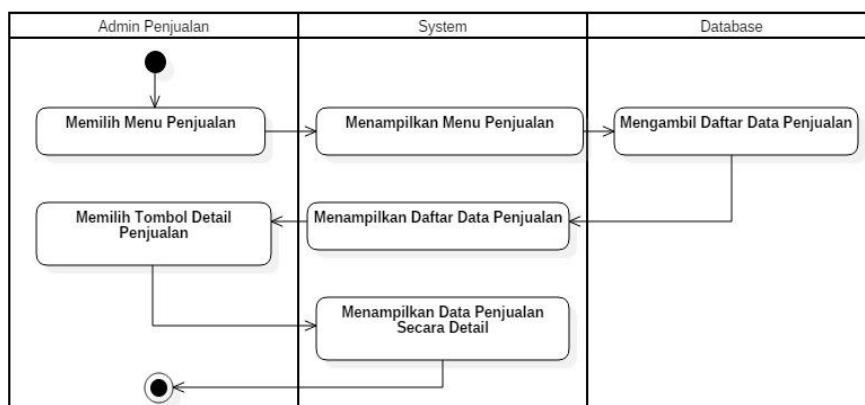
Pada *activity diagram* ini admin penjualan dapat melihat detail data sesi penjualan secara detail yang tersimpan pada *database*. *Activity diagram* dapat dilihat pada Gambar 21.



Gambar 21. Activity Diagram Melihat Detail Sesi Penjualan.

2.16. Activity Diagram Admin Penjualan Melihat Detail Penjualan

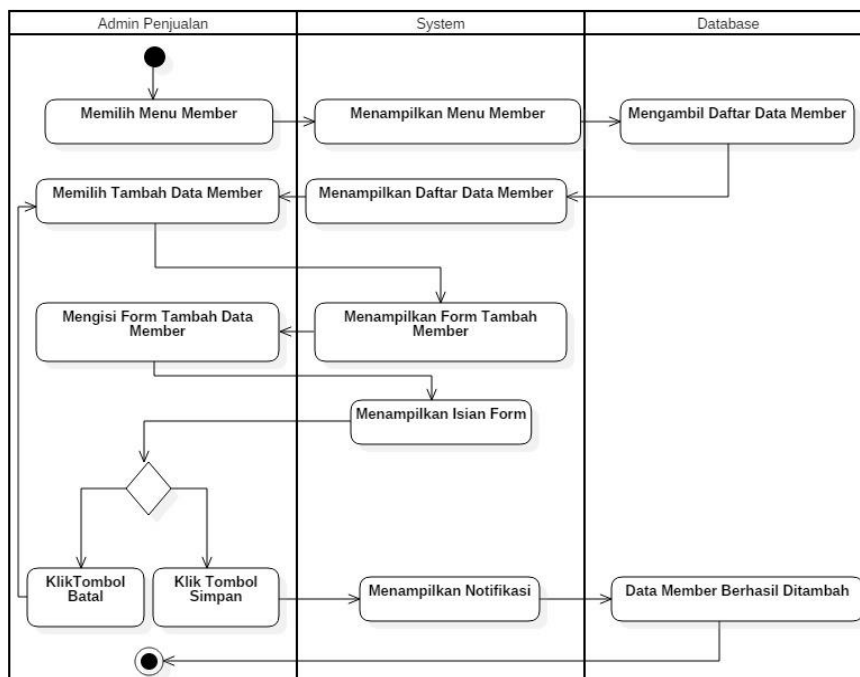
Pada *activity diagram* ini admin penjualan dapat melihat detail data penjualan secara detail dalam bentuk tabel. *Activity diagram* dapat dilihat pada Gambar 22.



Gambar 22. Activity Diagram Melihat Detail Penjualan.

2.17. Activity Diagram Admin Penjualan Menambah Member

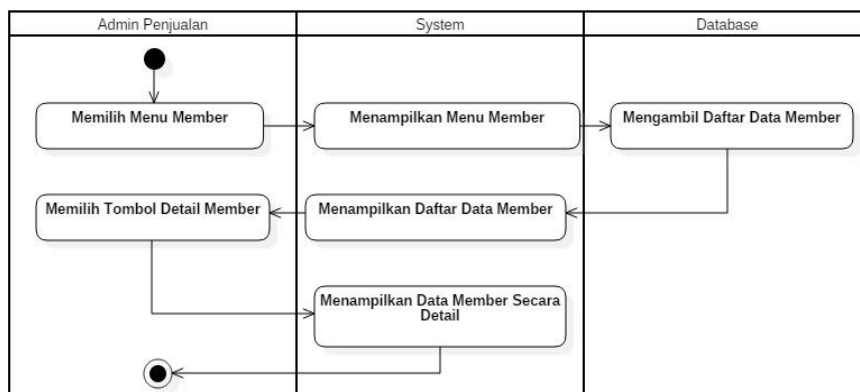
Pada *activity diagram* menambah data *member* menjelaskan proses penambahan data *member*. Admin Penjualan melakukan penambahan data *member* dengan mengisi *form* yang telah disediakan. Ketika semua data sudah terisi maka data tervalidasi dan data tersimpan ke *database*. *Activity diagram* menambah data *member* dapat dilihat pada Gambar 23.



Gambar 23. *Activity Diagram Menambah Member.*

2.18. Activity Diagram Admin Penjualan Melihat Detail Member

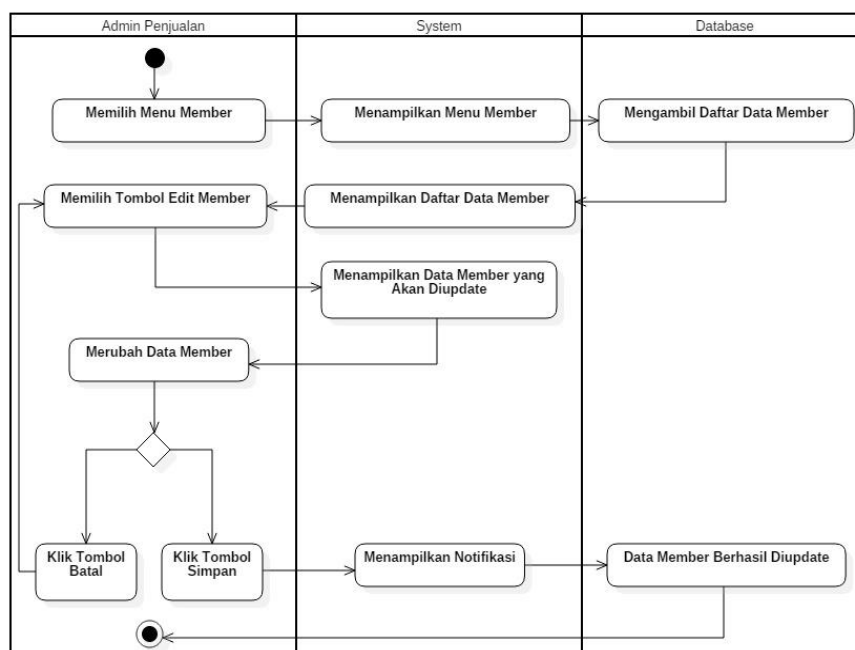
Pada *activity diagram* ini admin penjualan dapat melihat data *member* secara detail yang tersimpan pada *database*. *Activity diagram* dapat dilihat pada Gambar 24.



Gambar 24. *Activity Diagram Melihat Detail Member.*

2.19. Activity Diagram Admin Penjualan Mengedit Data Member

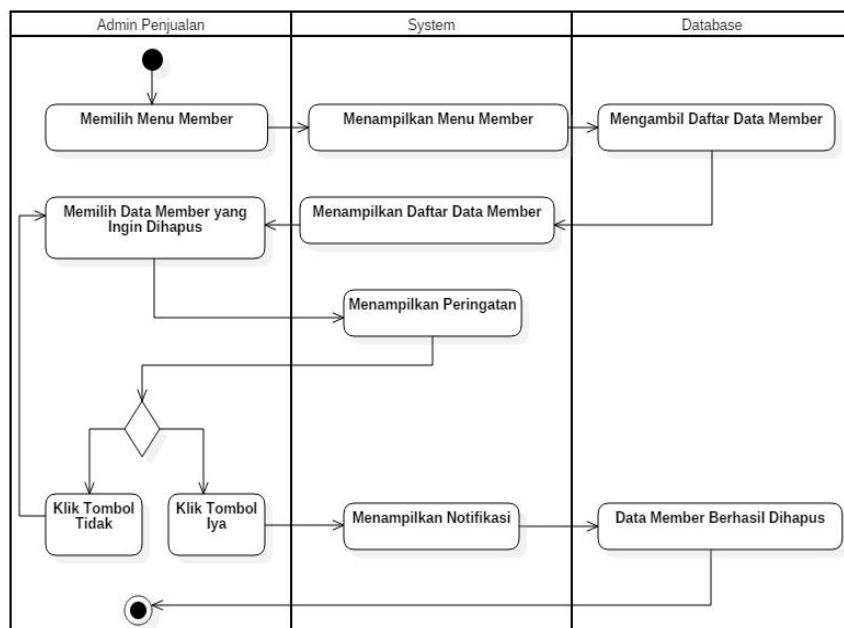
Pada *activity diagram* ini admin penjualan dapat mengubah data *member* yang tersimpan pada *database*. *Activity diagram* dapat dilihat pada Gambar 25.



Gambar 25. Activity Diagram Mengedit Data Member.

2.20. Activity Diagram Admin Penjualan Menghapus Data Member

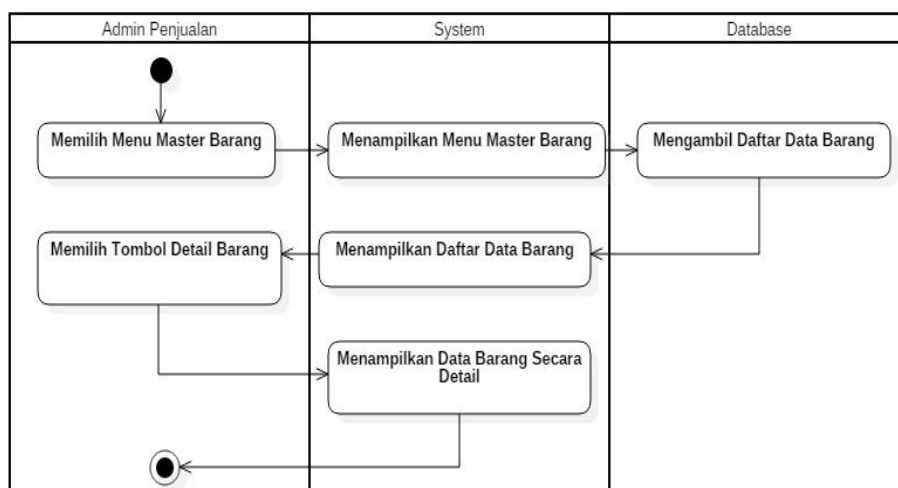
Pada *activity diagram* ini admin penjualan dapat menghapus data *member* yang tersimpan pada *database*. *Activity diagram* dapat dilihat pada Gambar 26.



Gambar 26. Activity Diagram Menghapus Data Member.

2.21. Activity Diagram Admin Penjualan Melihat Detail Master Barang

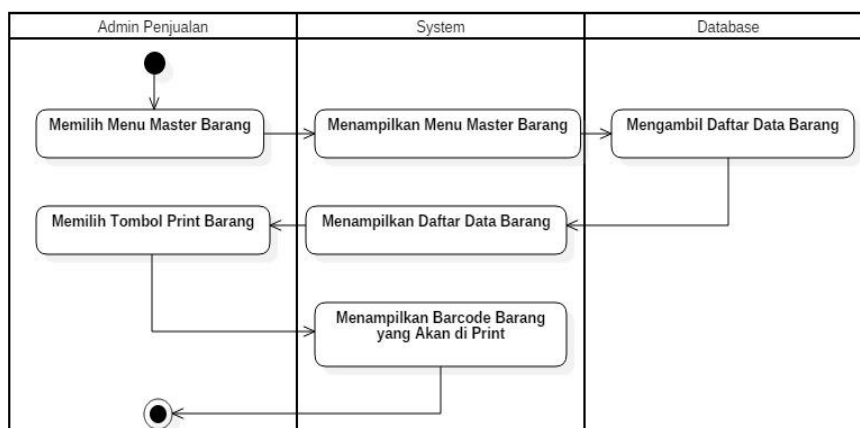
Pada *activity diagram* ini admin penjualan dapat melihat data barang secara detail yang tersimpan pada *database*. *Activity diagram* dapat dilihat pada Gambar 27.



Gambar 27. Activity Diagram Melihat Detail Master Barang.

2.22. Activity Diagram Admin Penjualan Mencetak Barcode Barang

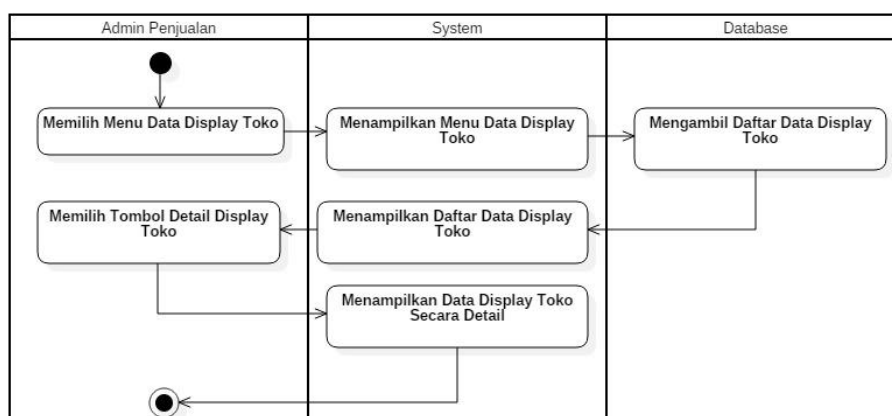
Activity diagram ini menjelaskan bagaimana admin penjualan melakukan *print barcode*. Pada Gambar 28 menunjukkan activity diagram *print barcode* barang.



Gambar 28. Activity Diagram Mencetak Barcode Barang.

2.23. Activity Diagram Admin Penjualan Melihat Display Toko

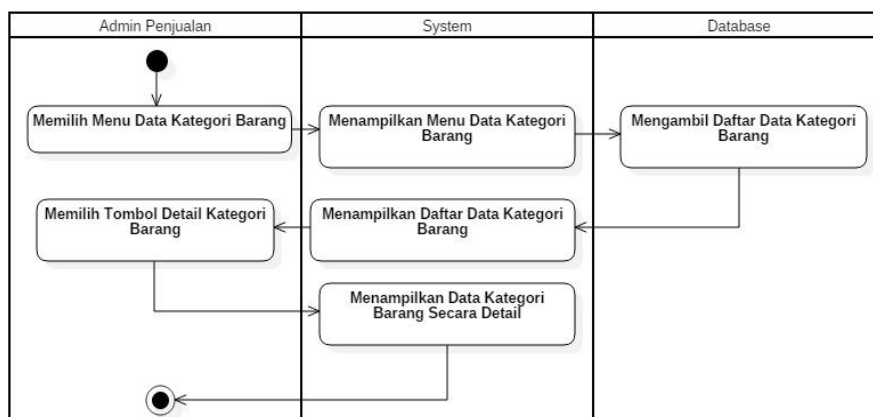
Pada activity diagram ini admin penjualan dapat melihat data *display* toko secara detail yang tersimpan pada *database*. Activity diagram dapat dilihat pada Gambar 29.



Gambar 29. Activity Diagram Melihat Display Toko.

2.24. Activity Diagram Admin Penjualan Melihat Detail Kategori

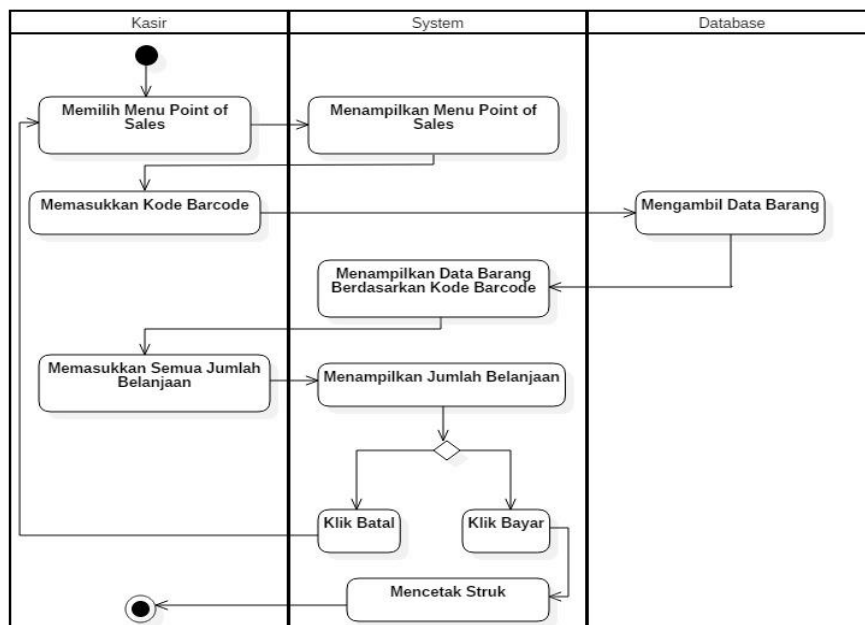
Pada *activity diagram* ini admin penjualan dapat melihat data kategori barang secara detail yang tersimpan pada *database*. *Activity diagram* dapat dilihat pada Gambar 30.



Gambar 30. Activity Diagram Melihat Detail Kategori Barang.

2.25. Activity Diagram Kasir Mengelola Point of Sales

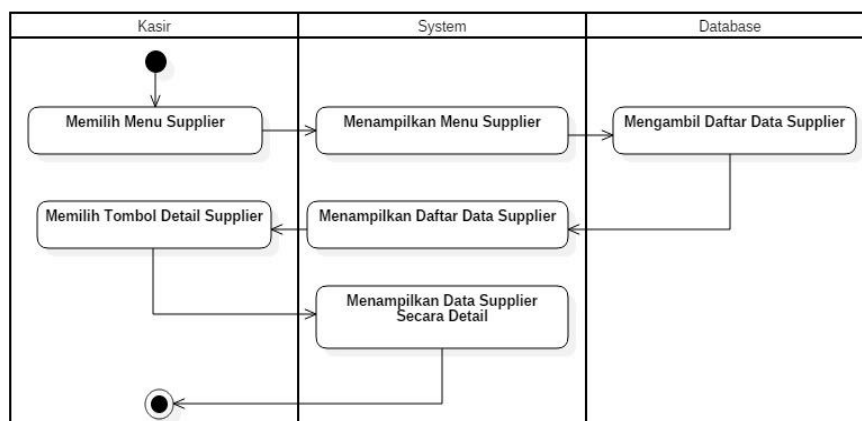
Pada *activity diagram* ini kasir dapat menambahkan transaksi penjualan dengan mengakses menu *point of sales*. Kasir memasukkan kode *barcode* barang, setelah muncul data barang yang diinputkan kasir memasukkan jumlah barang, diskon, kode *member*, dan jumlah pembayaran. Jika semua data sudah sesuai maka kasir akan mencetak struk belanjaan. *Activity diagram* mengelola *point of sales* dapat dilihat pada Gambar 31.



Gambar 31. Activity Diagram Kasir Mengelola Point of Sales.

2.26. Activity Diagram Kasir Melihat Detail Supplier

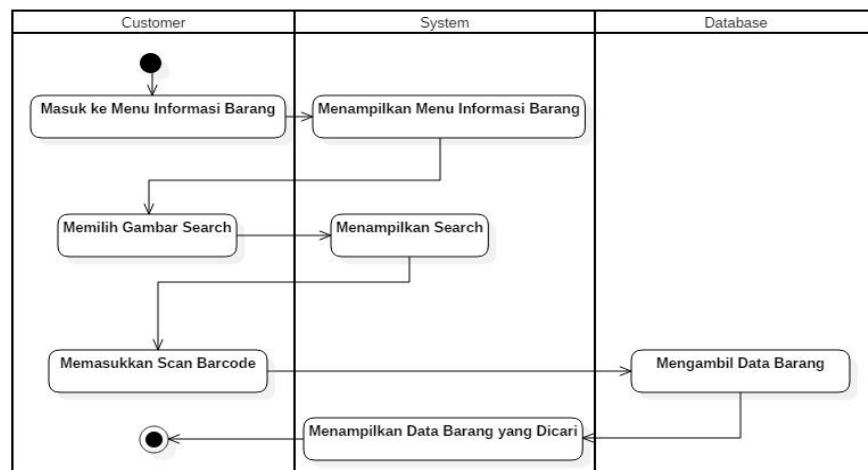
Pada *activity diagram* ini admin penjualan dapat melihat data *supplier* secara detail yang tersimpan pada *database*. Activity *diagram* dapat dilihat pada Gambar 32.



Gambar 32. Activity Diagram Kasir Melihat Detail Supplier.

2.27. Activity Diagram Customer Mencari Informasi Barang

Pada *activity diagram* ini *customer* dapat mencari jumlah stok barang yang masih tersedia. Tidak hanya itu *customer* juga dapat melihat barang secara detail. *Activity diagram* mencari informasi barang dapat dilihat pada Gambar 33.



Gambar 33. Activity Diagram Mencari Informasi Barang.

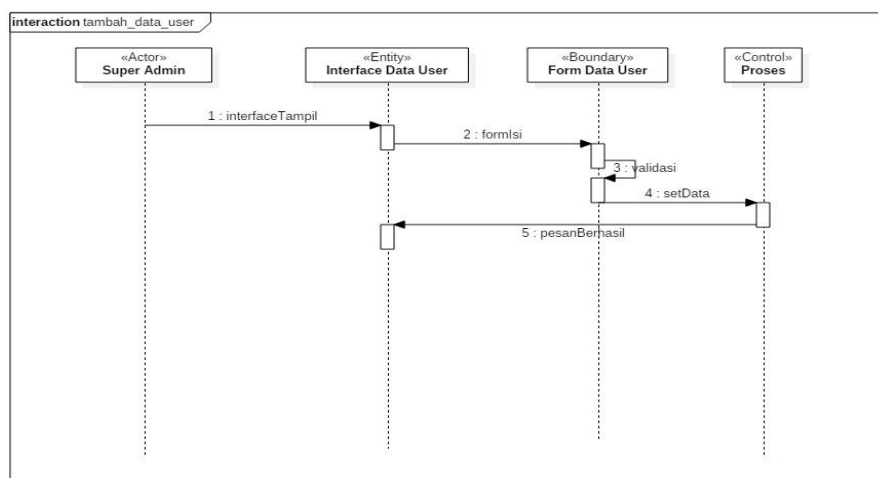
3. Sequence diagram

Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, *display*, dan sebagainya) yang berupa *message* yang digambarkan terhadap waktu.

3.1 Sequence Diagram Menambah User

Sequence diagram menambah *user* pada Gambar 34 dilakukan oleh aktor *Super Admin*. Setelah semua *form* selesai diisi aktor menekan tombol simpan dan sistem akan melakukan validasi dari *form* yang telah diisi. Jika tidak ada kesalahan maka sistem akan memproses data

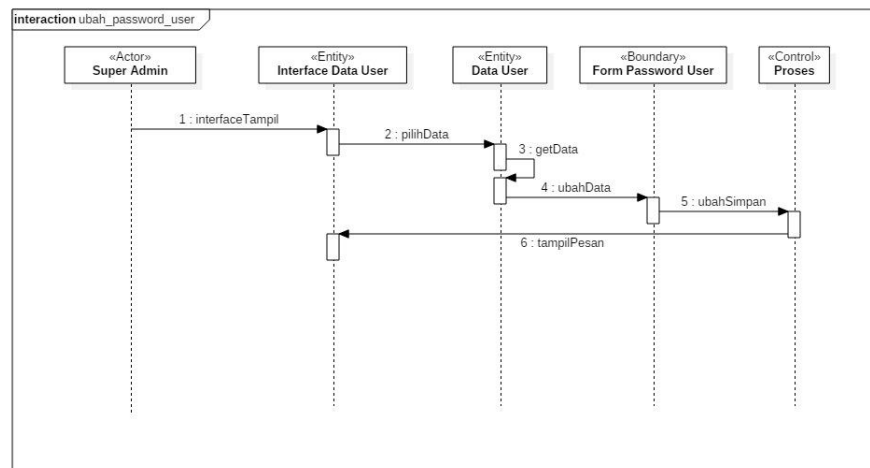
untuk disimpan ke *database*. *Sequence diagram* dapat dilihat pada Gambar 34.



Gambar 34. *Sequence Diagram Menambah User.*

3.2 *Sequence Diagram Mengedit Password User*

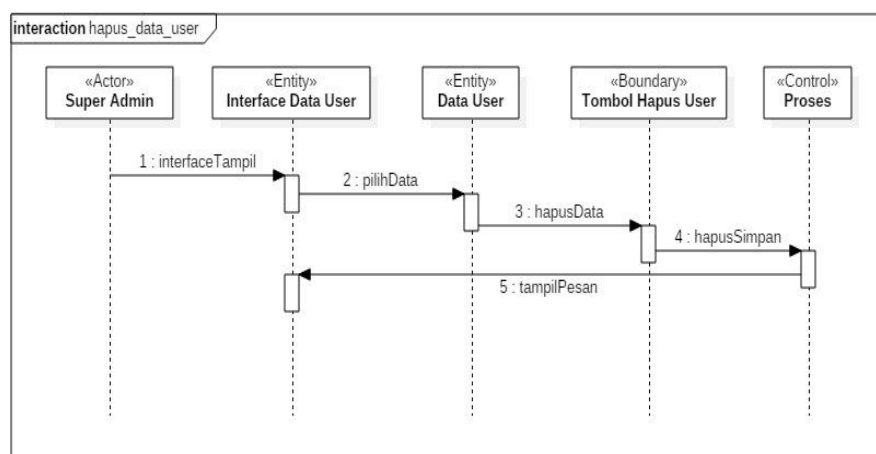
Sequence diagram edit *password user* pada Gambar 35 dilakukan oleh aktor *Super Admin*. *Sequence* ini menunjukkan aktor memulai proses dengan membuka *interface* kontrol pengguna, lalu aktor memilih edit *password user*. Kemudian aktor mengisi *form* edit *password user* yang akan diubah. Sistem akan melakukan proses penyimpanan dan membuka *interface* kontrol pengguna kembali. Gambar 35 merupakan *sequence diagram* dari edit *password user*.



Gambar 35. *Sequence Diagram Mengedit Password User.*

3.3 Sequence Diagram Menghapus User

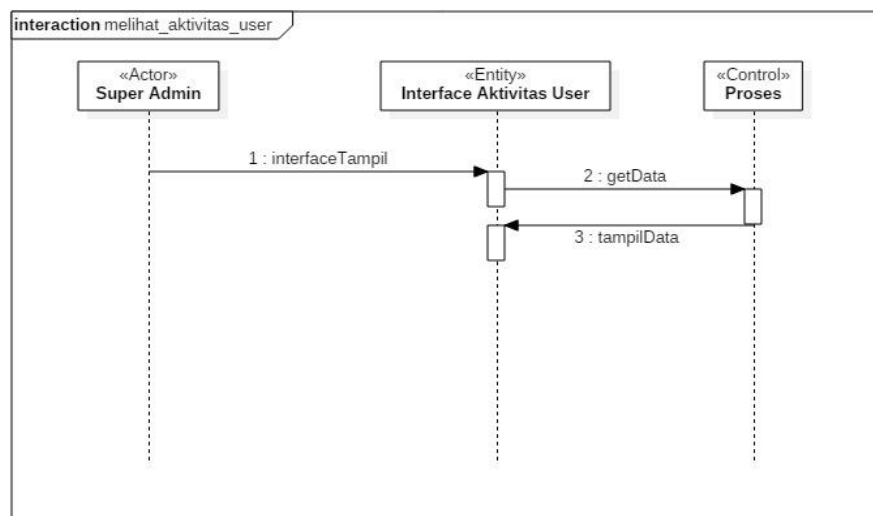
Sequence diagram menghapus *user* pada Gambar 36 dilakukan oleh aktor Super Admin. *Sequence* ini menunjukkan aktor memulai proses dengan membuka *interface* kontrol pengguna. Sistem akan mengambil data *user* dan akan ditampilkan pada *interface* kontrol pengguna. Gambar 36 menunjukkan *sequence diagram* hapus *user*.



Gambar 36. *Sequence Diagram Menghapus User.*

3.4 Sequence Diagram Aktivitas User

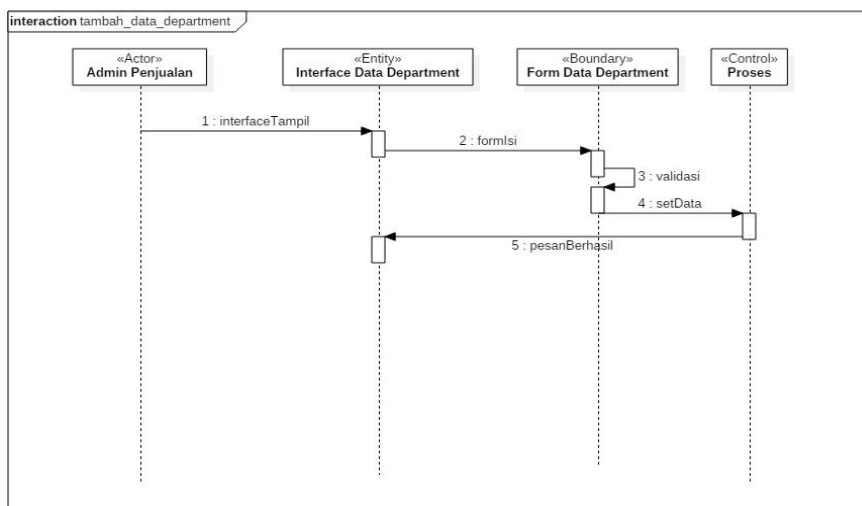
Sequence diagram melihat aktivitas *user* pada Gambar 37 dilakukan oleh aktor *Super Admin*. *Sequence* ini menunjukkan aktor memulai proses dengan membuka *interface* aktivitas *user*. Sistem akan mengambil data aktivitas *user* dan akan ditampilkan pada *interface* aktivitas *user*. Gambar 37 adalah *sequence diagram* aktivitas *user*.



Gambar 37. *Sequence Diagram* Aktivitas *User*.

3.5 Sequence Diagram Menambah Data Department

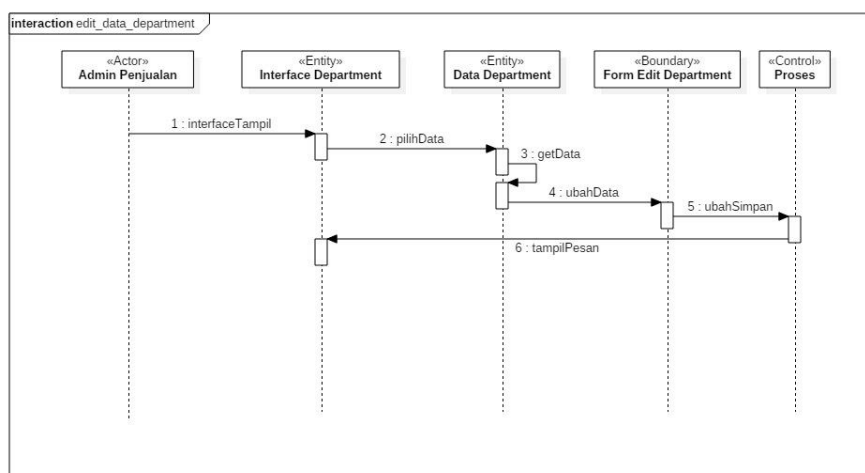
Sequence diagram menambah data *department* dilakukan oleh Admin Penjualan. Setelah pengisian *form* selesai aktor menekan tombol simpan dan sistem akan melakukan validasi dari *form* yang telah diisi. Jika tidak ada kesalahan maka sistem akan memproses data untuk disimpan ke *database*. *Sequence diagram* dapat dilihat pada Gambar 38.



Gambar 38. *Sequence Diagram Menambah Data Department.*

3.6 Sequence Diagram Mengedit Data Department

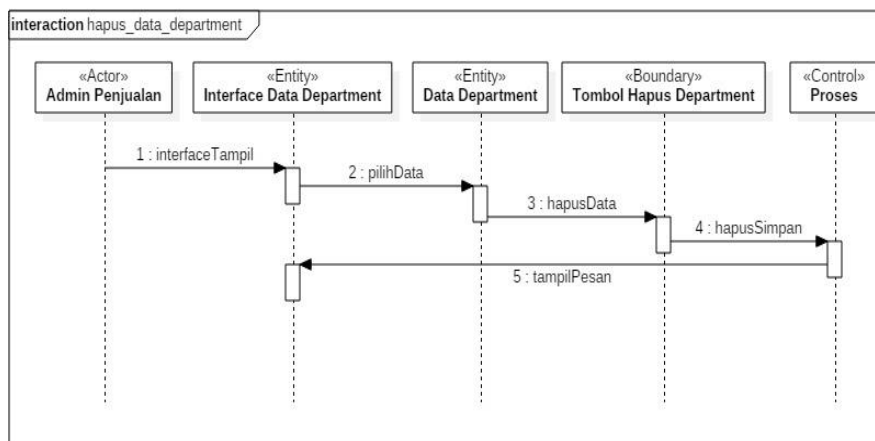
Sequence diagram edit data *department* pada Gambar 39 dilakukan oleh aktor Admin Penjualan. Setelah mengisi semua *form* maka sistem akan melakukan proses penyimpanan dan membuka *interface* data *department* kembali. Gambar 39 merupakan *sequence diagram* dari edit data *department*.



Gambar 39. *Sequence Diagram Mengedit Data Department.*

3.7 Sequence Diagram Menghapus Data Department

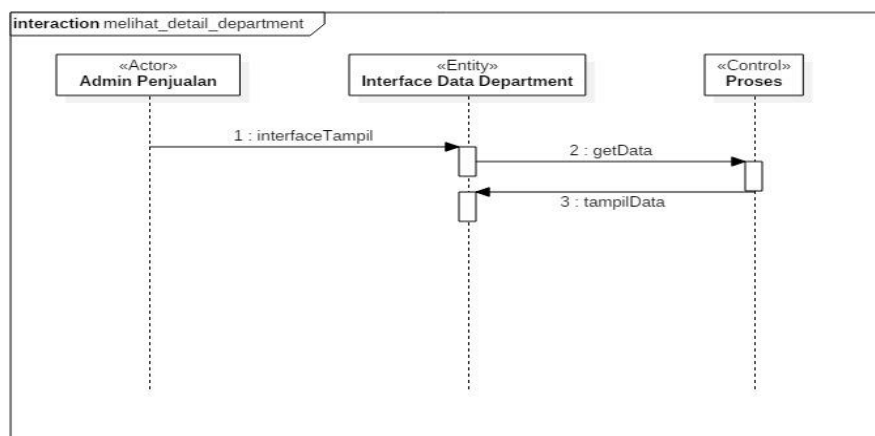
Sequence Diagram menghapus data *department* pada Gambar 40 dilakukan oleh aktor Admin Penjualan. Sistem akan mengambil data *department* dan akan ditampilkan pada *interface data department*. Gambar 40 menunjukkan *sequence diagram* hapus data *department*.



Gambar 40. *Sequence Diagram* Menghapus Data Department.

3.8 Sequence Diagram Melihat Data Department

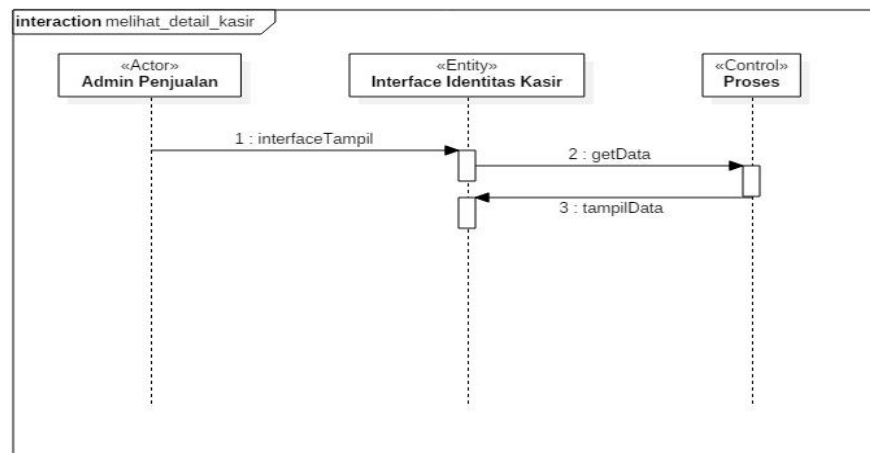
Sequence diagram ini dilakukan oleh aktor Admin Penjualan. Sistem akan mengambil data *department* dan akan ditampilkan. Gambar 41 adalah *sequence diagram* melihat data *department*.



Gambar 41. *Sequence Diagram* Melihat Data Department.

3.9 Sequence Diagram Melihat Detail Kasir

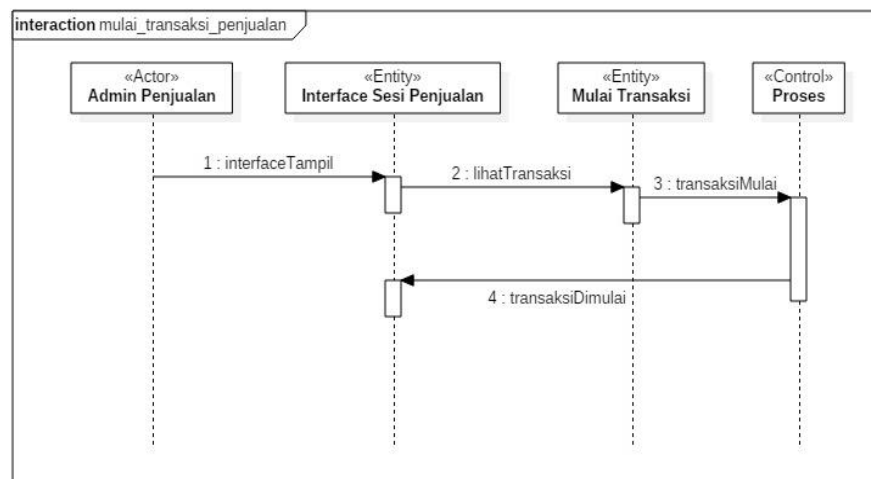
Sequence diagram melihat detail kasir akan mengambil data kasir dan akan ditampilkan pada *interface* identitas detail kasir. Gambar 42 menunjukkan *sequence diagram* lihat detail kasir.



Gambar 42. *Sequence Diagram* Melihat Detail Kasir.

3.10 Sequence Diagram Mulai Transaksi Penjualan

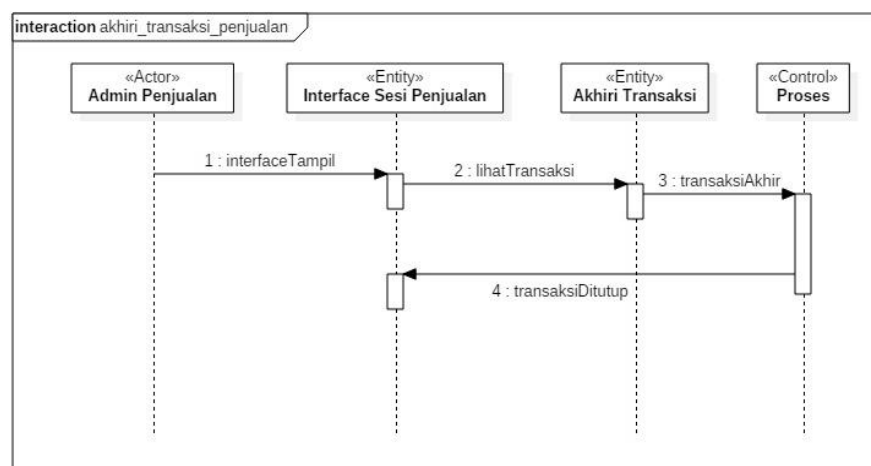
Sequence diagram mulai transaksi penjualan pada Gambar 43 dilakukan oleh aktor admin penjualan. *Sequence* ini menunjukkan aktor memulai proses dengan membuka *interface* sesi penjualan. Kemudian aktor akan memilih transaksi yang akan dimulai. Setelah *interface* sesi penjualan telah dibuka maka aktor memilih mulai transaksi dan akan diproses oleh sistem. *Sequence diagram* dapat dilihat pada Gambar 43.



Gambar 43. *Sequence Diagram* Mulai Transaksi Penjualan.

3.11 *Sequence Diagram* Akhiri Transaksi Penjualan

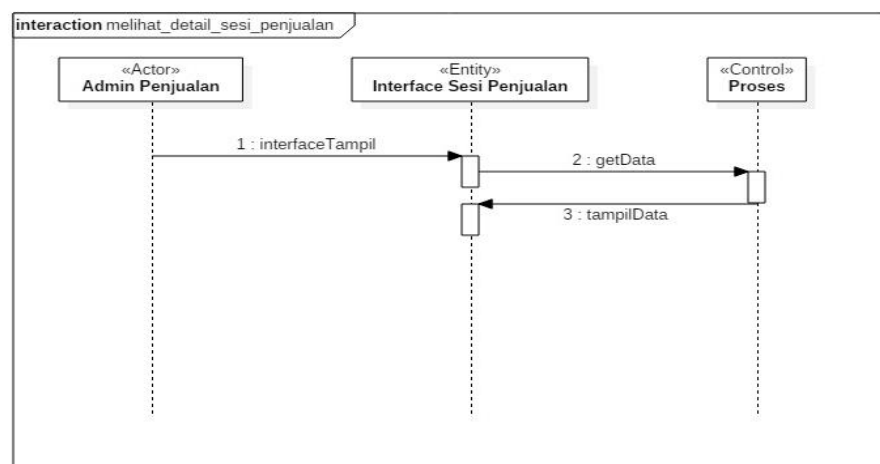
Sequence diagram akhiri transaksi penjualan pada Gambar 44 dilakukan oleh aktor admin penjualan. *Sequence* ini menunjukkan aktor memulai proses dengan membuka *interface* sesi penjualan. Kemudian aktor akan memilih transaksi yang akan diakhiri. Setelah *interface* sesi penjualan telah dibuka maka aktor memilih akhiri transaksi dan akan diproses oleh sistem. *Sequence diagram* dapat dilihat pada Gambar 44.



Gambar 44. *Sequence Diagram* Akhiri Transaksi Penjualan.

3.12 Sequence Diagram Melihat Detail Sesi Penjualan

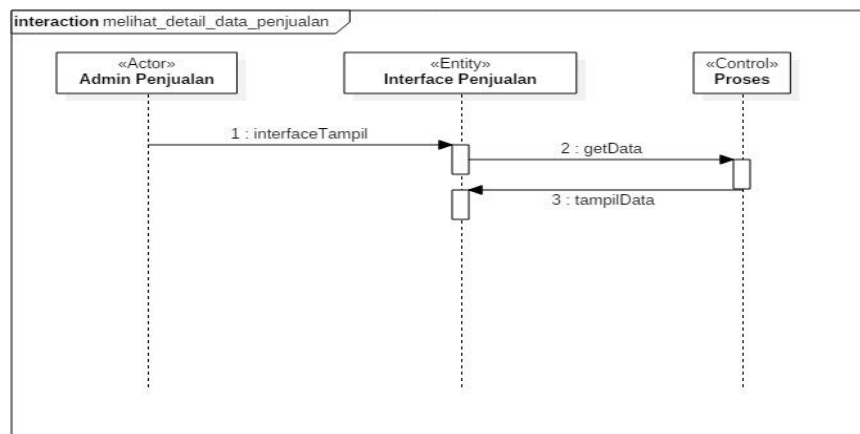
Sequence diagram melihat detail sesi penjualan pada Gambar 45 dilakukan oleh aktor Admin Penjualan. *Sequence* ini menunjukkan aktor memulai proses dengan membuka *interface* sesi penjualan. Sistem akan mengambil data detail sesi penjualan dan akan ditampilkan pada *interface* detail sesi penjualan. Gambar 45 menunjukkan *sequence diagram* lihat detail sesi penjualan.



Gambar 45. *Sequence Diagram* Melihat Detail Sesi Penjualan.

3.13 Sequence Diagram Melihat Detail Data Penjualan

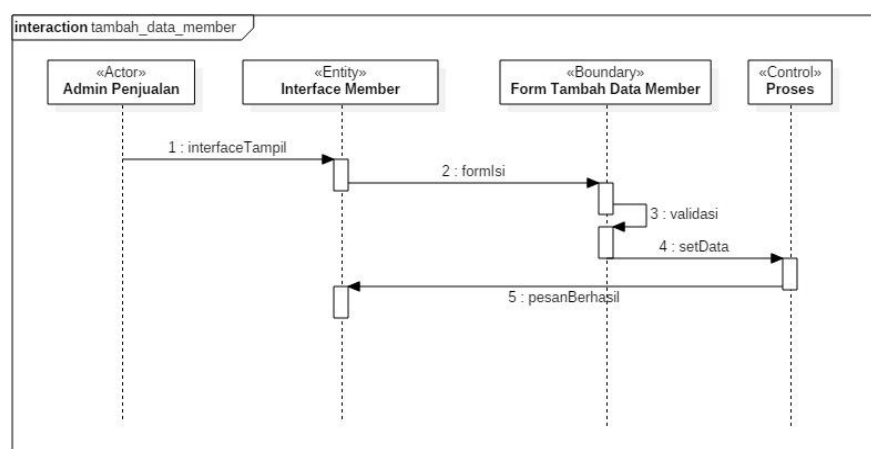
Sequence diagram ini dilakukan oleh aktor Admin Penjualan. Sistem akan mengambil data penjualan untuk ditampilkan. Gambar 46 menunjukkan *sequence diagram* detail data penjualan.



Gambar 46. *Sequence Diagram* Melihat Detail Data Penjualan.

3.14 *Sequence Diagram* Menambah Data Member

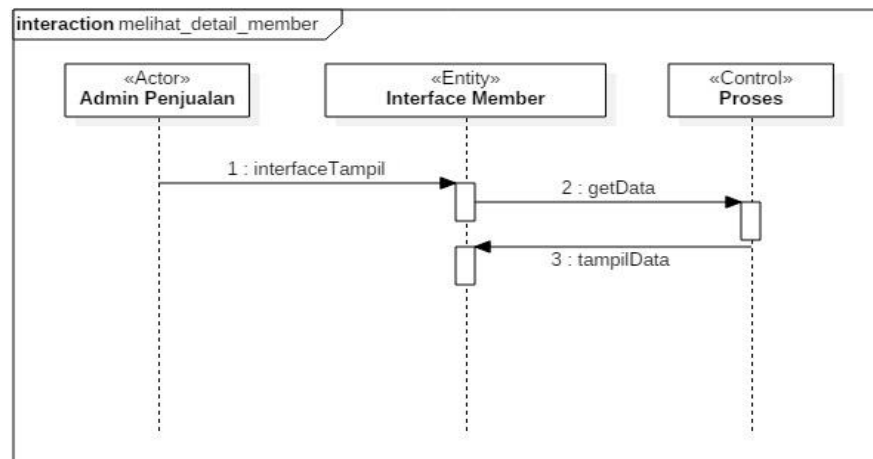
Sequence diagram menambah data *member* dilakukan oleh aktor Admin Penjualan. *Sequence diagram* menambah data *member* dapat dilihat pada Gambar 47.



Gambar 47. *Sequence Diagram* Menambah Data Member.

3.15 *Sequence Diagram* Melihat Data Member

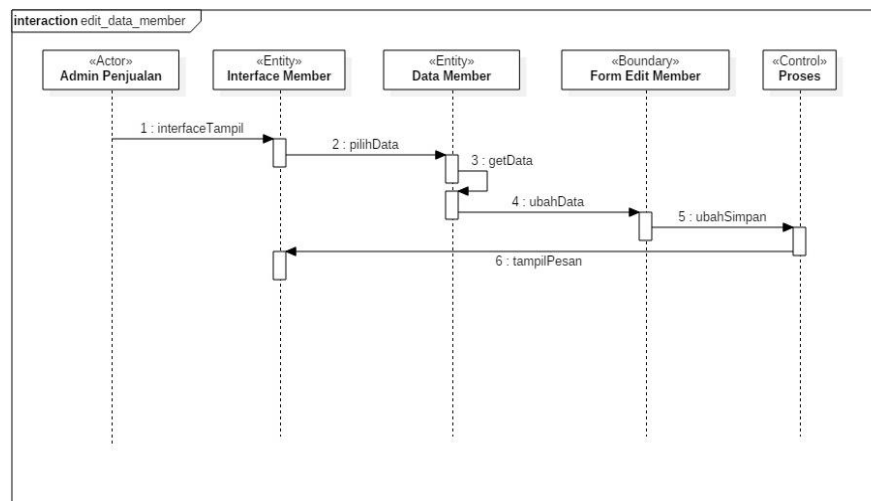
Sequence diagram melihat data member pada Gambar 48 dilakukan oleh aktor Admin Penjualan. Sistem akan mengambil data member dan akan ditampilkan pada *interface* data member. *Sequence diagram* melihat data detail *member* dapat dilihat pada Gambar 48.



Gambar 48. *Sequence Diagram Melihat Data Member.*

3.16 *Sequence Diagram Mengedit Data Member*

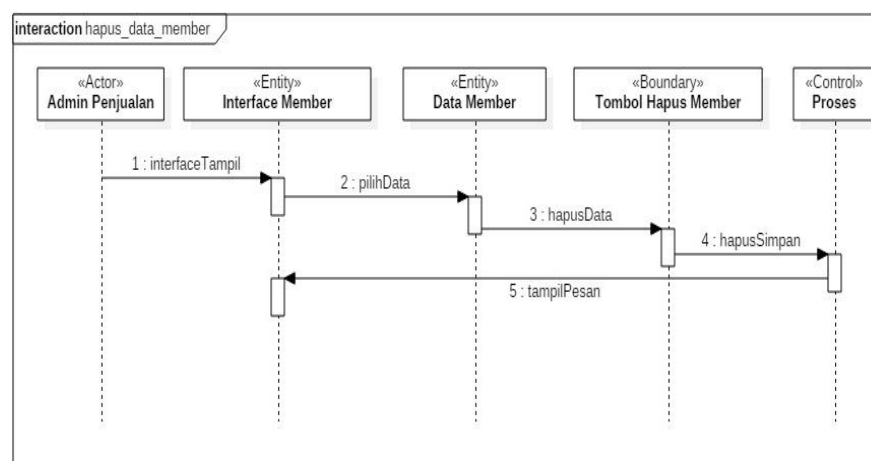
Sequence diagram mengedit data *member* pada Gambar 49 dilakukan oleh aktor Admin Penjualan. *Sequence* ini menunjukkan aktor memulai proses dengan membuka *interface* data *member*. Aktor lalu memilih data *member* yang akan diubah dan sistem akan menampilkan data *member* yang dipilih. Kemudian aktor mengisi *form* dari data *member* yang akan diubah dan menyimpannya. Sistem akan melakukan proses penyimpanan dan membuka *interface member* kembali. Gambar 49 merupakan *sequence diagram* dari edit data *member*.



Gambar 49. *Sequence Diagram Mengedit Data Member.*

3.17 Sequence Diagram Menghapus Member

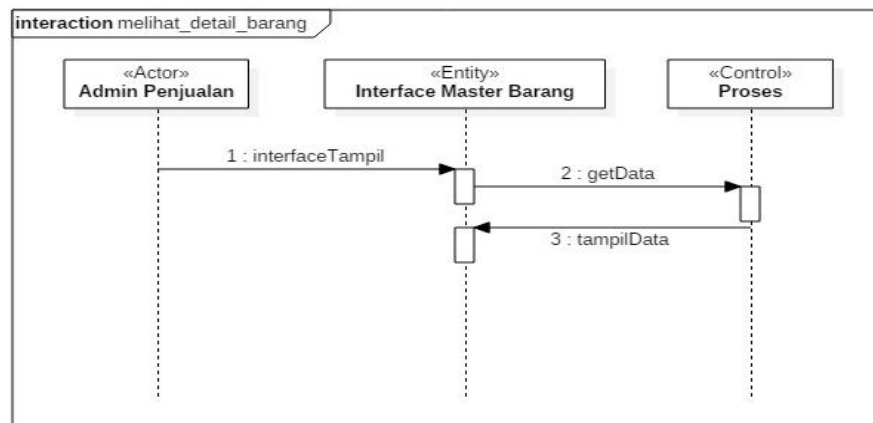
Sequence diagram menghapus member pada Gambar 50 dilakukan oleh aktor Admin Penjualan. *Sequence* ini menunjukkan aktor memulai proses dengan membuka *interface member*. Sistem akan mengambil data *member* dan akan ditampilkan pada *interface member*. Gambar 50 menunjukkan *sequence diagram* hapus member.



Gambar 50. *Sequence Diagram Manghapus Member.*

3.18 Sequence Diagram Melihat Detail Barang

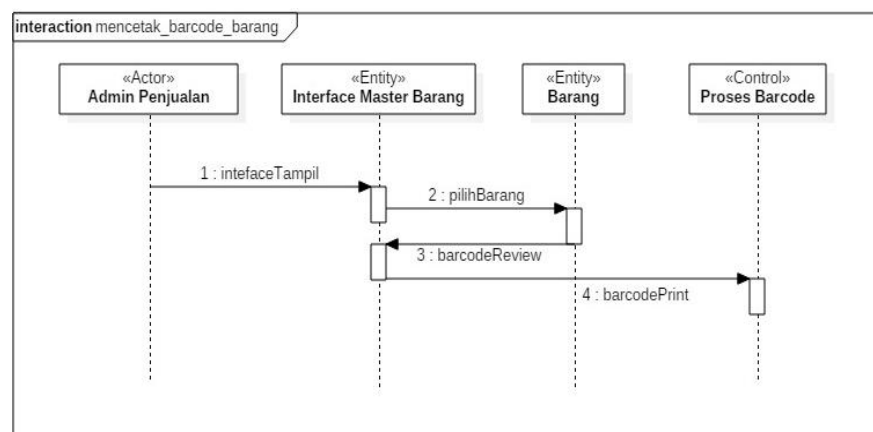
Sequence diagram ini dilakukan oleh aktor Admin Penjualan. *Sequence* ini menunjukkan detail barang. Gambar 51 menunjukkan *sequence diagram* lihat detail barang.



Gambar 51. *Sequence Diagram* Melihat Detail Barang.

3.19 Sequence Diagram Mencetak Barcode Barang

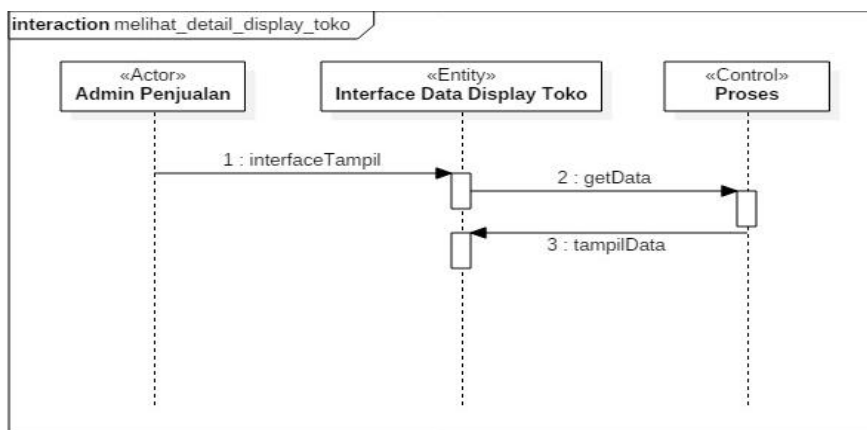
Sequence diagram ini dilakukan oleh Admin Penjualan. Aktor akan memilih barang yang akan dicetak label-nya dan akan menampilkan *review* dari *barcode*. Setelah melihat *review* aktor akan memproses pencetakan label. *Sequence diagram* dapat dilihat pada Gambar 52.



Gambar 52. *Sequence Diagram* Mencetak Barcode Barang.

3.20 Sequence Diagram Melihat Display Toko

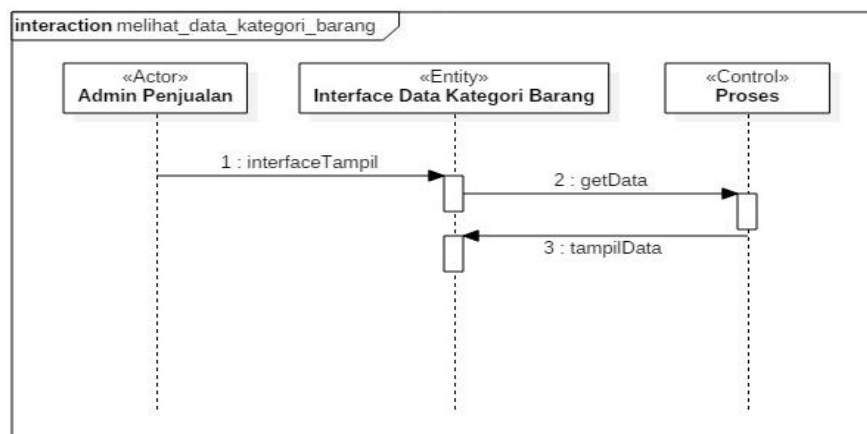
Sequence diagram ini dilakukan oleh Admin Penjualan. Sistem akan mengambil data *display* toko dan akan ditampilkan pada *interface display* toko. Gambar 53 menunjukkan *sequence diagram display* toko.



Gambar 53. *Sequence Diagram* Melihat Display Toko.

3.21 Sequence Diagram Melihat Kategori Barang

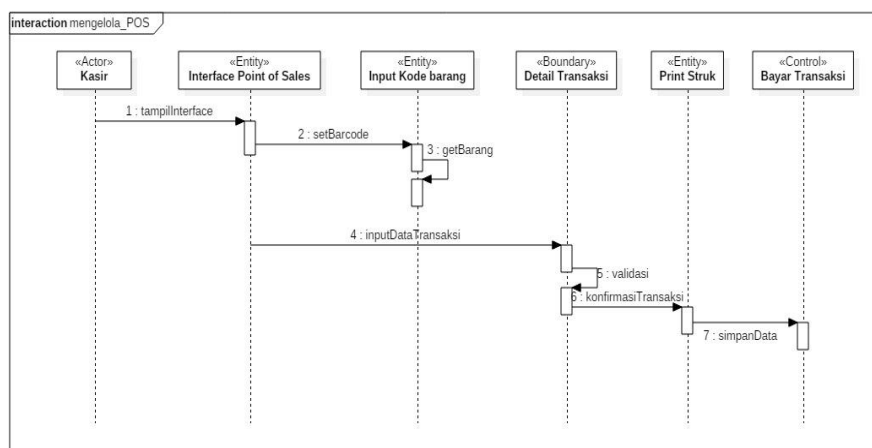
Sequence diagram ini dilakukan oleh aktor Admin Penjualan. Sistem akan mengambil data kategori barang dan akan menampilkannya. Gambar 54 menunjukkan *sequence diagram* lihat kategori barang.



Gambar 54. *Sequence Diagram* Melihat Kategori Barang.

3.22 Sequence Diagram Mengelola Menu Point of Sales

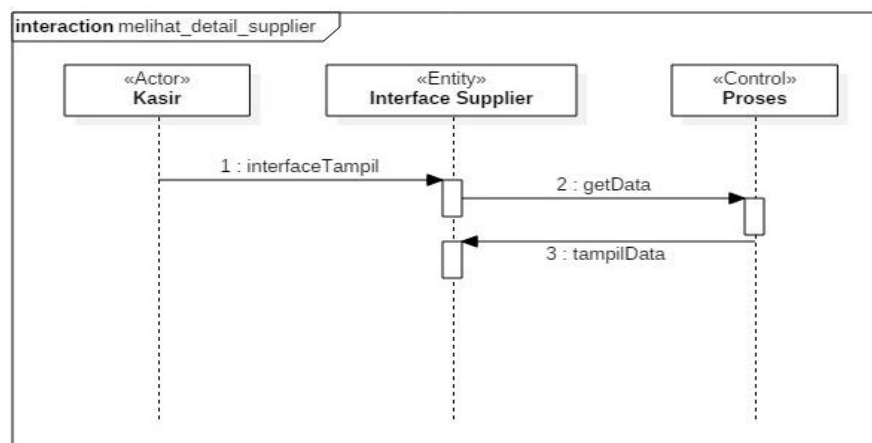
Sequence diagram ini dilakukan oleh Kasir. *Sequence* ini menunjukkan langkah-langkah mengelola *point of sales*. Gambar 55 adalah *sequence diagram* mengelola menu *point of sales*.



Gambar 55. *Sequence Diagram Mengelola Menu Point of Sales.*

3.23 Sequence Diagram Melihat Detail Supplier

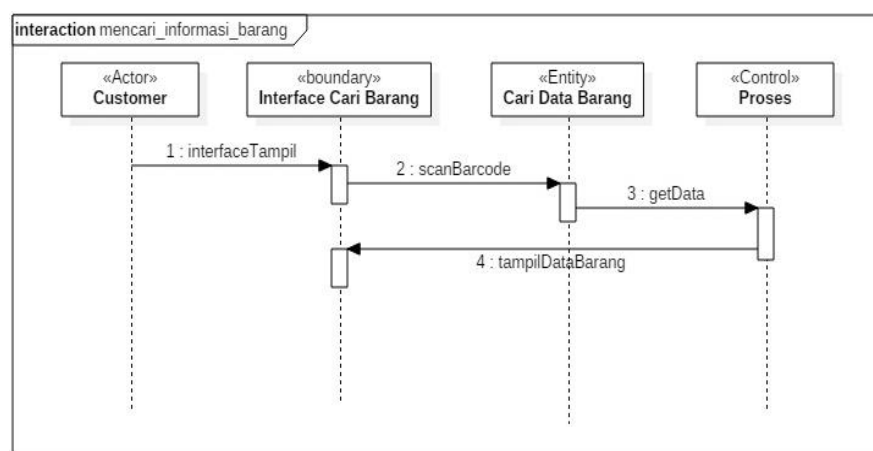
Sequence diagram ini dilakukan oleh Kasir. Sistem akan mengambil data *supplier* dan akan ditampilkan secara detail. Gambar 56 menunjukkan *sequence diagram* lihat detail *supplier*.



Gambar 56. *Sequence Diagram Melihat Detail Supplier.*

3.24 Sequence Diagram Mencari Informasi Barang

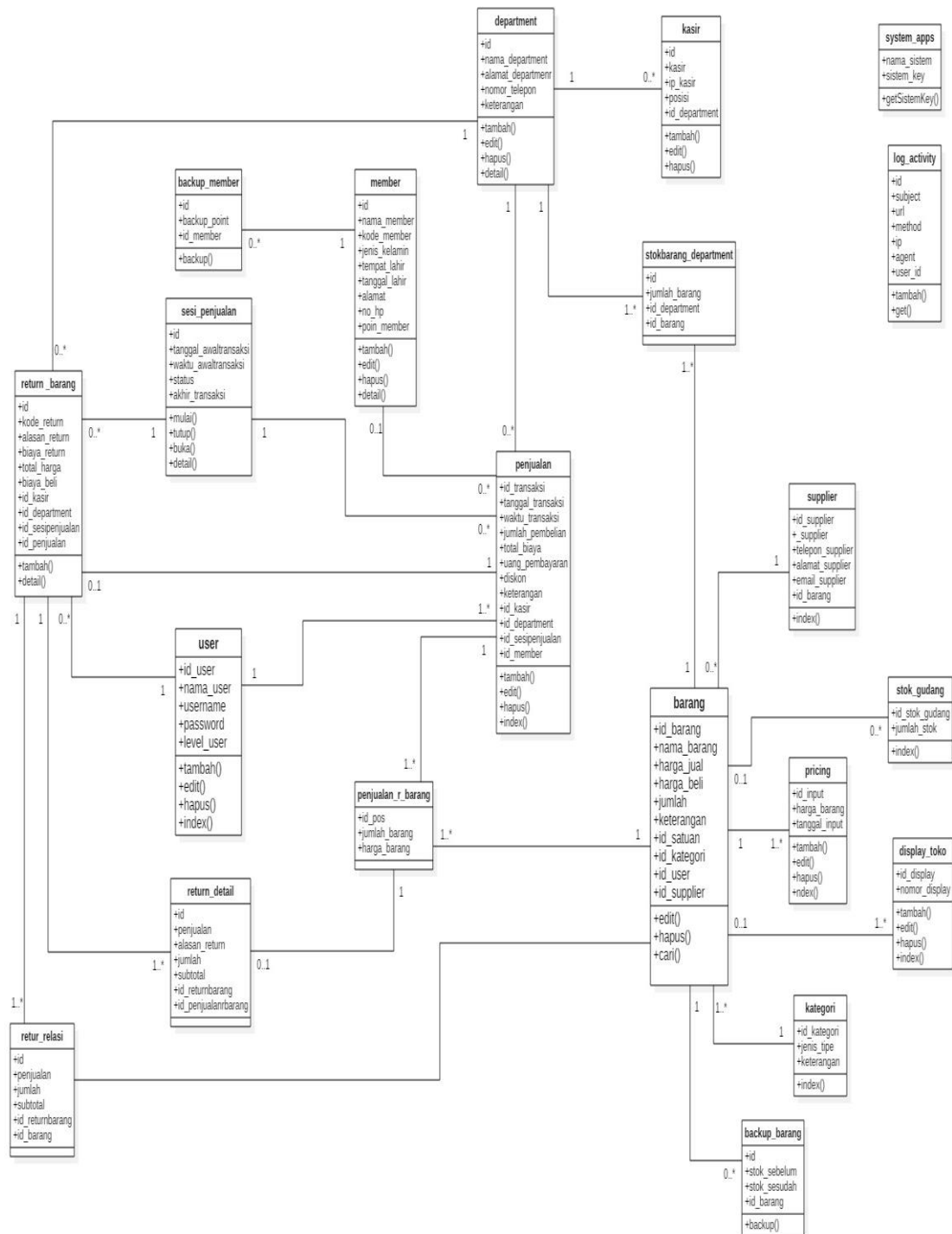
Sequence diagram mencari informasi barang dilakukan oleh *customer*. *Sequence* ini menunjukkan aktor memulai proses dengan membuka *interface* informasi barang. Kemudian aktor akan memasukkan kata kunci dan sistem akan memproses. Jika barang ditemukan maka sistem akan menampilkan data barang yang dicari. *Sequence diagram* dapat dilihat pada Gambar 57.



Gambar 57. Sequence Diagram Mencari Informasi Barang.

4. Class diagram

Class diagram merupakan sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. Pada Gambar 58 merupakan *class diagram* dari sistem *Customer Relationship Management (CRM)*.

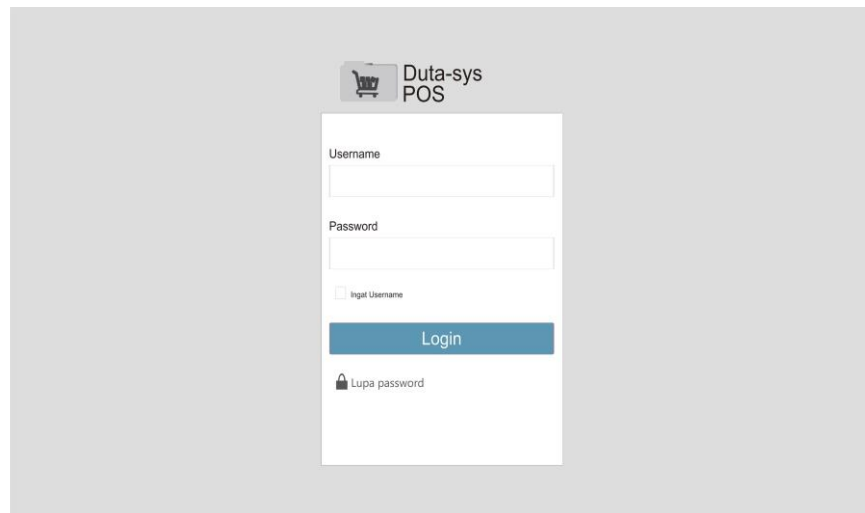


Gambar 58. *Class Diagram Customer Relationship Management.*

5. Desain antarmuka atau *interface*

5.1. Rancangan *Interface Login*

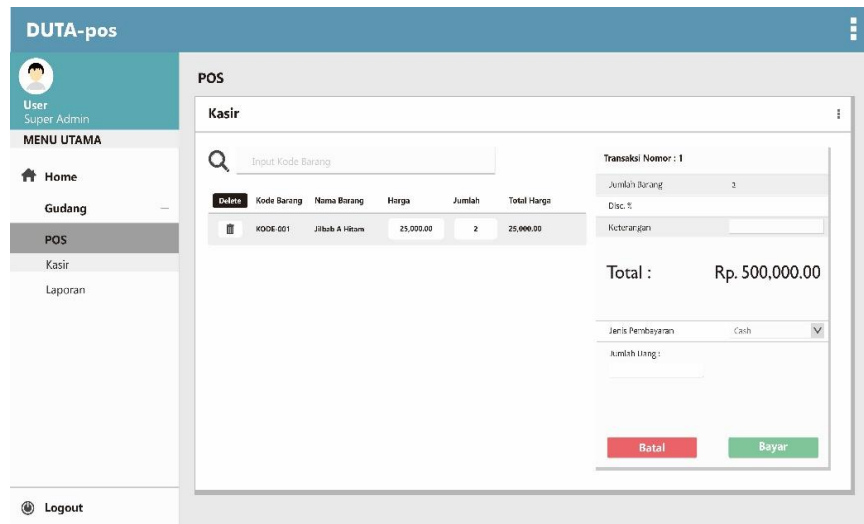
Interface ini merupakan *interface* login di mana *user* yang ingin menggunakan sistem harus masuk terlebih dahulu menggunakan akun yang dimiliki. Tampilan login dapat dilihat pada Gambar 59.



Gambar 59. Rancangan *Interface Login*.

5.2. Rancangan *Interface Point of Sales*

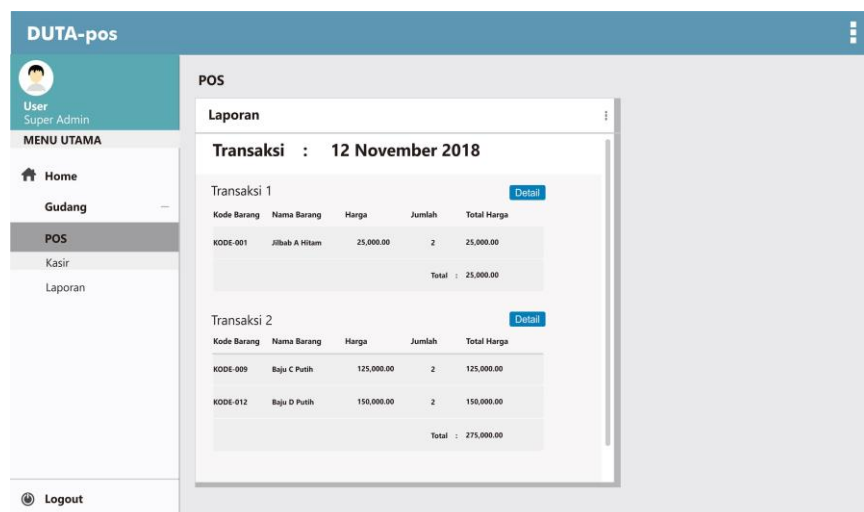
Interface ini merupakan tampilan dari sistem *point-of-sales* yang akan memproses transaksi yang terjadi pada pelanggan. Kasir dapat melakukan *input* kode barang pada kotak pencarian berdasarkan kode barang yang ada pada *barcode* setiap barang. Lalu kasir dapat menginputkan jumlah yang dibeli oleh *customer*, keterangan, dan juga jenis pembayaran yang dilakukan. Tampilan kasir dapat dilihat pada Gambar 60.



Gambar 60. Rancangan *Interface Point of Sales*.

5.3. Rancangan *Interface Detail Penjualan*

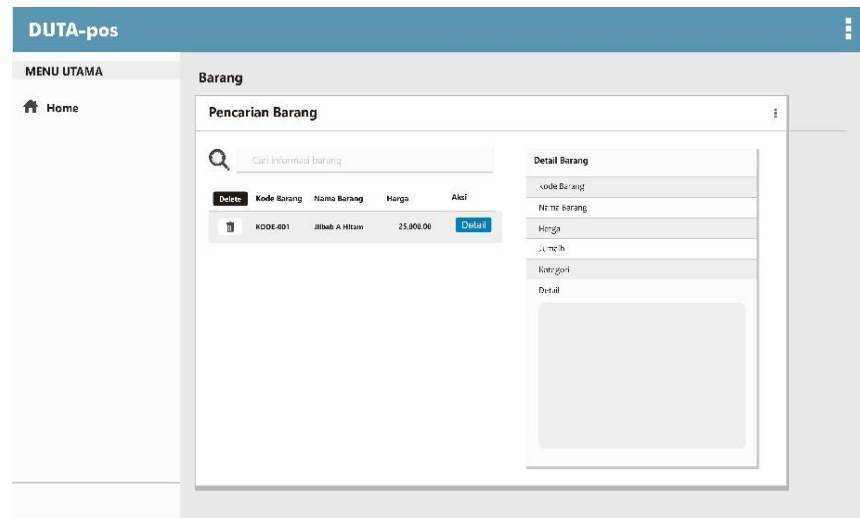
Pada *interface* ini menampilkan transaksi penjualan yang terjadi setiap harinya. Transaksi akan direkap per-hari ketika transaksi telah ditutup. Tampilan rekap transaksi dapat dilihat pada Gambar 61.



Gambar 61. Rancangan *Interface Detail Penjualan*.

5.4. Rancangan *Interface* Mencari Informasi Barang

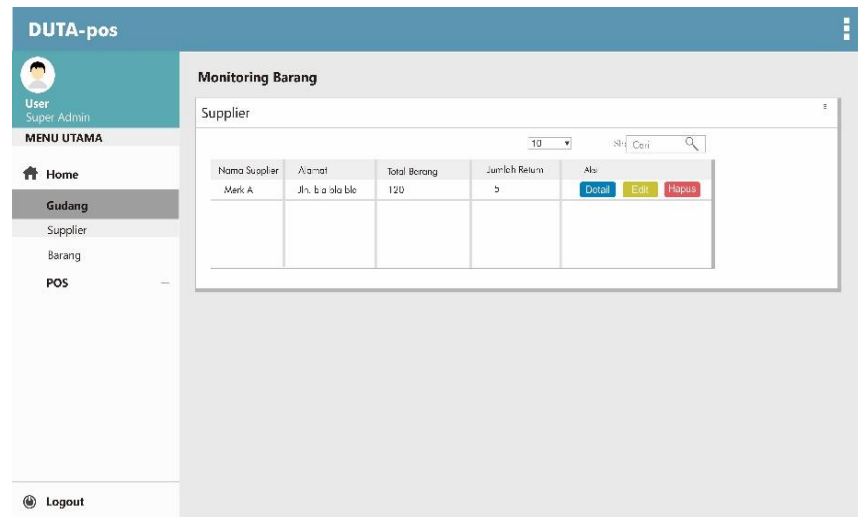
Pada *interface* ini menampilkan menu pencarian data barang untuk pelanggan. Di sini pelanggan dapat melakukan pencarian barang dan mendapat detail informasi pada barang yang dicari. Tampilan pencarian barang dapat dilihat pada Gambar 62.



Gambar 62. Rancangan *Interface* Mencari Informasi Barang.

5.5. Rancangan *Interface* Data Supplier

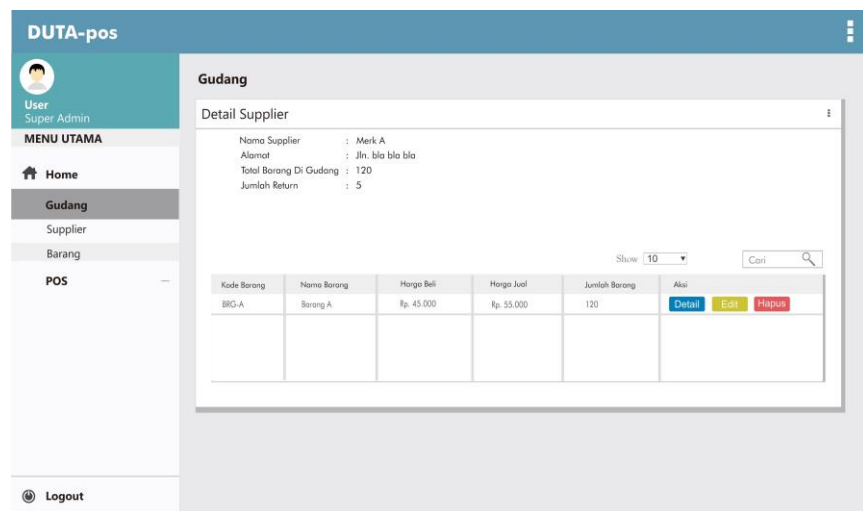
Pada *interface* ini menampilkan data-data *supplier* yang ada. Data *supplier* yang dilihat pada *interface* ini merupakan data yang dikirimkan oleh sistem SSG. Pada Gambar 63 menampilkan tampilan data *supplier*.



Gambar 63. Rancangan *Interface Data Supplier*.

5.6. Rancangan *Interface Detail Supplier*

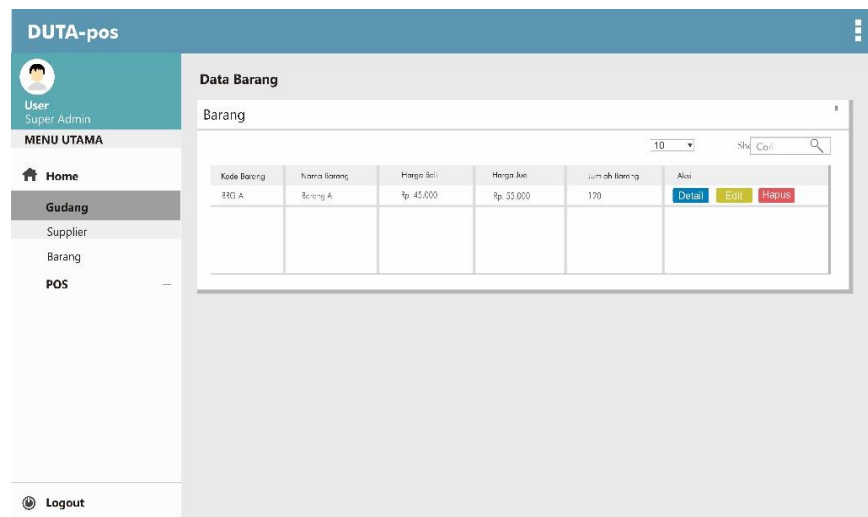
Interface ini merupakan tampilan detail dari informasi *supplier*. Pada *interface* ini dapat dilihat barang-barang yang masih ada di dalam gudang. Data barang juga didapat oleh sistem SSG. Tampilan detail *supplier* dapat dilihat pada Gambar 64.



Gambar 64. Rancangan *Interface Detail Supplier*.

5.7. Rancangan *Interface* Master Barang

Pada *interface* ini menampilkan data-data barang yang ada di sistem. Data barang yang ditampilkan dapat dilihat detailnya, dihapus, dan dapat diperbaharui. Tampilan data barang dapat dilihat pada Gambar 65.



Gambar 65. Rancangan *Interface* Master Barang.

5.8. Rancangan *Interface* Detail Barang

Interface ini merupakan tampilan detail dari informasi barang yang ada. Pada *interface* ini dapat dilihat detail dari barang berupa nama, kode, harga jual, harga beli sampai dengan lokasi barang. Tampilan detail barang dapat dilihat pada Gambar 66.

DUTA-pos

User: Super Admin

MENU UTAMA

- Home
- Gudang
- Supplier
- Barang
- POS

Data Barang

Detail Barang

Nama Barang : Jilbab Sogi Empat Kartina

Kode Barang : KAR-123

Harga Jual : Rp. 25.000

Harga Beli : Rp. 18.000

Jumlah : 200

Keterangan : Jilbab segi empat warna dasar

Supplier : Kartina

Input Terakhir : 20 - 01 - 2019 19:20

Lokasi Rak

Nomor Rak	Jumlah Barang	Aksi
Dsa21	12	Detail

Lokasi Display

Nomor Rak	Jumlah Barang	Aksi
Kak	20	Detail

Logout

Gambar 66. Rancangan *Interface* Detail Barang.

5.9. Rancangan *Interface* Display Toko

Pada *interface* ini menampilkan daftar dari barang yang ada di *display* toko. Data barang yang ditampilkan dapat dilihat detailnya, ditambah, dan dapat diperbaharui. Tampilan dapat dilihat pada Gambar 67.

DUTA-pos

User: Super Admin

MENU UTAMA

- Home
- Gudang
- Supplier
- Barang
- POS

Data Display

Display toko +

10 Can

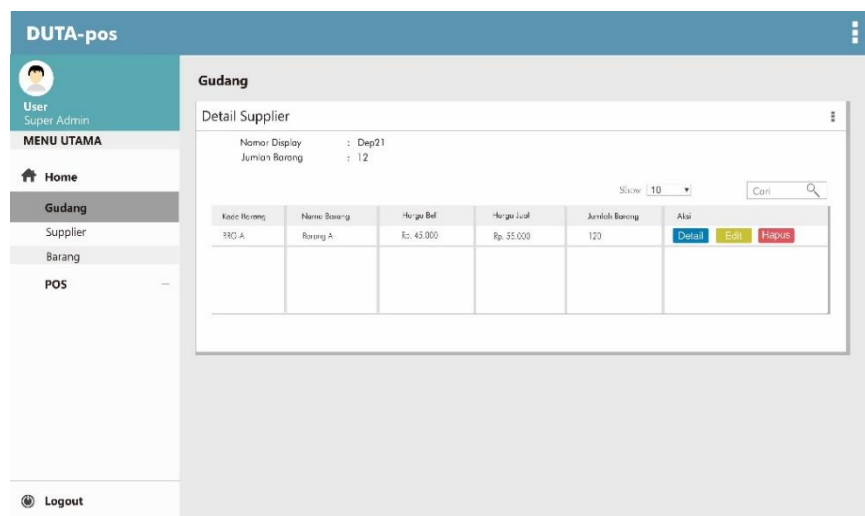
Nomor Display	Jumlah Barang	Aksi
Dsa21	12	Detail Edit Hapus

Logout

Gambar 67. Rancangan *Interface* Display Toko.

5.10. Rancangan *Interface* Detail *Display* Toko

Interface ini merupakan tampilan detail dari informasi barang yang ada di *display* toko. Pada *interface* ini dapat dilihat detail dari barang berupa nomor *display*, jumlah barang, dan lain sebagainya. Pada Gambar 68 menampilkan tampilan detail *display* toko.



Gambar 68. Rancangan *Interface* Detail *Display* toko.

5.11. Rancangan *Interface* Menambah Data *Member*

Interface ini menampilkan menu menambah data *member*. Pengguna akan menambahkan data *member* sesuai informasi yang diberikan calon *member*. Setelah *form* diisi maka data akan disimpan ke *database*. Tampilan rancangan dari menambah data *member* dapat dilihat pada Gambar 69.

DUTA-pos

User
Super Admin

MENU UTAMA

- Home
- Data Member**
- Gudang
- POS

Logout

Data Member

Tambah Member

Nama

Nomor Member

Jenis Kelamin ☒ Laki-laki ☐ Perempuan

Tempat Lahir

Tanggal Lahir

Alamat

No HP

Gambar 69. Rancangan *Interface* Menambah Data *Member*.

5.12. Rancangan *Interface* Data *Member*

Interface ini menampilkan informasi *member* yang tersedia di *database*. Pengguna dapat melihat informasi, merubah, dan menghapus data tersebut. Tampilan rancangan dari banyaknya data *member* dapat dilihat pada Gambar 70.

DUTA-pos

User
Super Admin

MENU UTAMA

- Home
- Data Member**
- Gudang
- POS

Logout

Data Member

Daftar Member

Show 10 entries

Kode	Nama	Alamat	Nomor Telepn	Aksi
187101235..	A	AA	08xxxx	<input type="button" value="Edit"/> <input type="button" value="Delete"/>
187102155..	B	BB	08xxxx	<input type="button" value="Edit"/> <input type="button" value="Delete"/>

Showing 1 to 2 of 2 entries

Gambar 70. Rancangan *Interface* Data *Member*.

G. Pengujian Sistem

Pengujian sistem ini dilakukan dengan *black box testing*. Pengujian yang dilakukan adalah menguji fitur-fitur dari setiap menu yang ada pada sistem. Pada Tabel 6 dapat dilihat bahwa tabel tersebut merupakan *test scenario* dari pengujian *black box testing* untuk super admin.

Tabel 6. *Test Scenario Black-Box* Super Admin

No.	Kelas Uji	Daftar Pengujian	Kasus Uji	Hasil yang Diharapkan
1.	Login	<i>Input data login</i>	Mengisi <i>username</i> dan <i>password</i> dengan benar.	Berhasil masuk ke sistem.
			Mengisi <i>username</i> dan <i>password</i> salah.	Sistem menolak dengan memberikan pesan <i>error</i> .
			Tidak mengisi salah satu <i>field</i> yang disediakan.	Sistem menolak dan memberikan pesan <i>error</i> .
2.	Fungsi Menambah User	<i>Input data user baru</i>	Mengisi <i>form input</i> dengan data yang sesuai.	Data disimpan ke <i>database</i> .
			Mengisi <i>form</i> dengan <i>username</i> yang sama.	Sistem tidak dapat menyimpan data.
			Mengisi <i>password</i> dan <i>confirm password</i> tidak sama.	Sistem tidak dapat menyimpan data.
			Tidak mengisi salah satu <i>field</i> yang disediakan.	Sistem tidak dapat menyimpan data.

Tabel 6. *Test Scenario Black-Box Super Admin (Lanjutan)*

No.	Kelas Uji	Daftar Pengujian	Kasus Uji	Hasil yang Diharapkan
3.	Fungsi Ubah Password	Input data ubah password	Mengisi <i>form</i> data dengan sesuai.	<i>Password user</i> berhasil diubah.
			Mengisi <i>password</i> Super Admin salah.	Sistem menolak mengubah <i>password user</i> .
			Mengisi <i>password</i> baru dan <i>confirm password</i> tidak sama.	Sistem menolak mengubah <i>password user</i> .
4.	Fungsi Ubah Level	Input data ubah level	Mengisi semua <i>form</i> data dengan sesuai.	<i>Level user</i> berhasil diubah.
			Mengisi <i>password</i> Super Admin salah.	Sistem menolak mengubah <i>level user</i> .
5.	Fungsi Hapus User	Memilih tombol hapus <i>user</i>	Mengisi <i>password</i> Super Admin dengan benar.	Data <i>user</i> berhasil dihapus.
			Mengisi <i>password</i> Super Admin salah.	Sistem menolak menghapus data <i>user</i> .
6.	Fungsi Aktivitas User	Mengakses menu aktivitas <i>user</i>	Memilih menu aktivitas <i>user</i> .	Menampilkan daftar data aktivitas <i>user</i> .
			Memilih tombol <i>copy</i>	Menduplikat daftar data aktivitas <i>user</i> .
			Memilih tombol <i>.excel</i> dan <i>.pdf</i>	<i>Mendownload</i> daftar data aktivitas <i>user</i> .
			Memilih tombol <i>print</i> .	Mencetak daftar data aktivitas <i>user</i> .

Pengujian *black box testing* pada sistem dilakukan juga untuk kasir, ada beberapa pengujian yang akan diuji untuk sistem kasir. Pengujian *black box testing* untuk kasir dapat dilihat pada Tabel 7.

Tabel 7. *Test Scenario Black-Box* pada Sistem Untuk Kasir

No.	Kelas Uji	Daftar Pengujian	Kasus Uji	Hasil yang Diharapkan
1.	Login	<i>Input data login</i>	Mengisi <i>username</i> dan <i>password</i> dengan benar.	Berhasil masuk ke sistem.
			Mengisi <i>username</i> dan <i>password</i> salah.	Sistem menolak dengan memberikan pesan <i>error</i> .
			Tidak mengisi salah satu <i>field</i> yang disediakan.	Sistem menolak dan memberikan pesan <i>error</i> .
2	Fungsi <i>Supplier</i>	Mengakses menu <i>supplier</i>	Memilih menu <i>supplier</i> .	Menampilkan daftar data <i>supplier</i> .
			Memilih tombol <i>copy</i> .	Menduplikat daftar data <i>supplier</i> .
			Memilih tombol .excel dan .pdf	<i>Mendownload</i> daftar data <i>supplier</i> .
			Memilih tombol <i>print</i> .	Mencetak daftar data <i>supplier</i> .

Tabel 7. *Test Scenario Black-Box* pada Sistem Untuk Kasir (Lanjutan)

No.	Kelas Uji	Daftar Pengujian	Kasus Uji	Hasil yang Diharapkan
3	Fungsi <i>point-of-sales</i>	Memasukkan data barang yang dibeli	<i>Scan barcode</i> dengan <i>barcode scanner</i> atau <i>input</i> pada <i>form</i> .	Barang masuk kedalam daftar barang yang dibeli.
			<i>Input</i> kode barang dengan data yang tidak <i>valid</i> .	Menampilkan pesan kesalahan barang tidak ditemukan.
		Mengubah jumlah per-barang yang dibeli	Menambah Jumlah barang pada kotak <i>input</i> = "3".	Jumlah per-barang bertambah.
			Mengubah jumlah barang pada kotak <i>input</i> menjadi nol atau lebih kecil.	Menampilkan pesan kesalahan, jumlah barang tidak dapat kurang dari nol.
		Menghapus data barang yang tidak jadi dibeli	Klik tombol "X" pada barang yang tidak jadi dibeli.	Barang yang dipilih terhapus.
			Mengisi kode <i>member</i> dengan benar.	Menampilkan data <i>member</i> .
		Melakukan proses transaksi	Mengisi kode <i>member</i> tidak sesuai.	Sistem menolak dan memberikan pesan bahwa kode <i>member</i> tidak terdaftar.
			Memilih jumlah diskon.	Menampilkan jumlah diskon dan memotong total belanjaan yang harus dibayar.
			Memilih jenis pembayaran tunai atau kredit.	Menampilkan jenis pembayaran tunai atau kredit.

Tabel 7. *Test Scenario Black-Box* pada Sistem Untuk Kasir (Lanjutan)

No.	Kelas Uji	Daftar Pengujian	Kasus Uji	Hasil yang Diharapkan
3	Fungsi <i>point-of-sales</i>	Melakukan proses transaksi	Mengisi keterangan diskon.	Menampilkan isian dari keterangan diskon.
			<i>Input</i> jumlah pembayaran kurang dari total belanjaan.	Menampilkan pesan kesalahan, jumlah pembayaran kurang.
			<i>Input</i> jumlah pembayaran pas dengan total belanjaan.	Menampilkan uang kembalian sebesar Rp 0,-.
			<i>Input</i> jumlah pembayaran lebih dari total belanjaan.	Menampilkan jumlah uang kembalian.
			Memilih tombol bayar.	Data transaksi disimpan ke <i>database</i> , struk barang tercetak.
			Memilih tombol batal.	Transaksi dibatalkan.

Pengujian *black box testing* pada sistem dilakukan juga untuk admin gudang, ada beberapa pengujian yang akan diuji pada sistem untuk admin gudang.

Pengujian *black box testing* untuk admin gudang dapat dilihat pada Tabel 8.

Tabel 8. *Test Scenario Black-Box* pada Sistem Untuk Admin Penjualan

No.	Kelas Uji	Daftar Pengujian	Kasus Uji	Hasil yang Diharapkan
1.	Login	<i>Input data login</i>	Mengisi <i>username</i> dan <i>password</i> dengan benar.	Berhasil masuk ke sistem.
			Mengisi <i>username</i> dan <i>password</i> salah.	Sistem menolak dengan memberikan pesan <i>error</i> .
			Tidak mengisi salah satu <i>field</i> yang disediakan.	Sistem menolak dan memberikan pesan <i>error</i> .
			Mengisi <i>form input</i> dengan data yang sesuai.	Data disimpan ke <i>database</i> , menampilkan pesan data berhasil ditambah.
2.	Fungsi Menambah Data <i>Department</i>	<i>Input data department baru</i>	Mengisi <i>form</i> dengan nama <i>department</i> yang sama.	Sistem menolak dan memberikan pesan <i>error</i> .
			Mengisi <i>form</i> dengan nomor telepon yang sama.	Sistem menolak dan memberikan pesan <i>error</i> .
			Tidak mengisi salah satu <i>field</i> yang disediakan.	Sistem menolak dan memberikan pesan <i>error</i> .
			Mengisi <i>form</i> data dengan sesuai.	Data <i>department</i> berhasil diubah.
3.	Fungsi <i>Edit Department</i>	<i>Input data edit department</i>	Tidak mengisi salah satu <i>field</i> yang disediakan.	Sistem menolak mengubah data <i>department</i> .
			Memilih tombol batal.	Sistem kembali ke menu <i>department</i> .

Tabel 8. *Test Scenario Black-Box* pada Sistem Untuk Admin Penjualan (Lanjutan)

No.	Kelas Uji	Daftar Pengujian	Kasus Uji	Hasil yang Diharapkan
4.	Fungsi Hapus <i>Department</i>	Memilih tombol hapus <i>department</i>	Memilih tombol oke.	Data <i>department</i> berhasil dihapus.
			Memilih tombol batal.	Kembali ke menu <i>department</i> .
5	Fungsi Lihat Detail <i>Department</i>	Mengakses menu <i>department</i>	Memilih tombol detail <i>department</i> .	Menampilkan data <i>department</i> secara detail.
			Mengisi <i>form input</i> dengan data yang sesuai.	Data disimpan ke <i>database</i> , menampilkan pesan data berhasil ditambah.
			Mengisi <i>form</i> dengan IP kasir yang sama.	Sistem menolak dan memberikan pesan error.
			Tidak memasukkan data <i>department</i> pada menu <i>department</i> .	Sistem tidak menampilkan daftar nama <i>department</i> .
6	Fungsi Menambah Identitas Kasir	<i>Input</i> data kasir baru	Memilih tombol <i>search</i> pada nama <i>department</i> .	Menampilkan daftar nama <i>department</i> .
			Memilih tombol " <i>close</i> " pada daftar nama <i>department</i> .	Sistem kembali ke <i>form</i> tambah identitas kasir.
			Memilih tombol " <i>batal</i> " pada <i>form</i> tambah identitas kasir.	Sistem kembali ke menu identitas kasir.
			Tidak mengisi <i>field</i> keterangan.	Tambah data identitas kasir bisa tetap tersimpan ke <i>database</i> .

Tabel 8. *Test Scenario Black-Box* pada Sistem Untuk Admin Penjualan (Lanjutan)

No.	Kelas Uji	Daftar Pengujian	Kasus Uji	Hasil yang Diharapkan
7	Fungsi "Search" Pada Menu Identitas Kasir	<i>Input</i> kata kunci.	Memasukkan kata kunci yang dicari.	Menampilkan data yang sama dengan hasil pencarian.
8	Fungsi Lihat Detail Identitas Kasir	Mengakses menu identitas kasir.	Memilih tombol detail identitas kasir.	Menampilkan data kasir secara detail.
			Mengisi <i>form</i> data dengan sesuai.	Data identitas kasir berhasil diubah.
9	Fungsi <i>Edit</i> Kasir	<i>Input</i> data <i>edit</i> kasir.	Memasukkan IP kasir yang sama dengan IP kasir yang lain.	Sistem menolak dan memberikan pesan <i>error</i> .
			Memilih tombol batal.	Sistem kembali ke menu identitas kasir.
			Memilih tombol oke.	Data kasir berhasil dihapus.
10	Fungsi Hapus Kasir	Memilih tombol hapus kasir.	Memilih tombol batal.	Kembali ke menu identitas kasir.
			Memilih tombol mulai transaksi.	Sesi penjualan statusnya "berjalan".
11	Fungsi Mengelola Sesi Penjualan	Memilih menu sesi penjualan.	Memilih tombol akhiri transaksi.	Sesi penjualan statusnya "ditutup".
			Memilih tombol buka kembali transaksi.	Sesi penjualan statusnya "berjalan".

Tabel 8. *Test Scenario Black-Box* pada Sistem Untuk Admin Penjualan (Lanjutan)

No.	Kelas Uji	Daftar Pengujian	Kasus Uji	Hasil yang Diharapkan
12	Fungsi Lihat Detail Data Penjualan	Mengakses menu sesi penjualan.	Memilih tombol detail sesi penjualan.	Menampilkan data penjualan secara detail.
		Mengakses tombol akses menu POS.	Memilih menu POS dengan status sesi penjualan "berjalan". Memilih menu POS dengan status sesi penjualan "ditutup".	Sistem menampilkan menu POS. Sesi menolak mengakses menu POS.
13	Fungsi Menu Penjualan	Mengakses tombol detail penjualan.	Memilih tombol detail penjualan.	Menampilkan data penjualan secara detail.
		Melakukan <i>return</i> barang.	Memilih tombol <i>return</i> barang.	Data barang yang di <i>return</i> berhasil disimpan ke <i>database</i> .
			Mengisi <i>form input</i> dengan data yang sesuai.	Data disimpan ke <i>database</i> , menampilkan pesan data berhasil ditambah.
14	Fungsi Menambah Member	<i>Input</i> data <i>member</i> baru	Mengisi <i>form</i> dengan kode <i>member</i> yang sama.	Sistem menolak dan memberikan pesan <i>error</i> .
			Tidak mengisi salah satu <i>field</i> yang disediakan.	Sistem menolak dan memberikan pesan <i>error</i> .

Tabel 8. *Test Scenario Black-Box* pada Sistem Untuk Admin Penjualan (Lanjutan)

No.	Kelas Uji	Daftar Pengujian	Kasus Uji	Hasil yang Diharapkan
14	Fungsi Menambah <i>Member</i>	<i>Input data member baru</i>	Memilih tombol "simpan".	Data disimpan ke <i>database</i> , menampilkan pesan data berhasil ditambah.
			Memilih tombol "batal".	Sistem kembali ke menu <i>member</i> .
			Tidak memilih jenis kelamin.	Sistem menolak dan memberikan pesan <i>error</i> .
15	Fungsi "Search" Pada Menu <i>Member</i>	<i>Input kata kunci</i>	Memasukkan kata kunci yang dicari.	Menampilkan data yang sama dengan hasil pencarian.
16	Fungsi Lihat Detail Data <i>Member</i>	Mengakses menu <i>member</i>	Memilih tombol detail <i>member</i> .	Menampilkan data <i>member</i> secara detail.
			Mengisi <i>form</i> data dengan sesuai.	Data <i>member</i> berhasil diubah.
			Memasukkan kode <i>member</i> yang sama dengan <i>member</i> yang lain.	Sistem menolak dan memberikan pesan <i>error</i> .
17	Fungsi <i>Edit Member</i>	<i>Input data edit member</i>	Memilih tombol batal.	Sistem kembali ke menu identitas kasir.
			Memilih tombol "simpan".	Data berhasil diubah dan disimpan ke <i>database</i> .
			Memilih tombol "batal".	Sistem kembali ke menu <i>member</i> .

Tabel 8. *Test Scenario Black-Box* pada Sistem Untuk Admin Penjualan (Lanjutan)

No.	Kelas Uji	Daftar Pengujian	Kasus Uji	Hasil yang Diharapkan
18	Fungsi Hapus <i>Member</i>	Memilih tombol hapus <i>member</i>	Memilih tombol "oke".	Data <i>member</i> berhasil dihapus.
			Memilih tombol "batal".	Kembali ke menu <i>member</i> .
			Mengisi <i>form input</i> dengan data yang sesuai.	Data disimpan ke <i>database</i> , menampilkan pesan data berhasil ditambah.
			Memilih nama <i>supplier</i> .	Nama <i>supplier</i> yang terpilih ditampilkan.
19	Fungsi Menambah Data Barang	<i>Input</i> data barang baru	Tidak menambahkan data <i>supplier</i> pada menu <i>supplier</i> .	Sistem tidak menampilkan daftar nama <i>supplier</i> .
			Menambahkan data <i>supplier</i> pada menu <i>supplier</i> .	Menampilkan daftar nama <i>supplier</i> .
			Memilih tombol "simpan" pada <i>form</i> tambah data barang.	Data disimpan ke <i>database</i> , menampilkan pesan data berhasil ditambah.
			Memilih tombol "batal" pada <i>form</i> tambah data barang.	Sistem kembali ke menu master barang.
			Tidak mengisi <i>field</i> keterangan.	Tambah data barang bisa tetap tersimpan ke <i>database</i> .
20	Fungsi Lihat Detail Data Barang	Memilih tombol "detail barang"	Memilih tombol detail barang.	Menampilkan data barang secara detail.

Tabel 8. *Test Scenario Black-Box* pada Sistem Untuk Admin Penjualan (Lanjutan)

No.	Kelas Uji	Daftar Pengujian	Kasus Uji	Hasil yang Diharapkan
21	Fungsi Cetak <i>Barcode</i> Barang	Memilih tombol "cetak <i>barcode</i> "	Memilih tombol "cetak".	<i>Barcode</i> barang berhasil di cetak.
			Memilih tombol "batal".	Kembali ke menu data barang.
			Mengisi <i>form input</i> dengan data yang sesuai.	Data disimpan ke <i>database</i> , menampilkan pesan data berhasil ditambah.
22	Fungsi Menambah Data <i>Display</i> Toko	<i>Input</i> data <i>display</i> toko baru	Mengisi <i>form</i> dengan nomor <i>display</i> yang sama.	Sistem menolak dan memberikan pesan <i>error</i> .
			Mengisi <i>form</i> dengan keterangan yang sama.	Sistem menolak dan memberikan pesan <i>error</i> .
			Tidak mengisi salah satu <i>field</i> yang disediakan.	Sistem menolak dan memberikan pesan <i>error</i> .
23	Fungsi Lihat Detail <i>Display</i> Toko	Mengakses tombol "detail <i>display</i> toko"	Memilih tombol "detail <i>display</i> toko".	Menampilkan data <i>display</i> toko secara detail.
24	Fungsi Hapus <i>Display</i> Toko	Memilih tombol hapus <i>display</i> toko	Memilih tombol "oke".	Data <i>display</i> toko berhasil dihapus.
			Memilih tombol "batal".	Kembali ke menu <i>display</i> toko.
25	Fungsi Lihat Detail Kategori Barang	Mengakses tombol "detail kategori barang"	Memilih tombol "detail kategori barang".	Menampilkan data kategori barang secara detail.

Tabel 8. *Test Scenario Black-Box* pada Sistem Untuk Admin Penjualan (Lanjutan)

No.	Kelas Uji	Daftar Pengujian	Kasus Uji	Hasil yang Diharapkan
26	Fungsi <i>Edit Display</i> Toko	<i>Input</i> data <i>display</i> toko	Mengisi <i>form</i> data dengan sesuai.	Data <i>display</i> toko berhasil diubah.
			Memasukkan nomor <i>display</i> yang sama dengan data yang lain.	Sistem menolak dan memberikan pesan <i>error</i> .
			Memasukkan keterangan yang sama dengan data yang lain.	Sistem menolak dan memberikan pesan <i>error</i> .
			Memilih tombol "simpan".	Data berhasil diubah dan disimpan ke <i>database</i> .
27	Fungsi Lihat Detail Rak Gudang	Mengakses tombol "detail rak gudang"	Memilih tombol "batal".	Sistem kembali ke menu <i>display</i> toko.
			Memilih tombol "detail rak gudang".	Menampilkan data rak gudang secara detail.

Pengujian *black box testing* pada sistem dilakukan juga untuk *customer*, ada beberapa pengujian yang akan diuji *customer*. Pengujian *black box testing* untuk *customer* dapat dilihat pada Tabel 9.

Tabel 9. *Test Scenario Black-Box* pada Sistem Untuk *Customer*

No.	Kelas Uji	Daftar Pengujian	Kasus Uji	Hasil yang Diharapkan
1.	Fungsi informasi barang	Menampilkan informasi barang.	Mencari berdasarkan <i>barcode</i> .	Menampilkan hasil pencarian.
			Mencari berdasarkan nama barang.	Menampilkan hasil pencarian berdasarkan nama barang.
		Menampilkan barang per kategori.	Memilih kategori barang.	Menampilkan barang kategori terpilih.

V. SIMPULAN DAN SARAN

A. Simpulan

Simpulan dari hasil penelitian ini sebagai berikut:

1. sistem ini telah berhasil melakukan pencatatan *point of sales*,
2. sistem ini telah berhasil mencetak bukti pembayaran transaksi penjualan,
3. sistem ini telah berhasil mengirim data ke Sistem *Business Intelligence Dashboard* dan Sistem Informasi Akuntansi menggunakan API (*Application Programming Interface*),
4. sistem ini telah berhasil menerima data dari Sistem *Supplier Relationship Management* yang kemudian memanfaatkan data tersebut,
5. sistem ini dapat membantu *customer* dalam mencari informasi data barang yang tersedia di setiap *department*, dan
6. dari hasil pengujian *Black Box Testing* Sistem *Customer Relationship Management* (CRM) dapat berfungsi sesuai kebutuhan pengguna.

B. Saran

Berdasarkan perancangan dan hasil implementasi program sistem, maka beberapa saran yang perlu diperhatikan dalam mengembangkan sistem ini sebagai berikut:

1. Sistem ini dalam metode pencariannya menggunakan Metode *Search Engine* untuk pengumpulan data pencarian yang dilakukan oleh *customer*,
2. Perlu dibuat satu sistem sebagai proses pengiriman data melalui API,
3. Menambahkan fitur akun *member* agar *member* dapat melihat informasi personalnya, dan
4. Menambahkan fitur yang lebih berfokus terhadap pelayanan *customer* dalam penyampaian informasi terbaru dari perusahaan.

DAFTAR PUSTAKA

- Bootstrap. (2014). About · Bootstrap.
- Budiardjo, E. K., Irwiensyah, F., & Obstetri, D. (2008). Analisis Fitur Crm Untuk Meningkatkan Kepuasan Pasien - Berbasis Pada Framework of Dynamic CRM Studi Kasus : Departemen Obstetri Dan Ginekologi Fkui - Rscm Jakarta. *Perspective*, 2008(semnasIF), 349–357.
- Dharwiyanti, S., & Wahono, R. S. (2003). Pengantar Unified Modeling Language (UML). *IlmuKomputer.Com*, 1–13. <https://doi.org/http://dx.doi.org/10.1109/ICCSN.2010.19>
- Dyantina, O., Afrina, M., & Ibrahim, A. (2012). Penerapan Customer Relationship Management (CRM) Berbasis Web (Studi Kasus Pada Sistem Informasi Pemasaran di Toko YEN-YEN) 1,2,3. *Sistem Informasi*, 4(2), 516–529.
- Hamidin, D. (2008). Model Customer Relationship Management (CRM) di INSTITUSI PENDIDIKAN. *Seminar Nasional Aplikasi Teknologi Informasi 2008 (SNATI 2008)*, 2008(Snati), 31–34.
- Hidayat, D. A. (2014). Rancang Bangun Aplikasi Point of Sale (POS) berbasis Web dengan Pemanfaatan Trigger pada Distribution Store CV. NMRQ. *JustIN*, 2(1).
- Humas Kementerian Koperasi dan UKM. (2017). Ratio Wirausaha Indonesia Naik Jadi 3,1 Persen - www.depkop.go.id.
- Kurniawan, I., & Sani, R. R. (2019). Pemodelan SCRUM dalam Pengembangan Sistem Informasi Kesehatan pada Klinik Ar-Rokhim Sragen Kabupaten Sragen. *Journal of Informastion System*, 4(1), 76–86.
- Mahdiana, D. (2011). “Analisa dan Rancangan Sistem Informasi Pengadaan Barang dengan Metodologi Berorientasi Obyek: Studi Kasus PT . Liga Indonesia.” *Jurnal Telematika*, 3, 36–43.
- Nurseto, Tejo. (2010). Jurnal Ekonomi & Pendidikan, Volume 7 Nomor 1, April 2010. *Jurnal Ekonomi & Pendidikan*, 7(April), 58–81

- Oracle Corporation. (2018). MySQL :: MySQL 8.0 Reference Manual :: 1.3.1 What is MySQL?
- Otwell, T. (2012). Introduction - Laravel - The PHP Framework For Web Artisans.
- PHP: What is PHP? - Manual. (2001). Retrieved January 22, 2019, from <http://php.net/manual/en/intro-what-is.php>
- Phpmyadmin. (2018). About phpMyAdmin. Retrieved March 23, 2019, from <https://www.phpmyadmin.net/>
- Ruseno, N. (2019). Implementasi SCRUM Pada Pengembangan Aplikasi Sistem Reservasi Online Menggunakan PHP. *Gerbang*, 9(1), 8–15.
- Schwaber, K., & Sutherland, J. (2017). The Scrum Guide: The Definitive The Rules of the Game. *Scrum.Org and ScrumInc*, (November <https://doi.org/10.1053/j.jrn.2009.08.012>
- Sitepoint. (2013). Best PHP Frameworks for 2014 — SitePoint.
- Speed, J., & Engineering, S. P. (2009). Arif Setyawan, Joko Wandyatmono Sistem Informasi Penggajian Pegawai Kecamatan Geneng Kabupaten Ngawi Universitas Surakarta, 1(3), 1–7.