



**RANCANG BANGUN APLIKASI *M*ARKETPLACE STIKOM SURABAYA  
BERBASIS ANDROID**



Oleh:

**Nanda Rizka Maulana**

**15410100119**

---

**FAKULTAS TEKNOLOGI DAN INFORMATIKA  
INSTITUT BISNIS DAN INFORMATIKA STIKOM SURABAYA  
2019**

**RANCANG BANGUN APLIKASI *MARKETPLACE* STIKOM SURABAYA  
BERBASIS ANDROID**

**TUGAS AKHIR**

Diajukan sebagai salah satu syarat untuk menyelesaikan

Program Sarjana



**FAKULTAS TEKNOLOGI DAN INFORMATIKA  
INSTITUT BISNIS DAN INFORMATIKA STIKOM SURABAYA  
2019**

**TUGAS AKHIR**  
**RANCANG BANGUN APLIKASI MARKETPLACE STIKOM SURABAYA**  
**BERBASIS ANDROID**

Dipersiapkan dan disusun oleh:

**Nanda Rizka Maulana**

**NIM: 15410100119**

Telah diperiksa, diuji, dan disetujui oleh Dewan Pengaji

Pada : Juli 2019

**Susunan Dewan Pengaji**

Pembimbing

I. **Dr. Bambang Hariadi, M.Pd.**

NIDN 0719106401

II. **Tan Amelia, S.Kom., M.MT.**

NIDN 0728017602

Pengaji

I. **Teguh Sutanto, M.Kom.**

NIDN 0713027801

INSTITUT BISNIS  
DAN INFORMATIKA  
stikom SURABAYA

X/8/19  
Medieke  
Z/19

Tugas Akhir ini telah diterima sebagai salah satu persyaratan



Dekan Fakultas Teknologi dan Informatika

**FAKULTAS TEKNOLOGI DAN INFORMATIKA**  
**INSTITUT BISNIS DAN INFORMATIKA STIKOM SURABAYA**  
**2019**

**SURAT PERNYATAAN**  
**PERSETUJUAN PUBLIKASI DAN KEASLIAN KARYA ILMIAH**

Sebagai mahasiswa Institut Bisnis dan Informatika Stikom Surabaya, saya:

Nama : Nanda Rizka Maulana  
Nim : 15.41010.0119  
Program Studi : S1 Sistem Informasi  
Fakultas : Fakultas Teknologi dan Informatika  
Jenis Karya : Tugas Akhir  
Judul Karya : **RANCANG BANGUN APLIKASI *MARKEPLACE*  
STIKOM SURABAYA BERBASIS ANDROID**

Menyatakan dengan sesungguhnya bahwa:

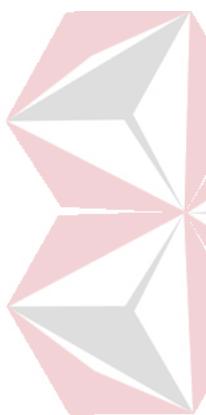
1. Demi pembangunan Ilmu Pengetahuan, Teknologi dan Seni, saya menyetujui memberikan kepada Institut Bisnis dan Informatika Stikom Surabaya Hak Bebas Royaliti Non-Eksklusif (Non-Exclusive royaliti Free Right) atas seluruh isi/sebagian karya ilmiah saya tersebut di atas untuk disimpan, dialihmediakan dan dikelolah dalam bentuk pangkalan data (database) untuk selanjutnya didistribusikan atau dipublikasikan demi kepentingan akademis dengan tetap mencantumkan nama saya sebagai penulis atau pencipta dan sebagai pemilik Hak Cipta
2. Karya tersebut di atas adalah karya asli saya, bukan plagiat baik sebagian maupun keseluruhan, kutipan, karya atau pendapat orang lain yang ada dalam karya ilmiah ini adalah semata hanya rujukan yang dicantumkan dalam Daftar Pustaka saya
3. Apabila dikemudian hari ditemukan dan terbukti terdapat tindakan plagiat pada karya ilmiah ini, maka saya bersedia untuk menerima pencabutan terhadap gelar kesarjanaan yang telah diberikan kepada saya.

Demikian surat pernyataan ini saya buat dengan sebenarnya.

Surabaya, Juli 2019  
Yang menyatakan

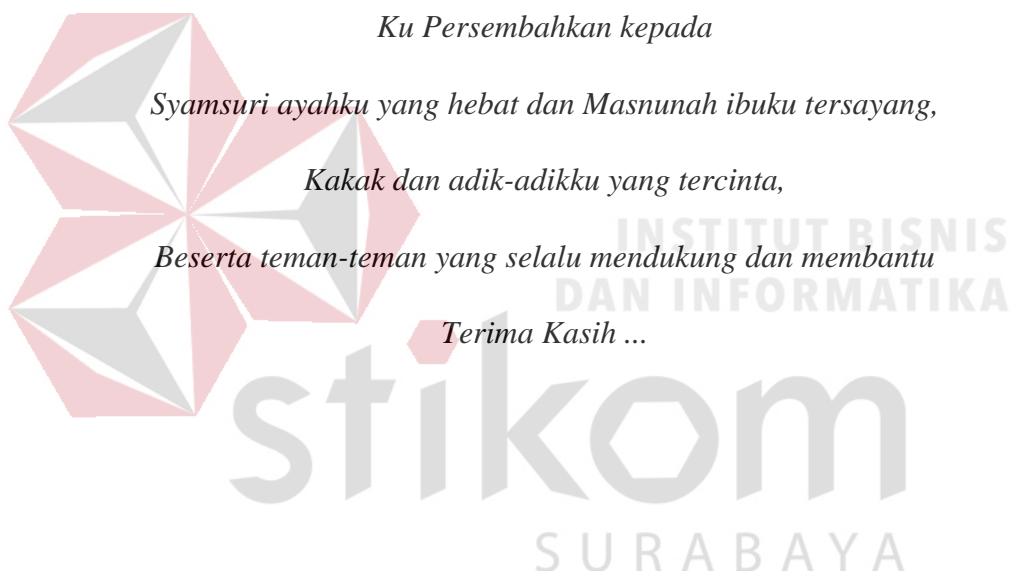


Nanda Rizka Maulana  
Nim: 15410100119



*“Hidup itu sederhana,  
Jika merasa lelah istirahatlah,  
Jika merasa stress berliburlah,  
Jika merasa bodoh belajarlah,  
Jika merasa kesulitan carilah bantuan,  
Jika merasa jauh dari Tuhan maka dekatilah dengan beribadah”*

**Stikom**  
INSTITUT BISNIS  
DAN INFORMATIKA  
SURABAYA



*Ku Persembahkan kepada*

*Syamsuri ayahku yang hebat dan Masnunah ibuku tersayang,*

*Kakak dan adik-adikku yang tercinta,*

*Beserta teman-teman yang selalu mendukung dan membantu*

*Terima Kasih ...*

**stikom**  
SURABAYA

## ABSTRAK

Saat ini Stikom Surabaya telah memiliki marketplace dalam bentuk web yang dapat menampung karya mahasiswa dan dosen, marketplace web ini sudah memiliki fitur-fitur utama yang diperlukan untuk dapat melakukan transaksi jual beli. Jenis marketplace Stikom Surabaya ini adalah global, yaitu marketplace yang menjual berbagai macam jenis produk yang tidak memiliki keterkaitan dengan produk lainnya.

Berdasarkan penjabaran di atas dapat diketahui bahwa masalah yang terjadi yaitu dibutuhkan platform yang dapat memperluas pangsa pasar marketplace Stikom Surabaya dan mempermudah pengguna untuk mengakses marketplace Stikom Surabaya. Data statistik Pangsa pasar Operation System (OS) global dari 2009-2018 disimpulkan bahwa pada saat kuartal kedua 2018 sebanyak 88% pangsa pasar OS global dikuasai oleh Android sementara 11.9% adalah IOS dan sisa 0.1% nya adalah OS lainnya. Platform baru yang dikembangkan untuk marketplace Stikom Surabaya akan menggunakan Operation System Android. Berdasarkan permasalahan yang ada maka akan dilakukan penelitian untuk merancang dan membangun aplikasi *marketplace* Stikom Surabaya berbasis android.

Kesimpulan yang dapat dapat diambil berdasarkan hasil implementasi dan evaluasi dari pembuatan aplikasi *Marketplace* Stikom Surabaya berbasis Android adalah menghasilkan aplikasi yang dapat digunakan sebagai wadah untuk melakukan transaksi penjualan dan pembelian. Aplikasi ini juga dapat membantu penjual maupun pembeli untuk memonitor dan memproses transaksi yang sedang berlangsung.

**Kata Kunci:** Rancang Bangun, Aplikasi *Marketplace*, Android

## KATA PENGANTAR

Dengan memanjangkan puji syukur kehadirat Tuhan Yang Maha Esa atas berkah dan rahmat-Nya sehingga penulis dapat menyelesaikan Tugas Akhir ini yang berjudul “Rancang Bangun Aplikasi *Marketplace* Stikom Surabaya Berbasis Android”. Adapun maksud Tugas Akhir ini disusun untuk memenuhi persyaratan dalam menyelesaikan Program Studi Strata Satu (S1) di Institut Bisnis dan Informatika Stikom Surabaya.

Dalam penyusunan laporan Tugas Akhir ini tidak lepas dari bantuan banyak pihak yang benar-benar memberikan masukan dan dukungan kepada penulis. Untuk itu, pada kesempatan ini perkenankan penulis untuk mengucapkan terima kasih kepada:

1. Orangtua serta saudara saya yang selalu mendukung dan mendoakan saya sehingga mampu untuk menyelesaikan Tugas Akhir ini.
2. Bapak Dr. Bambang Hariadi, M.Pd . selaku Dosen Pembimbing I yang telah memberikan arahan, bimbingan, pengalaman serta motivasi dalam proses pembuatan laporan Tugas Akhir ini.
3. Ibu Tan Amelia, S.Kom., M.MT. selaku Dosen Pembimbing II yang telah meluangkan waktunya untuk membimbing, mengarahkan, mengoreksi serta memberikan banyak masukan positif dalam proses pembuatan laporan Tugas Akhir ini.
4. Bapak Teguh Sutanto, M.Kom. selaku Dosen Penguji yang telah memberikan kritik, masukan, maupun saran dalam membantu penyempurnaan laporan Tugas Akhir ini.

5. Bapak Dr. Anjik Sukmaaji, S.Kom., M.Eng. selaku Ketua Program Studi S1 Sistem Informasi.

Semoga Tuhan Yang Maha Esa memberikan karma baik-Nya kepada pihak- pihak yang telah memberikan banyak hal positif untuk penulis. Penulis menyadari adanya kekurangan-kekurangan di dalam penulisan laporan Tugas Akhir ini. Untuk itu penulis mengharapkan saran dan kritik yang membangun dari pembaca sebagai perbaikan dan pemebelajaran di masa yang akan datang. Semoga laporan Tugas Akhir ini dapat menambah wawasan bagi pembaca.

Surabaya, Juli 2019



## DAFTAR ISI

ABSTRAK .....	vi
KATA PENGANTAR .....	vii
DAFTAR ISI.....	ix
DAFTAR GAMBAR .....	xiii
DAFTAR TABEL.....	xix
BAB I PENDAHULUAN .....	1
1.1. Latar Belakang .....	1
1.2. Rumusan Masalah.....	4
1.3. Batasan Masalah .....	4
1.4. Tujuan .....	5
1.5 Manfaat .....	5
1.6. Sistematika Penulisan .....	5
BAB II LANDASAN TEORI .....	7
2.1. <i>Marketplace</i> .....	7
2.2 <i>Seller</i> (penjual/pedagang) .....	8
2.3 <i>Android</i> .....	9
2.4 <i>Android Studio</i> .....	9
2.5 API ( <i>Application Programming Interface</i> ).....	10
2.6 Software Development Kit (SDK).....	11
2.7 JSON WEB TOKEN (JWT) .....	12
2.8 MySQL .....	13
2.9 UML ( <i>Unified Modeling Language</i> ).....	14
2.10 <i>Black Box Testing</i> .....	16
2.11 <i>System Development Life Cycle</i> (SDLC) .....	17
BAB III ANALISIS DAN PERANCANGAN SISTEM .....	20

3.1	Communication.....	21
3.1.1	Identifikasi masalah .....	21
3.1.2	Analisis Kebutuhan .....	23
3.1.3	Identifikasi kebutuhan data .....	24
3.2	Planning .....	24
3.3	Modeling.....	26
3.3.1	Desain Arsitektur .....	26
3.3.2	Use Case bisnis .....	27
3.3.3	Activity diagram.....	29
3.3.4	<i>Usecase</i> sistem .....	39
3.3.5	Usecase Description.....	44
3.3.6	<i>Sequence Diagram</i> .....	52
3.3.7	Class diagram.....	62
3.3.8	Statechart diagram.....	70
3.3.9	Desain antarmuka pengguna .....	71
3.3.10	Desain uji coba.....	79
BAB IV IMPLEMENTASI DAN EVALUASI.....		82
4.1.	Implementasi.....	82
4.1.1.	Implementasi Kebutuhan Software dan Hardware .....	82
4.1.2.	Implementasi Sistem .....	82
4.1.3.	Tampilan halaman login dan membuat akun baru .....	83
4.1.4.	Tampilan halaman utama .....	84

4.1.5. Tampilan halaman membuka toko .....	88
4.1.6. Tampilan dashboard penjual .....	89
4.1.7. Tampilan halaman penjual mengelola etalase.....	92
4.1.8. Tampilan halaman penjual mengelola produk .....	94
4.1.9. Tampilan halaman pembeli melakukan pemesanan.....	97
4.1.10. Tampilan halaman pembeli melakukan pembelian.....	98
4.1.11. Tampilan halaman pembeli konfirmasi barang diterima .....	99
4.1.12. Tampilan halaman laporan riwayat pemesanan .....	100
4.1.13. Tampilan halaman laporan riwayat pembayaran .....	100
4.1.14. Tampilan halaman laporan riwayat penjualan .....	101
4.2. Evaluasi Sistem.....	102
4.2.1. Uji coba fungsi perangkat lunak .....	102
4.2.2. Uji coba login dan membuat akun baru .....	102
4.2.3. Uji coba pembeli melakukan pemesanan.....	105
4.2.4. Uji coba pembeli melakukan pembelian .....	107
4.2.5. Uji coba penjual melakukan penjualan .....	108
4.2.6. Uji coba membuka toko .....	110
4.2.7. Uji coba penjual mengelola etalase.....	111
4.2.8. Uji coba penjual mengelola produk .....	114
4.2.9. Uji coba laporan riwayat pemesanan .....	117
4.2.10. Uji coba laporan riwayat pembayaran.....	118

4.2.11. Uji coba laporan riwayat penjualan.....	119
4.3 Evaluasi Kesesuaian Akhir .....	120
4.3.1 Membuat Akun Baru.....	120
4.3.2 Membuka Toko .....	121
4.3.3 Mengelola Etalase .....	122
4.3.4 Mengelola Produk .....	122
4.3.5 Melakukan Pemesanan.....	123
4.3.6 Melakukan Pembelian .....	124
4.3.7 Melakukan Penjualan .....	126
4.3.8 Melihat Riwayat Pesanan.....	128
4.3.9 Melihat Riwayat Pembayaran .....	128
4.3.10 Melihat Riwayat Penjualan .....	129
BAB V PENUTUP.....	130
5.1 Kesimpulan .....	130
5.2 Saran .....	130
DAFTAR PUSTAKA .....	131
BIODATA.....	133

## DAFTAR GAMBAR

Gambar 1.1 Pangsa pasar <i>Operation System (OS) global</i> dari 2009-2018 .....	3
Gambar 2.1 JWT .....	12
Gambar 2.2 JWT Header.....	12
Gambar 2.3 JWT PAYLOAD .....	13
Gambar 2.4 JWT Signature.....	13
Gambar 2.5 SDLC Waterfall.....	17
Gambar 3. 1 Tahapan SDLC .....	20
Gambar 3.2 Desain Arsitektur Marketplace Stikom Surabaya .....	26
Gambar 3. 3 <i>Use Case</i> bisnis .....	27
Gambar 3. 4 Pembeli membuat akun baru .....	29
Gambar 3. 5 Penjual membuka toko .....	30
Gambar 3. 6 Penjual mengelola produk .....	31
Gambar 3. 7 Pengguna mengelola saldo .....	32
Gambar 3. 8 Penjual mengelola etalase .....	33
Gambar 3. 9 Pembeli melakukan pemesanan .....	34
Gambar 3. 10 Pembeli melakukan pembelian.....	35
Gambar 3. 11 Penjual melakukan penjualan.....	36
Gambar 3. 12 Pengolahan laporan pemesanan dari pembeli .....	37
Gambar 3. 13 pengolahan laporan pembelian dari pembeli.....	38
Gambar 3. 14 Pengolahan laporan penjualan dari penjual.....	38
Gambar 3. 15 Pembeli membuat akun baru .....	39
Gambar 3. 16 Penjual membuka toko .....	39
Gambar 3. 17 Penjual mengelola produk .....	40

Gambar 3. 18 Pengguna mengelola data saldo .....	40
Gambar 3. 19 Penjual mengelola etalase .....	40
Gambar 3. 20 Pembeli melakukan pemesanan .....	41
Gambar 3. 21 Pembeli melakukan pembelian.....	41
Gambar 3. 22 Penjual melakukan penjualan.....	42
Gambar 3. 23 Pengolahan laporan pemesanan dari pembeli .....	42
Gambar 3. 24 Pengolahan laporan pembelian dari pembeli .....	43
Gambar 3. 25 Pengolahan laporan penjualan dari penjual.....	43
Gambar 3.26 <i>Sequence Diagram</i> Membuat Akun .....	53
Gambar 3.27 <i>Sequence Diagram</i> Membuka Toko.....	54
Gambar 3.28 <i>Sequence Diagram</i> Mengelola Produk.....	54
Gambar 3.29 Mengelola saldo .....	55
Gambar 3.30 <i>Sequence Diagram</i> Mengelola Etalase.....	56
Gambar 3.31 <i>Sequence Diagram</i> Pembeli Melakukan Pemesanan .....	57
Gambar 3.32 <i>Sequence Diagram</i> Pembeli Melakukan Pembelian .....	58
Gambar 3.33 <i>Sequence Diagram</i> Penjual Melakukan Penjualan.....	59
Gambar 3.34 <i>Sequence Diagram</i> Pengolahan Laporan Pemesanan dari Pembeli	60
Gambar 3.35 <i>Sequence Diagram</i> Pengolahan Laporan Pembayaran dari Pembeli .....	61
Gambar 3.36 <i>Sequence Diagram</i> Pengolahan Laporan Penjualan dari Penjual ...	62
Gambar 3.37 <i>Class Diagram</i> Membuat Akun Baru.....	63
Gambar 3.38 <i>Class diagram</i> penjual membuat toko.....	63
Gambar 3.39 <i>Class diagram</i> penjual mengelola produk .....	64
Gambar 3.40 <i>Class diagram</i> mengelola saldo .....	65

Gambar 3.41 <i>Class diagram</i> penjual mengelola etalase .....	66
Gambar 3.42 <i>Class diagram</i> penjual melakukan pemesanan .....	66
Gambar 3.43 <i>class diagram</i> penjual melakukan pembelian .....	67
Gambar 3.44 <i>class diagram</i> penjual melakukan penjualan .....	68
Gambar 3.45 <i>class diagram</i> pengolahan laporan pemesanan .....	69
Gambar 3.46 <i>class diagram</i> pengolahan laporan pembayaran .....	69
Gambar 3.47 <i>class diagram</i> pengolahan laporan penjualan .....	70
Gambar 3. 48 <i>Statechart diagram</i> class modelPenjualan. ....	71
Gambar 3.49 Desain antarmuka penjual mengelola produk .....	74
Gambar 3.50 Desain antarmuka mengelola etalase .....	74
Gambar 3. 51 Desain antarmuka pemesanan .....	75
Gambar 3.52 Desain antarmuka pembelian dan pembayaran .....	76
Gambar 3.53 Desain antarmuka penjualan .....	76
Gambar 3.54 Desain antarmuka laporan pemesanan .....	77
Gambar 3.55 Desain antarmuka laporan pembayaran .....	78
Gambar 3. 56 Desain antarmuka laporan penjualan .....	78
Gambar 4.1 Tampilan halaman login.....	83
Gambar 4.2 Tampilan halaman membuat akun baru .....	84
Gambar 4.3 Tampilan halaman utama aplikasi.....	85
Gambar 4.4 Tampilan detail produk .....	86
Gambar 4.5 Tampilan informasi toko .....	87
Gambar 4.6 Tampilan <i>pop up</i> belum buka toko.....	88
Gambar 4.7 Tampilan form buka toko .....	88
Gambar 4.8 Tampilan dashboard penjual .....	89

Gambar 4.9 Tampilan list pesanan.....	90
Gambar 4.10 Tampilan terima pesanan .....	91
Gambar 4.11 Tampilan tolak pesanan.....	91
Gambar 4.12 Tampilan kirim pesanan .....	91
Gambar 4.13 Tampilan list etalase.....	92
Gambar 4.14 Tampilan tambah etalase .....	93
Gambar 4. 15 Tampilan <i>pop up</i> aksi hapus dan edit .....	93
Gambar 4.16 Tampilan edit etalase.....	94
Gambar 4.17 Tampilan konfirmasi hapus etalase .....	94
Gambar 4.18 Tampilan list produk .....	95
Gambar 4.19 Tampilan tambah produk .....	95
Gambar 4. 20 Tampilan pop up aksi hapus dan edit .....	96
Gambar 4. 21 Tampilan edit produk .....	96
Gambar 4.22 Tampilan konfirmasi hapus produk.....	97
Gambar 4.23 Tampilan Menambahkan Barang Ke Keranjang .....	97
Gambar 4.24 Tampilan keranjang .....	98
Gambar 4.25 Tampilan halaman pembeli melakukan pembelian .....	99
Gambar 4.26 Tampilan konfirmasi barang diterima .....	99
Gambar 4.27 Tampilan laporan riwayat pemesanan.....	100
Gambar 4.28 Tampilan laporan riwayat pembayaran .....	101
Gambar 4.29 Tampilan laporan riwayat penjualan .....	101
Gambar 4.30 Hasil Uji Coba Login User terdaftar .....	103
Gambar 4.31 Hasil Uji Coba Login User Tidak Terdaftar.....	103
Gambar 4.32 Hasil Uji Coba Validasi Isian Kosong .....	104

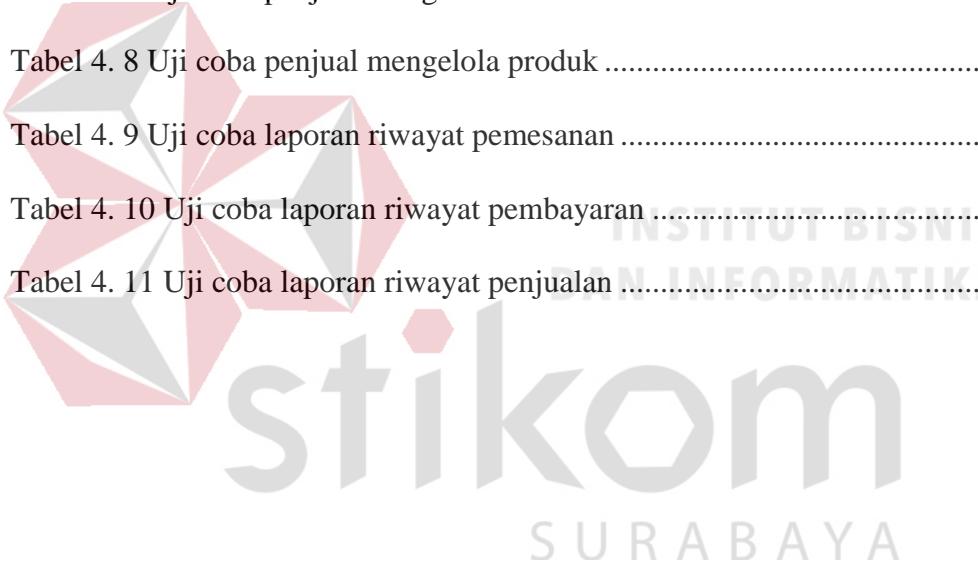
Gambar 4.33 Hasil Uji Coba Mendaftarkan Akun Baru.....	104
Gambar 4.34 Hasil uji coba menguji tombol detail .....	105
Gambar 4.35 Hasil uji coba menguji tombol tambah .....	106
Gambar 4.36 Hasil uji coba tombol keranjang.....	106
Gambar 4.37 Hasil uji coba menampilkan data keranjang .....	107
Gambar 4.38 Hasil uji coba menguji tombol bayar .....	108
Gambar 4.39 Hasil uji coba list pesanan.....	109
Gambar 4.40 Hasil uji coba detail pesanan .....	109
Gambar 4.41 Hasil uji coba tombol slide.....	109
Gambar 4.42 Hasil Uji Coba Validasi Isian.....	110
Gambar 4.43 Hasil Uji Tombol Simpan .....	111
Gambar 4.44 Hasil uji coba menampilkan data etalase .....	112
Gambar 4.45 Hasil uji coba edit etalase.....	112
Gambar 4.46 Hasil uji coba hapus etalase .....	113
Gambar 4.47 Hasil uji coba simpan etalase .....	113
Gambar 4.48 Hasil uji coba menampilkan data produk .....	114
Gambar 4.49 Hasil uji coba edit produk .....	115
Gambar 4. 50 Hasil uji coba hapus produk .....	115
Gambar 4.51 Hasil uji coba tambah produk.....	116
Gambar 4.52 Hasil uji coba validasi isian tambah produk.....	116
Gambar 4.53 Hasil uji coba simpan produk.....	117
Gambar 4. 54 Hasil uji coba menampilkan riwayat pemesanan .....	117
Gambar 4. 55 Hasil uji coba detail riwayat pemesanan .....	118
Gambar 4.56 Hasil uji coba menampilkan riwayat pembayaran .....	118

Gambar 4.57 Hasil uji coba detail riwayat pembayaran .....	119
Gambar 4. 58 Hasil uji coba menampilkan riwayat penjualan .....	119
Gambar 4. 59 Hasil uji coba menampilkan riwayat penjualan .....	120
Gambar 4. 60 Hasil evaluasi mengisi form pendaftaran .....	120
Gambar 4. 61 Hasil evaluasi menggunakan akun terdaftar.....	121
Gambar 4. 62 Hasil evaluasi mengisi form buat toko.....	121
Gambar 4. 63 Hasil evaluasi menambah etalase.....	122
Gambar 4. 64 Hasil evaluasi menambah data produk.....	122
Gambar 4. 65 Hasil evaluasi pembeli memilih dan melihat .....	123
Gambar 4. 66 Hasil evaluasi menambah barang ke keranjang .....	123
Gambar 4. 67 Hasil evaluasi melakukan checkout daftar keranjang .....	124
Gambar 4. 68 Hasil evaluasi mengisi data pembelian .....	124
Gambar 4. 69 Hasil evaluasi melakukan pembayaran .....	125
Gambar 4. 70 Hasil evaluasi penjual menerima pesanan.....	125
Gambar 4. 71 Hasil evaluasi penjual memilih daftar pesanan .....	126
Gambar 4. 72 Hasil evaluasi penjual memproses pesanan .....	126
Gambar 4. 73 Hasil evaluasi penjual menyerahkan produk.....	127
Gambar 4. 74 Hasil evaluasi penjual menerima dana ketika pembeli konfirmasi barang diterima pada distribution point .....	127
Gambar 4. 75 Hasil evaluasi pembeli melihat pesanan.....	128
Gambar 4. 76 Hasil evaluasi pembeli melihat pembayaran.....	128
Gambar 4. 77 Hasil evaluasi pembeli melihat daftar penjualan.....	129

## DAFTAR TABEL

Tabel 3. 1 Tabel Hasil Wawancara .....	21
Tabel 3.2 Kebutuhan pengguna .....	23
Tabel 3. 3 Kebutuhan Fungsional .....	23
Tabel 3.4 Jadwal Kegiatan .....	25
Tabel 3. 5 <i>Usecase Description</i> Membuat Akun Baru .....	44
Tabel 3. 6 <i>Usecase Description</i> Membuka Toko.....	44
Tabel 3. 7 <i>Usecase Description</i> Mengelola Produk.....	45
Tabel 3. 8 Mengelola saldo .....	46
Tabel 3. 9 <i>Usecase Description</i> Mengelola Etalase.....	47
Tabel 3. 10 <i>Usecase Description</i> Melakukan Pemesanan .....	48
Tabel 3. 11 <i>Usecase Description</i> Melakukan Pembelian.....	49
Tabel 3. 12 <i>Usecase Description</i> Melakukan Penjualan.....	50
Tabel 3. 13 <i>Usecase Description</i> Pengolahan Laporan Pemesanan .....	51
Tabel 3. 14 <i>Usecase Description</i> Pengolahan Laporan Pembelian.....	51
Tabel 3. 15 <i>Usecase Description</i> Pengolahan Laporan Penjualan.....	52
Tabel 3. 16 Desain uji coba login dan membuat akun baru .....	79
Tabel 3. 17 Desain uji coba pembeli melakukan pemesanan.....	79
Tabel 3. 18 Desain uji coba pembeli melakukan pembelian.....	79
Tabel 3. 19 Desain uji coba penjual melakukan penjualan .....	80
Tabel 3. 20 Desain uji coba penjual membuka toko .....	80
Tabel 3. 21 Desain uji coba penjual mengelola etalase .....	80
Tabel 3. 22 Desain uji coba penjual mengelola produk .....	80
Tabel 3. 23 Desain uji coba pengolahan laporan pemesanan dari pembeli .....	81

Tabel 3. 24 Desain uji coba pengolahan laporan pembayaran dari pembeli.....	81
Tabel 3. 25 Desain uji coba laporan penjualan dari penjual .....	81
Tabel 4. 1 Kebutuhan <i>Software</i> dan <i>Hardware</i> .....	82
Tabel 4.2 Uji coba login dan membuat akun .....	102
Tabel 4. 3 Uji coba pembeli melakukan pemesanan.....	105
Tabel 4. 4 Uji coba pembeli melakukan pembelian.....	107
Tabel 4. 5 Uji coba penjual melakukan penjualan .....	108
Tabel 4. 6 Uji coba membuka toko .....	110
Tabel 4. 7 Uji coba penjual mengelola etalase.....	111
Tabel 4. 8 Uji coba penjual mengelola produk .....	114
Tabel 4. 9 Uji coba laporan riwayat pemesanan .....	117
Tabel 4. 10 Uji coba laporan riwayat pembayaran .....	118
Tabel 4. 11 Uji coba laporan riwayat penjualan .....	119



## **BAB I**

### **PENDAHULUAN**

#### **1.1. Latar Belakang**

Saat ini Stikom Surabaya telah memiliki aplikasi *Marketplace* dalam bentuk web, *Marketplace* web ini sudah memiliki fitur-fitur utama yang diperlukan untuk dapat melakukan transaksi jual beli. Fitur-fitur tersebut seperti penambahan produk, memasukkan barang ke keranjang, melakukan proses *checkout*, melakukan proses pembayaran, dan memproses pesanan. *Marketplace* Stikom Surabaya ini menggunakan jenis *marketplace* global, yaitu *marketplace* yang menjual berbagai macam jenis produk yang tidak memiliki keterkaitan dengan produk lainnya.

Semua program studi di Stikom Surabaya menyelenggarakan mata kuliah kewirausahaan, yang mana mengharuskan mahasiswa menciptakan produk yang memiliki nilai jual. Produk yang diciptakan dari mata kuliah wirausaha itu nantinya akan dipamerkan di lobi kampus pada setiap ujian akhir semester. Setelah acara pameran ataupun mata kuliah selesai, produk yang mahasiswa ciptakan hanya sebatas tugas kuliah saja dan tidak berlanjut. Disisi lain, dosen Stikom Surabaya juga memiliki grup riset dan penelitian yang menghasilkan karya.

Untuk dapat mengatasi permasalahan diatas perlu adanya platform yang dapat menampung karya mahasiswa dan dosen seperti *marketplace* agar karya ataupun produk yang telah diciptakan tetap berlanjut. *Marketplace* merupakan kegiatan menyediakan tempat kegiatan usaha berupa toko internet

(Ridhoni, 2018). Salah satu keuntungan *marketplace* adalah tidak adanya biaya sewa, penjual hanya perlu mendaftar sebagai penjual untuk dapat mempromosikan dagangannya. Dengan jumlah *marketplace* yang semakin menjamur saat ini, penjual dapat menempatkan produknya di manapun tanpa harus menyewa lapak. Semakin banyak *marketplace* yang digunakan penjual tentunya akan memperluas jangkauan pemasaran produk.

*Marketplace* merupakan pasar yang disediakan secara online, dimana menjadi tempat bertemunya antara penjual dan pembeli secara elektronik sehingga dapat meminimalisir kelemahan pada sistem transaksi tradisional (Mansur, 2015). Kegiatan yang dapat dilakukan penjual pada *marketplace* seperti manajemen toko, manajemen produk, manajemen etalase dan melakukan proses konfirmasi pemesanan. Sedangkan kegiatan yang dapat dilakukan pembeli seperti melakukan pencarian produk, proses pembelian, pembayaran.

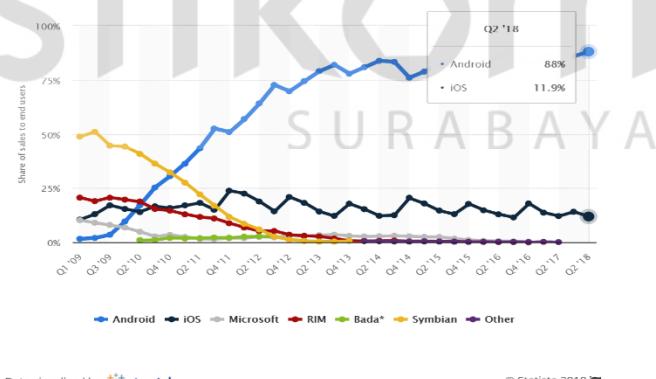
Proses transaksi dapat terjadi setiap saat membuat penjual harus segera memproses pesanan pembeli. Penjual membutuhkan aplikasi yang dapat melakukan manajemen toko, manajemen produk, manajemen etalase dan proses konfirmasi pemesanan. Sedangkan pembeli membutuhkan aplikasi yang dapat melakukan proses pencarian produk, proses pembelian, pembayaran.

*Marketplace* ini juga menyediakan metode pembayaran dengan saldo yang berada pada setiap kartu *Radio Frequency Identification* (RFID) mahasiswa, dosen, ataupun karyawan. Pengguna RFID yang memiliki *smartphone* dengan fitur *Near-field communication* (NFC) juga dapat

melakukan login aplikasi dan melakukan transaksi pembayaran dengan cara menempelkan kartu pada *smartphone* mereka. Untuk mengisi saldo pada kartu RFID setiap orang hanya perlu deposit melalui tempat yang telah disediakan.

Aplikasi *Marketplace* Stikom Surabaya berbasis *mobile* dibuat untuk mempermudah dalam pengaksesan *Marketplace* Stikom Surabaya. Basis *mobile* dipilih karena melihat fakta dari semakin banyaknya jumlah *smartphone* yang bertujuan untuk mempermudah perkerjaan manusia, sehingga waktu yang digunakan semakin cepat dan mudah (Muzakir, 2016).

Selain itu, sebelum ditentukan platform apa yang akan ditambahkan agar dapat meningkatkan jangkauan pengguna, maka diperlukan data statistik dari Statista yang dapat menggambarkan pangsa pasar OS *global* dari kuartal pertama 2009 hingga kuartal kedua 2018. Data pangsa pasar yang paling banyak digunakan saat ini dapat dilihat pada gambar dibawah ini.



Gambar 1.1 Pangsa pasar *Operation System* (OS) *global* dari 2009-2018  
(Sumber : Statista)

Dari data statistik tersebut disimpulkan bahwa pada saat kuartal kedua 2018 sebanyak 88% pangsa pasar OS *global* dikuasai oleh Android sementara 11.9% adalah IOS dan sisa 0.1% nya adalah OS lainnya. Dari kesimpulan

tersebut maka telah ditentukan bahwa platform baru yang dikembangkan akan menggunakan *Operation System* Android.

Kesimpulan data statistik di atas menunjukkan pangsa pasar *Operation System* Android lebih banyak dibandingkan dengan jumlah pengguna IOS, sehingga ditentukan aplikasi *Marketplace* Stikom Surabaya yang dibangun berbasis *Android* agar mempermudah penjual atau pembeli dalam melakukan transaksi melalui *smartphone*. Aplikasi ini ditujukan untuk civitas Stikom dengan harapan dapat meningkatkan semangat wirausaha.

### **1.2. Rumusan Masalah**

Berdasarkan latar belakang di atas, dapat dirumuskan masalah bagaimana merancang dan membangun aplikasi *marketplace* Stikom Surabaya berbasis android.

### **1.3. Batasan Masalah**

1. Aplikasi dikembangkan pada sistem operasi berbasis android.
2. Aplikasi memiliki dua pengguna yaitu penjual dan pembeli.
3. Aplikasi ini menggunakan JWT (Json Web Token) sebagai authentikasi.
4. Aplikasi ini hanya melakukan pembayaran melalui deposit saldo RFID.
5. Pengiriman barang menggunakan *distribusi point* di Stikom Surabaya.
6. Email yang digunakan menggunakan domain “@stikom.edu”.
7. Aplikasi ini hanya menampilkan badge dari member premium.
8. Aplikasi ini tidak membahas mengenai pengembalian barang.
9. Aplikasi ini tidak memiliki stok untuk produk.

#### 1.4. Tujuan

Berdasarkan latar belakang dan rumusan masalah di atas, maka tujuan dari penelitian ini adalah untuk merancang dan membangun aplikasi *marketplace* Stikom Surabaya berbasis android.

#### 1.5 Manfaat

- a) Bagi penjual *Marketplace* Stikom Surabaya.
  - 1. Dapat mengelola toko online mereka sendiri.
  - 2. Mempermudah dalam proses transaksi konfirmasi pemesanan.
  - 3. Dapat memonitor status transaksi.
  - 4. Dapat mempromosikan usahanya.
- b) Bagi pembeli *marketplace* Stikom Surabaya.
  - 1. Dapat mempermudah proses transaksi pembelian produk.
  - 2. Dapat menggunakan saldo RFID untuk pembayaran.
  - 3. Dapat mempermudah proses pencarian produk.

#### 1.6. Sistematika Penulisan

Sistematika penulisan yang digunakan dalam penyusunan laporan ini dibedakan dengan pembagian bab sebagai berikut.

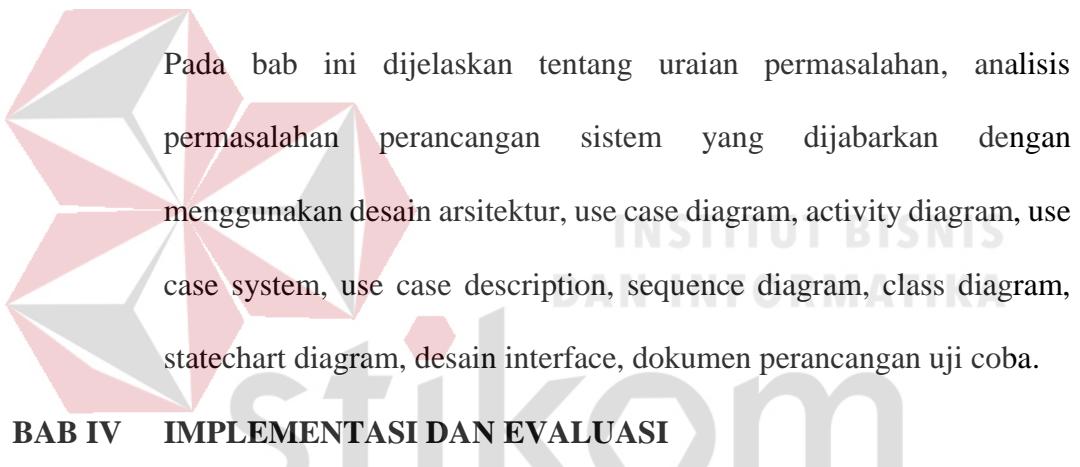
### BAB I PENDAHULUAN

Pada bab ini dijelaskan latar belakang, rumusan masalah, batasan masalah, tujuan, manfaat, dan sistematika penulisan yang berisi penjelasan singkat pada masing-masing bab.

## **BAB II LANDASAN TEORI**

Pada bab ini dijelaskan teori yang digunakan untuk menyelesaikan permasalahan utama yaitu teori mengenai *marketplace*, *Seller Android*, *Android Studio*, *Application Programming Interface (API)*, *Software Development Kit (SDK)*, *JSON WEB TOKEN (JWT)*, *MySQL*, *Unified Modeling Language (UML)*, *Black Box Testing*, *System Development Life Cycle (SDLC)*.

## **BAB III ANALISIS DAN PERANCANGAN SISTEM**



Pada bab ini dijelaskan tentang uraian permasalahan, analisis permasalahan perancangan sistem yang dijabarkan dengan menggunakan desain arsitektur, use case diagram, activity diagram, use case system, use case description, sequence diagram, class diagram, statechart diagram, desain interface, dokumen perancangan uji coba.

## **BAB IV IMPLEMENTASI DAN EVALUASI**

Pada bab ini dijelaskan tentang implementasi dari aplikasi yang dibuat secara keseluruhan dan memberikan penjelasan dari rancangan *input* dan *output* serta melakukan pengujian terhadap aplikasi yang dibuat untuk mengetahui apakah aplikasi tersebut telah dapat menyelesaikan permasalahan yang dihadapi sesuai dengan yang diharapkan.

## **BAB V PENUTUP**

Pada bab ini dijelaskan kesimpulan dari program yang telah selesai dibuat dan saran untuk proses pengembangan selanjutnya.

## **BAB II**

### **LANDASAN TEORI**

#### **2.1. *Marketplace***

Menurut Ridhoni (2018) *Marketplace online* merupakan kegiatan menyediakan tempat kegiatan usaha berupa toko internet di mall internet sebagai tempat *merchant*(penjual) menjual barang dan jasa.

Menurut Sadgotra dan Saputra (2013) *Online Market* adalah segala usaha yang dilakukan untuk melakukan pemasaran suatu produk atau jasa melalui atau menggunakan media internet atau jaringan www, sedangkan *place* sendiri dalam kamus bahasa inggris artinya tempat. Dapat disimpulkan pengertian *Online Market Place* adalah tempat untuk melakukan pemasaran produk atau jasa melalui media internet.

Dari pendapat di atas dapat disimpulkan bahwa, *marketplace* merupakan sebuah tempat yang disediakan oleh pihak ketiga sebagai tempat untuk menjual atau memasarkan produk atau jasa yang menggunakan media internet. *Marketplace* biasa juga disebut sebagai tempat yang disediakan untuk bertemu penjual dan pembeli.

Secara garis besar *marketplace* terdiri dari 3 (tiga) jenis yaitu :

1. *Marketplace vertical*, *marketplace* yang menjual produk dari berbagai sumber namun produk yang mereka jual hanya dari satu jenis. Misalnya, *marketplace* yang hanya menjual produk mobil dari yang bekas maupun yang baru.
2. *Marketplace horizontal*, *marketplace* yang menjual berbagai jenis produk namun semua jenis barang yang dijual saling memiliki

keterkaitan satu sama lainnya. Misalnya, *marketplace* yang menjual produk komputer dan aksesorisnya.

3. *Marketplace global*, *marketplace* yang menjual berbagai macam produk yang bahkan barang satu dengan yang lainnya tidak saling berkaitan. Misalnya, *marketplace global* di indonesia adalah tokopedia, lazada, dan bukalapak.

## 2.2 Seller (penjual/pedagang)

Pedagang adalah pekerjaan yang berhubungan dengan menjual dan membeli barang untuk mendapatkan untung (Sutami, 2005). Pedagang mempunyai berbagai macam jenis. Berikut adalah jenis-jenis pedagang :

1. Pedagang Besar / Distributor / Agen Tunggal

Distributor adalah pedagang yang mendapatkan produk barang dagangan langsung dari tangan pertama produsen.

2. Pedagang Menengah / Agen / Grosir

Agen adalah pedagang yang mendapatkan barang dagangannya dari distributor.

3. Pedagang Eceran

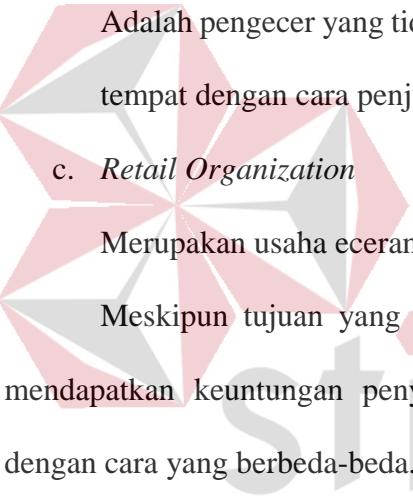
Pengecer adalah pedagang yang menjualan barang secara langsung kepada pemakai akhir. Menurut Kotler (1997) ada tiga jenis bentuk Pengecer, yaitu : *Store retailers*, *non-store retailers*, dan *retail organization*.

- a. *Store Retailers* terdiri atas :

- i. Toko Khusus : adalah pengecer yang menjual barang dengan kategori khusus contohnya toko buku.

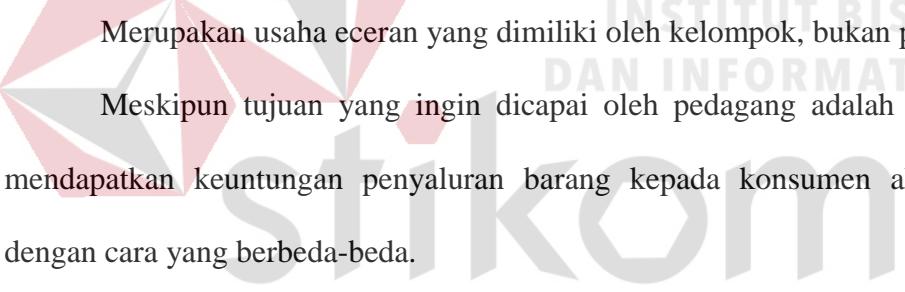
- ii. Toko Serba Ada : adalah pengecer yang mempunyai beberapa lini produk dengan penggolongan barang dan peletekan pada tempat tertentu.
- iii. Pasar Swalayan : adalah pengecer yang beroperasi secara besar, memiliki biaya rendah dengan volume penjualan yang tinggi contohnya Hypermart.
- iv. Toko kebutuhan sehari-hari : adalah pengecer yang relatif kecil dan berada pada wilayah pemukiman.

b. *Non-Store Retailers*



Adalah pengecer yang tidak menjual produknya pada toko atau pada sebuah tempat dengan cara penjualan yang berbeda.

c. *Retail Organization*



Merupakan usaha eceran yang dimiliki oleh kelompok, bukan perorangan. Meskipun tujuan yang ingin dicapai oleh pedagang adalah sama, yaitu mendapatkan keuntungan penyaluran barang kepada konsumen akhir namun dengan cara yang berbeda-beda.

### 2.3 *Android*

Menurut Kasman (2013) *Android* merupakan sistem operasi telepon seluler dan komputer *tablet* layar sentuh berbasis linux. Saat ini sistem operasi *android* sudah memiliki versi Oreo (8.0). *Android* memiliki biaya lisensi yang lebih murah dan sifatnya yang *semi-open source*, dengan dukungan dari berbagai layanan *google*.

### 2.4 *Android Studio*

*Android studio* adalah perangkat lunak pendukung untuk pengembangan aplikasi *android*. Menurut Juansyah (2015) *Android Studio* adalah IDE (*Integrated*

*Development Environment*) resmi untuk pengembangan aplikasi android dan bersifat gratis. *Android Studio* menggantikan aplikasi *eclipse* pada tahun 2013 sebagai IDE resmi untuk pengembangan aplikasi *android*.

*Android studio* mempunyai banyak fitur-fitur baru dibandingkan dengan *Eclipse* sebagai berikut :

1. Menggunakan *Gradle-based build system* yang fleksibel.
2. Bisa mem-build multiple APK.
3. Template support untuk *Google Services* dan berbagai macam tipe perangkat.
4. *Layout editor* yang lebih bagus.

*Built-in support* untuk *Google Cloud Platform*, sehingga mudah untuk integrasi dengan *Google Cloud Messaging* dan *App Engine*. *Import library* langsung dari *Maven repository* dan masih banyak lagi lainnya.

## 2.5 API (*Application Programming Interface*)

*Application Programming Interface* (API) menyediakan abstraksi untuk permasalahan dan menentukan bagaimana klien harus berinteraksi dengan komponen perangkat lunak yang menerapkan solusi untuk permasalahan tersebut. Komponen itu sendiri biasanya didistribusikan sebagai pustaka perangkat lunak yang memungkinkan untuk digunakan dalam beberapa aplikasi. Intinya, API mendefinisikan blok bangunan yang dapat digunakan kembali dan memungkinkan potongan modular fungsionalitas untuk dimasukkan ke dalam aplikasi pengguna akhir (Reddy, 2011).

API juga dapat dijelaskan dengan menganalogikan apabila akan dibangun suatu rumah. Dengan menyewa kontraktor yang menangani bagian yang berbeda-beda, pemilik rumah dapat memberikan tugas yang perlu dilakukan oleh kontraktor tanpa harus mengetahui bagaimana cara kontraktor menyelesaikan tugas tersebut.

Dari analogi tersebut, rumah merupakan *software* yang akan dibangun, dan kontraktor merupakan API yang mengerjakan bagian tertentu dari *software* tersebut tanpa harus diketahui bagaimana prosedur dalam melakukan pekerjaan tersebut.

API dapat diklasifikasikan menjadi beberapa kategori, diantaranya :

- 1) *Operating System*, API yang digunakan untuk fungsi dasar yang dapat dilakukan oleh komputer. Seperti proses *input/output*, eksekusi program.
- 2) *Programming Languages*, API yang digunakan untuk memperluas kapabilitas dalam melakukan eksekusi terhadap suatu bahasa pemrograman.
- 3) *Application Services*, API yang digunakan untuk mengakses data dan layanan yang disediakan dari suatu aplikasi.
- 4) *Application Services*, API yang digunakan untuk mengakses data dan layanan yang disediakan dari suatu aplikasi.
- 5) *Infrastructure Services*, Digunakan untuk mengakses infrastruktur dari suatu komputer. Infrastruktur di sini adalah komputer beserta *peripheral* seperti *storage*, aplikasi, dan lain-lain.
- 6) *Web Services*, API yang digunakan untuk mengakses *content* dan layanan yang disediakan oleh suatu *web application*.

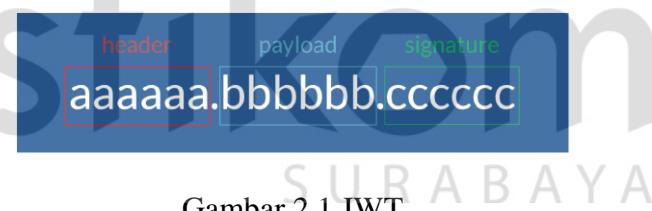
## 2.6 Software Development Kit (SDK)

Menurut Safaat (2012) Android SDK adalah tools API yang diperlukan untuk mulai mengembangkan aplikasi pada platform android menggunakan bahasa pemograman java.

SDK merupakan satu set perkakas pengembangan software yang digunakan dalam pembuatan aplikasi dalam paket software tertentu. SDK juga merupakan tools bagi para programmer yang ingin mengembangkan aplikasi berbasis google android.

## 2.7 JSON WEB TOKEN (JWT)

*JSON Web Token (JWT)* adalah keamanan berbasis *JavaScript Object Notation (JSON)* *token encoding* yang memungkinkan identitas dan keamanan informasi untuk dibagikan di seluruh domain keamanan. Hal ini memungkinkan *client* untuk mendapatkan *token* dengan memberikan *username* dan *password* mereka. Sebuah *token* umumnya diterbitkan oleh penyedia layanan dan dikonsumsi oleh pihak yang mengandalkan konten untuk mengidentifikasi subjek *token* dengan tujuan yang berhubungan dengan keamanan. JWT adalah format representasi yang sederhana ditujukan untuk ruang yang dapat digunakan seperti *HTTP (Header Authorization)* dan parameter *URI Request*. JWT adalah suatu cara merepresentasikan klaim yang akan ditransfer antara dua pihak. Kelebihan yang ditawarkan oleh JWT adalah *stateless*. JWT memungkinkan pengguna untuk berbagi, menggunakan kembali dan kontrol akses ke sumberdaya (Tanaem, Manongga, & Iriani, 2016).



Gambar 2.1 JWT

*JSON Web Token* terdiri dari tiga bagian yang dipisahkan oleh titik “.” (gambar 2.1) yaitu *Header*, *Payload* dan *Signature*. *Header* biasanya terdiri dari dua bagian: jenis *token*, yaitu JWT, dan algoritma *hashing* seperti *hash-based message authentication code (HMAC)* dan *secure hash algorithm (SHA256)*.

```
{
  "alg": "HS256",
  "typ": "JWT"
}
```

Gambar 2.2 JWT Header

Bagian kedua dari JWT adalah *payload*, yang berisi klaim. Klaim adalah pernyataan tentang suatu entitas (biasanya, pengguna) dan metadata tambahan seperti pada gambar 2.2.

```
{
  "iss": "scotch.io",
  "exp": 1300819380,
  "name": "Chris Sevilleja",
  "admin": true
}
```

Gambar 2.3 JWT PAYLOAD

Bagian ketiga dan terakhir dari kami *JSON Web Token* akan menjadi *signature*. *Signature* ini terdiri dari *hash* dari komponen-komponen yaitu *header*, *payload* dan kunci rahasia. Berdasarkan komponen yang terdapat pada JWT, akan di *sign-in* dengan kunci rahasia seperti yang dituangkan pada gambar 2.3.

```
HMACSHA256(
  base64UrlEncode(header) + "." +
  base64UrlEncode(payload),
  "secret"
)
```

Gambar 2.4 JWT Signature.

## 2.8 MySQL

Menurut Raharjo (2011) MySQL merupakan *software relational database management system* (RDBMS) yang dapat mengelola database dengan sangat cepat, dapat menampung data dalam jumlah sangat besar, dapat diakses oleh banyak user (*multi-user*), dan dapat melakukan suatu proses secara sinkron atau berbarengan (*multi-threaded*).

Menurut Haryanti dan Irianto (2011) *Mysql* merupakan *database* yang digunakan oleh situs-situs terkemuka di internet untuk menyimpan datanya. *Software database* MySQL kini dilepas sebagai software manajemen *database* yang *open source*.

Dari kedua pendapat di atas dapat diambil kesimpulan bahwa, Mysql merupakan *software* yang dapat digunakan untuk mengelola *database* dan Mysql juga digunakan untuk menyimpan data dalam jumlah yang sangat besar. *Software database* MySQL bekerja menggunakan bahasa SQL (*Structure Query Language*) yang merupakan bahasa *standart* untuk memanipulasi database. MySQL merupakan database yang populer dan bersifat *open source*.

## 2.9 UML (*Unified Modeling Language*)

Menurut Gata (2013) *Unified Modeling Language* (UML) merupakan bahasa spesifikasi standar yang digunakan untuk mendokumentasikan, menspesifikasi dan membangun perangkat lunak. UML merupakan metodologi dalam pengembangan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan sistem.

Tools yang digunakan dalam perancangan berorientasi objek berbasiskan UML adalah sebagai berikut :

- 1) Use Case Diagram, merupakan pemodelan yang digunakan untuk menggambarkan kelakukan (*behavior*) sistem informasi yang akan dibuat. Use case digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut.

Simbol simbol yang terdapat pada use case diagram yaitu :

- i. *Use Case* menggambarkan fungsionalitas yang disediakan sistem sebagai unit yang bertukar pesan antar unit dengan aktor dan dinyatakan dengan kata kerja.
- ii. *Actor* adalah abstraksi dari orang atau sistem lain yang mengaktifkan fungsi dari target sistem.

iii. Asosiasi antara aktor dan *use case*, digambarkan dengan garis tanpa panah yang mengindikasikan siapa yang meminta interaksi secara langsung.

2) Diagram Aktivitas (*Activity Diagram*), Menggambarkan aliran kerja atau aktivitas dari sebuah sistem atau proses bisnis. Simbol-simbol yang digunakan dalam *Activity Diagram* yaitu :

- i. Start Point, yang merupakan awal aktivitas,
- ii. End Point, akhir aktivitas,
- iii. *Activities*, menggambarkan proses atau kegiatan bisnis
- iv. Percabangan, digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel atau menggabungkan dua kegiatan menjadi satu yaitu penggabungan.
- v. *Decision Points*, menggambarkan pilihan untuk pengambilan keputusan.
- vi. *Swimlane*, pembagian *activity diagram* untuk menunjukkan *stakeholder* yang melakukan aktivitas.

3) Diagram Urutan (*Sequence Diagram*), merupakan diagram yang digunakan untuk menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek. Simbol yang digunakan pada *sequence diagram* yaitu:

- i. *Entity class*, merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data.
- ii. *Boundary class*, berisi kumpulan kelas yang menjadi interfaces atau interaksi antara satu atau lebih aktor dengan sistem seperti tampilan form entry dan form cetak.

- iii. *Control class*, suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab terhadap entitas.
- iv. *Message*, simbol mengirim pesan antar *class*.
- v. *Activation*, mewakili sebuah eksekusi operasi dari objek.

4) *Class Diagram* (Diagram Kelas), Merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas yang juga memperlihatkan aturan dan tanggung jawab entitas yang menentukan perilaku sistem. *Class diagram* juga menunjukkan atribut dan operasi dari sebuah kelas dan batasan yang berhubungan dengan objek yang dikoneksikan. *Class diagram* meliputi : kelas, relasi, atribut, operasi dan *visibility*. Hubungan antar kelas mempunyai keterangan disebut dengan Kardinalitas.

## 2.10 *Black Box Testing*

Menurut Luthfi (2017), *black box testing* bertujuan agar sistem terbebas dari *bugs* dan *error*. *Black box testing* berfokus untuk mengetahui apakah semua fungsi pada sistem telah berjalan semestinya sesuai dengan kebutuhan.

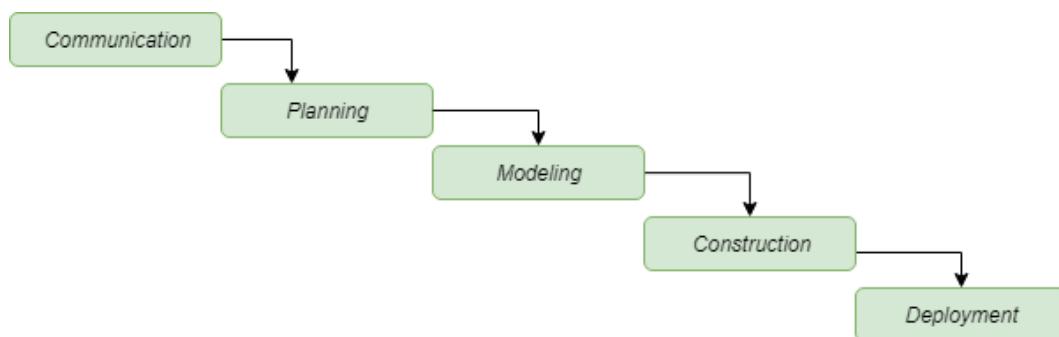
Menurut Romeo (2003), *black box testing* merupakan *testing* yang dilakukan tanpa pengetahuan detil struktur internal dari sistem atau komponen yang dites. *Black box testing* juga disebut sebagai *behavioral testing*, *specification-based testing*, *input/output testing* atau *functional testing*. *Black box testing* berfokus pada kebutuhan fungsional pada *software*, berdasarkan pada spesifikasi kebutuhan dari *software*.

Berdasarkan dua pendapat diatas maka dapat disimpulkan bahwa *black box testing* merupakan pengujian yang dilakukan hanya dengan mengamati hasil eksekusi melalui data uji, dan memeriksa fungsional dari perangkat lunak tanpa harus memiliki pengetahuan detil mengenai struktur internal dari sistem yang dites.

## 2.11 System Development Life Cycle (SDLC)

SDLC adalah metode yang akan digunakan untuk pembuatan aplikasi *marketplace* Stikom Surabaya, dan bertujuan untuk menghasilkan perangkat lunak yang sesuai atau bahkan melampaui keinginan pengguna. Menurut Dewanto (2004) SDLC merupakan suatu urutan dari beberapa proses secara bertahap di dalam merancang dan mengembangkan sistem yang dikenal juga dengan nama *information system development* menyediakan pendekatan alur perangkat lunak secara terurut, berikut adalah gambaran model *system development life Cycle* (SDLC).

Metode *waterfall* atau disebut juga metode air terjun yang menggambarkan pendekatan yang sistematis dan juga berurutan pada pengembangan perangkat lunak. Pengembangan perangkat lunak dimulai dengan spesifikasi kebutuhan pengguna lalu berlanjut melalui tahapan-tahapan perencanaan (*planning*), permodelan (*modeling*), konstruksi (*construction*), serta penyerahan sistem ke pengguna (*deployment*), yang diakhiri dengan dukungan pada perangkat lunak lengkap yang dihasilkan (Pressman, 2015). Tahapan metode *waterfall* dapat dilihat pada gambar di bawah ini.



Gambar 2.5 SDLC Waterfall

SDLC dengan model *waterfall* memiliki tahapan yang terdiri dari :

## 1. *Communication*

Langkah ini merupakan analisis terhadap kebutuhan perangkat lunak (*software*), dan tahap untuk mengadakan pengumpulan data dengan melakukan pertemuan dengan *customer*, maupun mengumpulkan data-data tambahan baik yang ada di jurnal, artikel, maupun dari internet.

## 2. *Planning*

Proses *planning* merupakan lanjutan dari proses *communication (analysis requirement)*. Tahapan ini akan menghasilkan dokumen *user requirement* atau bisa dikatakan sebagai data yang berhubungan dengan keinginan *user* dalam pembuatan *software*, termasuk rencana yang akan dilakukan.

## 3. *Modeling*

Proses *modeling* ini akan menerjemahkan syarat kebutuhan ke sebuah perancangan *software* yang dapat diperkirakan sebelum dibuat *coding*. Proses ini berfokus pada rancangan struktur data, arsitektur *software*, representasi *interface*, dan detail (algoritma) prosedural. Tahapan ini akan menghasilkan dokumen yang disebut *software requirement*.

## 4. *Construction*

*Construction* merupakan proses membuat kode (*coding*). *Coding* atau pengkodean merupakan penerjemahan desain dalam bahasa yang bisa dikenali oleh komputer. Programmer akan menerjemahkan transaksi yang diminta oleh *user*. Tahapan inilah yang merupakan tahapan secara nyata dalam mengerjakan suatu *software*, artinya penggunaan komputer akan dimaksimalkan dalam tahapan ini. Setelah pengkodean selesai maka akan dilakukan *testing* terhadap sistem yang telah

dibuat. Tujuan testing adalah menemukan kesalahan-kesalahan terhadap sistem tersebut untuk kemudian bisa diperbaiki.

### **5. *Deployment***

Tahapan ini bisa dikatakan final dalam pembuatan sebuah *software* atau sistem. Setelah melakukan analisis, desain dan pengkodean maka sistem yang sudah jadi akan digunakan oleh *user*. Kemudian *software* yang telah dibuat harus evaluasi jika ada kekurangan dan dilakukan pemeliharaan secara berkala.



## BAB III

### ANALISIS DAN PERANCANGAN SISTEM

Metode penelitian yang digunakan untuk membantu dalam pembuatan aplikasi *marketplace* Stikom Surabaya dengan metode *System Development Life Cycle* (SDLC) model *waterfall*. Tahapan yang harus dilakukan sesuai dengan model *waterfall* yaitu :

<i>Communication</i>		
Instrumen	Langkah-langkah	Luaran
Observasi, wawancara	Identifikasi masalah.	Infomasi yang diperlukan, masalah yang terjadi, dan solusi yang akan diberikan.
Observasi, wawancara	Analisis kebutuhan pengguna	Kebutuhan pengguna.
Observasi, wawancara	Identifikasi Data	Data yang diperlukan dalam pembuatan aplikasi.
<i>Planning</i>		
Instrument	Langkah-langkah	Luaran
-	Pembuatan Jadwal Kerja Penelitian	Jadwal Kerja Penelitian
<i>Modeling</i>		
Instrument	Langkah-langkah	Luaran
Draw.io, Astah UML	Perancangan desain proses fungsional.	<ol style="list-style-type: none"> <li>a. Desain arsitektur</li> <li>b. <i>Use case diagram</i></li> <li>c. <i>Activity diagram</i></li> <li>d. <i>Use case system</i></li> <li>e. <i>Use case description</i></li> <li>f. <i>Sequene diagram</i></li> <li>g. <i>Class diagram</i></li> <li>h. <i>Statechart diagram</i></li> </ol>
Draw.io	Perancangan antarmuka pengguna	Desain antarmuka pengguna
Microsoft Word	Perancangan uji coba	Dokumen perancangan uji coba
<i>Construction</i>		
Instrument	Langkah-langkah	Luaran
Android studio	Pembuatan program (coding)	Aplikasi
Android studio	Uji coba program	Hasil uji coba

Gambar 3. 1 Tahapan SDLC

### 3.1 Communication

Tahapan *communication* adalah tahap yang pertama kali dilakukan dalam pembuatan aplikasi dengan model *waterfall*. Tahapan ini dimulai dengan melakukan langkah pengumpulan data dengan instrumen studi pustaka, dan wawancara terhadap staff SSI(solusi sistem informasi) yang akan menghasilkan proses bisnis, permasalahan, peluang dan tujuan.

#### 3.1.1 Identifikasi masalah

##### a. Studi Pustaka

Studi pustaka dilakukan untuk mengumpulkan data dengan membaca dan mempelajari buku literatur, artikel, jurnal, dan informasi lainnya sebagai bahan tinjauan pustaka yang berkaitan dengan penelitian ini.

##### b. Wawancara

Wawancara dilakukan terhadap Staff Solusi Sistem Informasi (SSI) Stikom Surabaya selaku tim pengembang aplikasi web. Wawancara dilakukan agar mendapatkan informasi yang diperlukan, masalah yang terjadi, dan solusi yang akan diberikan. Hasil dari tahap wawancara akan digunakan untuk acuan pengembangan aplikasi marketplace Stikom Surabaya. Berikut beberapa pertanyaan dan jawaban dari wawancara yang dilakukan :

Tabel 3. 1 Tabel Hasil Wawancara

No.	Pertanyaan	Jawaban
1.	Ada berapa jenis pengguna pada aplikasi <i>web marketplace</i> Stikom Surabaya?	Pada aplikasi marketplace terdapat dua jenis pengguna yaitu pembeli dan penjual. Seorang pembeli dapat menjadi penjual dengan syarat sudah membuka toko sebelumnya.
2.	Apa saja yang dapat dilakukan pengguna sebagai pembeli ?	Pengguna sebagai pembeli dapat melihat semua list produk yang ada dan dapat menambahkannya ke keranjang kecuali produk miliknya sendiri(jika punya toko). Pengguna sebagai pembeli juga

		dapat melakukan proses checkout, pembayaran, dan pengambilan barang ke distribution point.
3.	Apa saja yang dapat dilakukan pengguna sebagai penjual ?	Pengguna sebagai penjual dapat melakukan pengelolaan produk miliknya seperti penambahan produk, edit produk, dan hapus produk dengan syarat sudah membuat etalase sebelumnya. Penjual juga dapat memproses penjualan dari pesanan pembeli serta penyerahan barang ke distribution point. Jika penjual memiliki saldo toko dari penjualan yang telah selesai, penjual dapat melakukan proses penarikan dana yang akan dikirim pada rekeningnya.

Dari hasil wawancara yang didapat maka proses bisnis yang terdapat pada aplikasi web marketplace yaitu membuat akun baru, membuka toko, mengelola produk, mengelola etalase, melakukan pemesanan, dan melakukan penjualan. Proses bisnis ini nantinya akan digunakan untuk perancangan aplikasi marketplace Stikom Surabaya berbasis android.

#### c. Observasi

Observasi dilakukan dengan melihat dan mengamati secara langsung kegiatan warga stikom dan lingkungan stikom. Tujuan melakukan observasi adalah untuk menghasilkan data tambahan yang tidak didapatkan selama wawancara.

Berikut temuan yang didapatkan pada saat proses observasi, yaitu :

- 1) Pengguna sebagai penjual dapat melakukan penarikan dana.
- 2) Pengguna sebagai pembeli dapat melihat detail riwayat pemesanan
- 3) Pengguna sebagai pembeli dapat melihat detail riwayat pembelian
- 4) Pengguna sebagai penjual dapat melihat detail riwayat penjualan.

### 3.1.2 Analisis Kebutuhan

Untuk dapat membangun dan mengembangkan aplikasi, diperlukan perancangan spesifikasi aplikasi yang tepat, agar aplikasi yang akan dibuat memiliki deskripsi fungsi yang sesuai dengan yang dibutuhkan oleh pengguna.

#### 1) Kebutuhan pengguna

Tabel 3.2 Kebutuhan pengguna

No	Pengguna	Kebutuhan Sistem
KP.1	Pembeli	Melakukan pembelian barang
KP.2	Penjual	Melakukan penjualan barang

#### 2) Kebutuhan Fungsional

Tabel 3. 3 Kebutuhan Fungsional

No	Kebutuhan Fungsional	No. Kebutuhan Pengguna
KF.1	Membuat akun baru	KP.1
KF.2	Membuka toko	KP.2
KF.3	Mengelola etalase	KP.2
KF.4	Mengelola produk	KP.2
KF.5	Melakukan pemesanan	KP.1
KF.6	Melakukan pembelian	KP.1
KF.7	Melakukan penjualan	KP.2
KF.8	Mengelola saldo	KP.2&KP.1
KF.9	Pengolahan laporan pemesanan	KP.1
KF.10	Pengolahan laporan pembelian	KP.1
KF.11	Pengolahan laporan penjualan	KP.2

#### 3) Kebutuhan non Fungsional

Kebutuhan non fungsional merupakan kebutuhan yang tidak termasuk pada fungsi, seperti :

- Karakteristik sistem
  - Membatasi fungsi pengguna dengan hak akses melalui login.
  - Terdapat proses konfirmasi penyerahan dan pengambilan barang.

- b. Perangkat lunak
  - i. Menggunakan minimal windows 7.
  - ii. Android studio
  - iii. Sqlyog
  - iv. Xampp (apache, mysql, php 5)
  - v. Minimal OS Smartphone android lollipop (API 21)
- c. Perangkat keras
  - i. Minimal memori 1GB untuk smartphone, 2 GB untuk server
  - ii. Minimal penyimpanan tersedia 2GB untuk *smartphone* dan 10GB untuk



### 3.1.3 Identifikasi kebutuhan data

Setelah melakukan tahap identifikasi masalah dan identifikasi pengguna, maka selanjutnya akan dilakukan identifikasi data. Pada aplikasi ini akan dibutuhkan data sebagai berikut :

- a. Data Penjualan
- b. Data Pembelian
- c. Data Toko
- d. Data Produk
- e. Data Saldo
- f. Data Pengguna

## 3.2 Planning

Dalam tahapan ini dilakukan sebuah penjadwalan terhadap aktifitas kegiatan yang dilakukan berupa jadwal pelaksanaan mengenai merancang dan

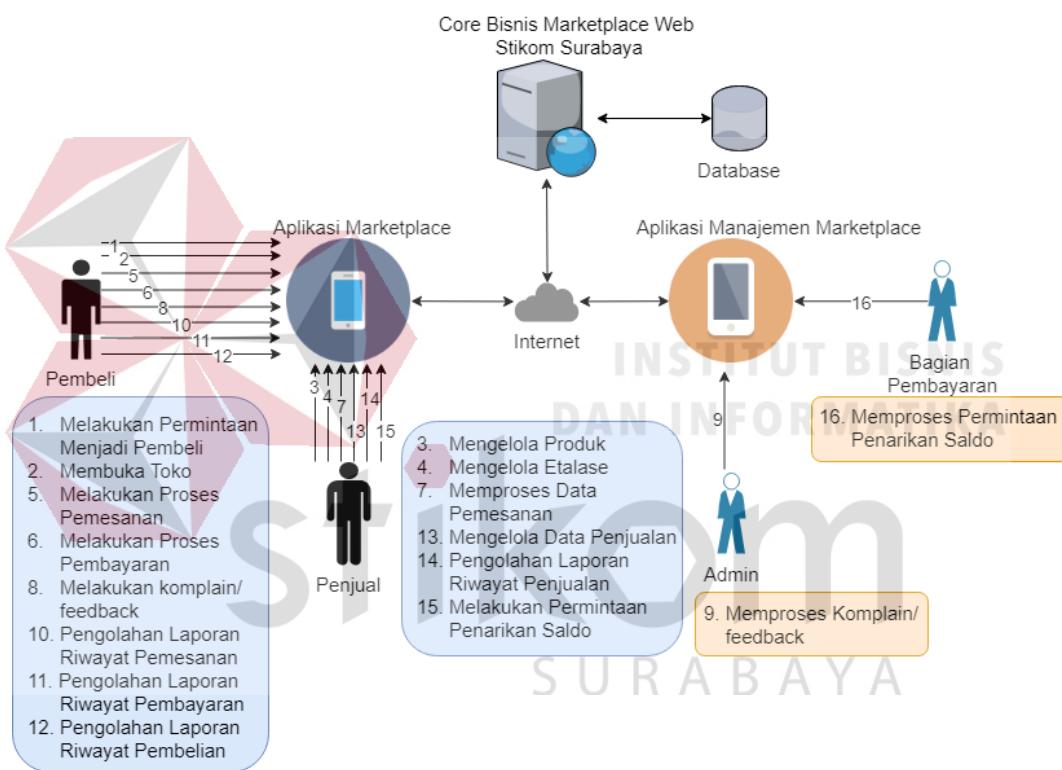
membangun aplikasi *marketplace* Stikom Surabaya berbasis android. Jadwal perancangan yang akan dibuat dapat dilihat pada tabel 3.1 :

Tabel 3.4 Jadwal Kegiatan

### 3.3 Modeling

Tahap selanjutnya adalah tahap desain atau perancangan. Tahap perancangan adalah tahapan untuk merancang aplikasi. Menurut SDLC ada beberapa tahapan yang harus dilakukan. Tahapan yang dilakukan seperti pembuatan Desain arsitektur, *Use case diagram*, *Activity diagram*, *Use case sistem*, *Use case description*, *Sequence diagram*, *Class diagram*, *Statechart diagram*.

#### 3.3.1 Desain Arsitektur

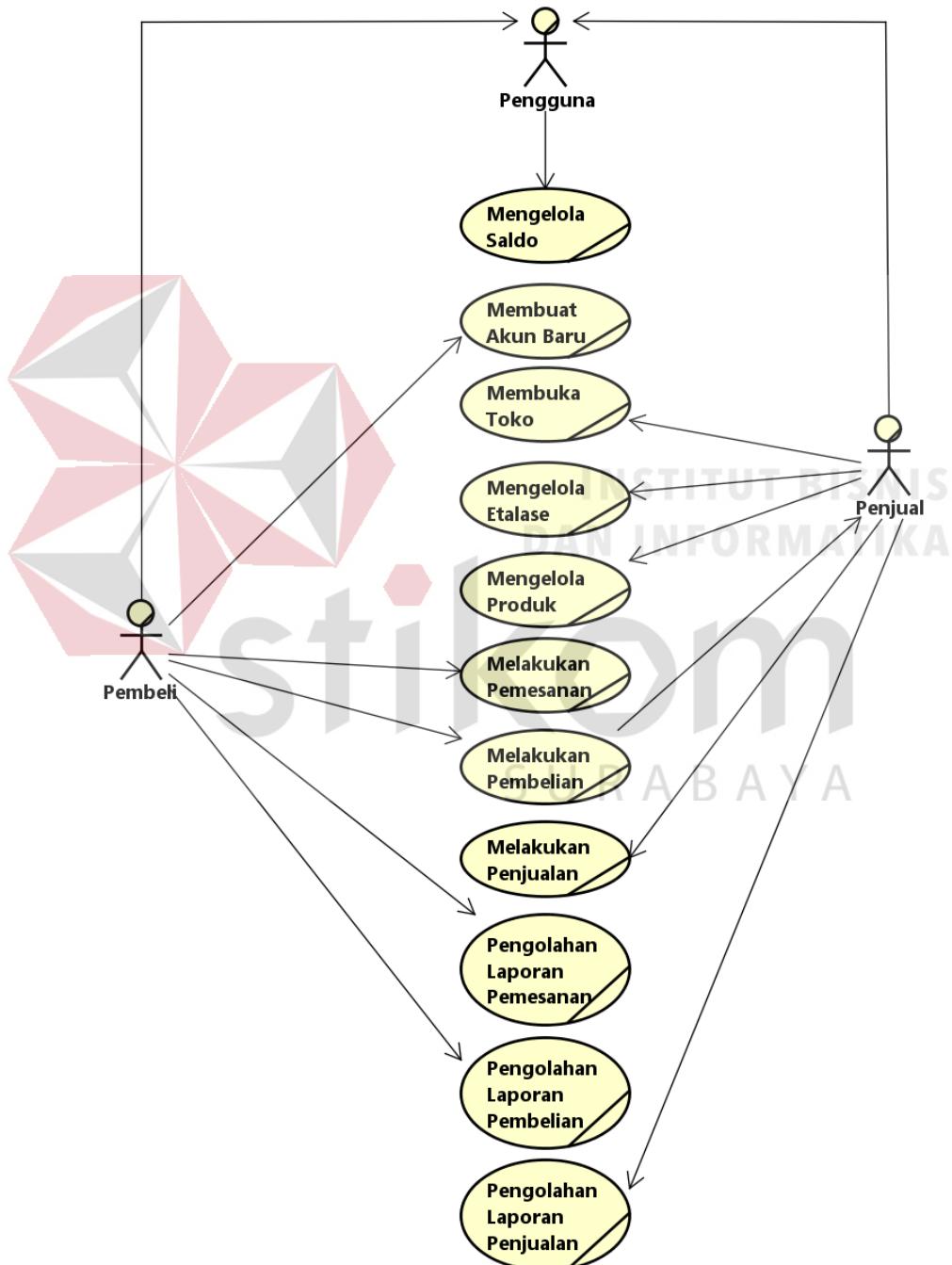


Gambar 3.2 Desain Arsitektur Marketplace Stikom Surabaya

Pada desain arsitektur di atas ketika pengguna login sebagai pembeli, pengguna dapat membuka toko untuk dapat menjadi penjual. Pembeli juga dapat melakukan proses pemesanan, proses pembayaran, pengolahan laporan riwayat pemesanan, pengolahan laporan riwayat pembayaran, pengolahan laporan riwayat pembelian.

Setelah membuat toko maka pengguna dapat bertindak sebagai penjual yang dapat membuka toko, mengelola produk yang akan mereka jual, mengelola etalase dari produk, memproses data pemesanan, mengelola data penjualan, melakukan permintaan penarikan saldo.

### 3.3.2 Use Case bisnis



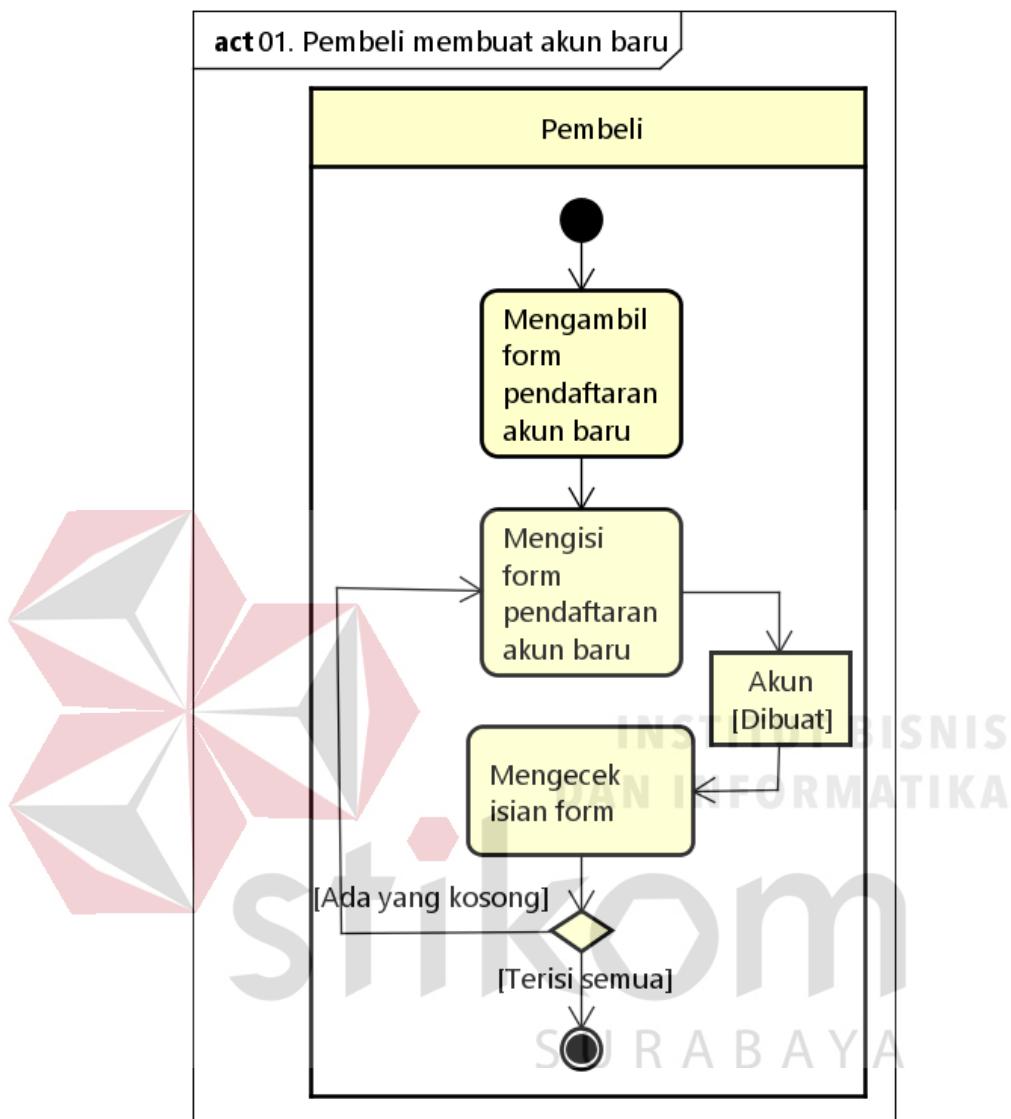
Gambar 3. 3 Use Case bisnis

Proses bisnis yang terjadi pada Marketplace Stikom Surabaya sebagai berikut :

- a. Membuat akun baru, adalah proses yang bertujuan membuat akun baru untuk pembeli agar nantinya dapat digunakan untuk melakukan proses yang lainnya.
- b. Membuka toko, adalah proses pertama yang dilakukan oleh penjual agar dapat menjadi penjual dan dapat melakukan proses pengelolaan produk.
- c. Mengelola produk, adalah proses yang digunakan untuk menambah, mengedit, memperbarui, dan menghapus produk yang ada pada toko yang dimiliki oleh penjual.
- d. Mengelola data saldo, adalah proses yang digunakan untuk mengelola data saldo seperti penarikan saldo.
- e. Mengelola etalase, adalah proses untuk mengatur produk-produk sesuai dengan etalase.
- f. Melakukan pemesanan, adalah proses transaksi yang dilakukan oleh pembeli untuk melakukan proses pemesanan.
- g. Melakukan pembelian, adalah proses yang dilakukan setelah proses pemesanan agar produk yang telah dipesan sebelumnya dapat segera dikirim oleh penjual.
- h. Melakukan penjualan, adalah transaksi yang dilakukan oleh penjual untuk memproses pesanan dari pembeli.
- i. Pengolahan laporan pemesanan dan pengolahan laporan pembelian, adalah riwayat dari transaksi pemesanan dan pembayaran yang telah dilakukan oleh pembeli.
- j. Pengolahan laporan penjualan, adalah riwayat transaksi dari proses penjualan yang dilakukan oleh penjual.

### 3.3.3 Activity diagram

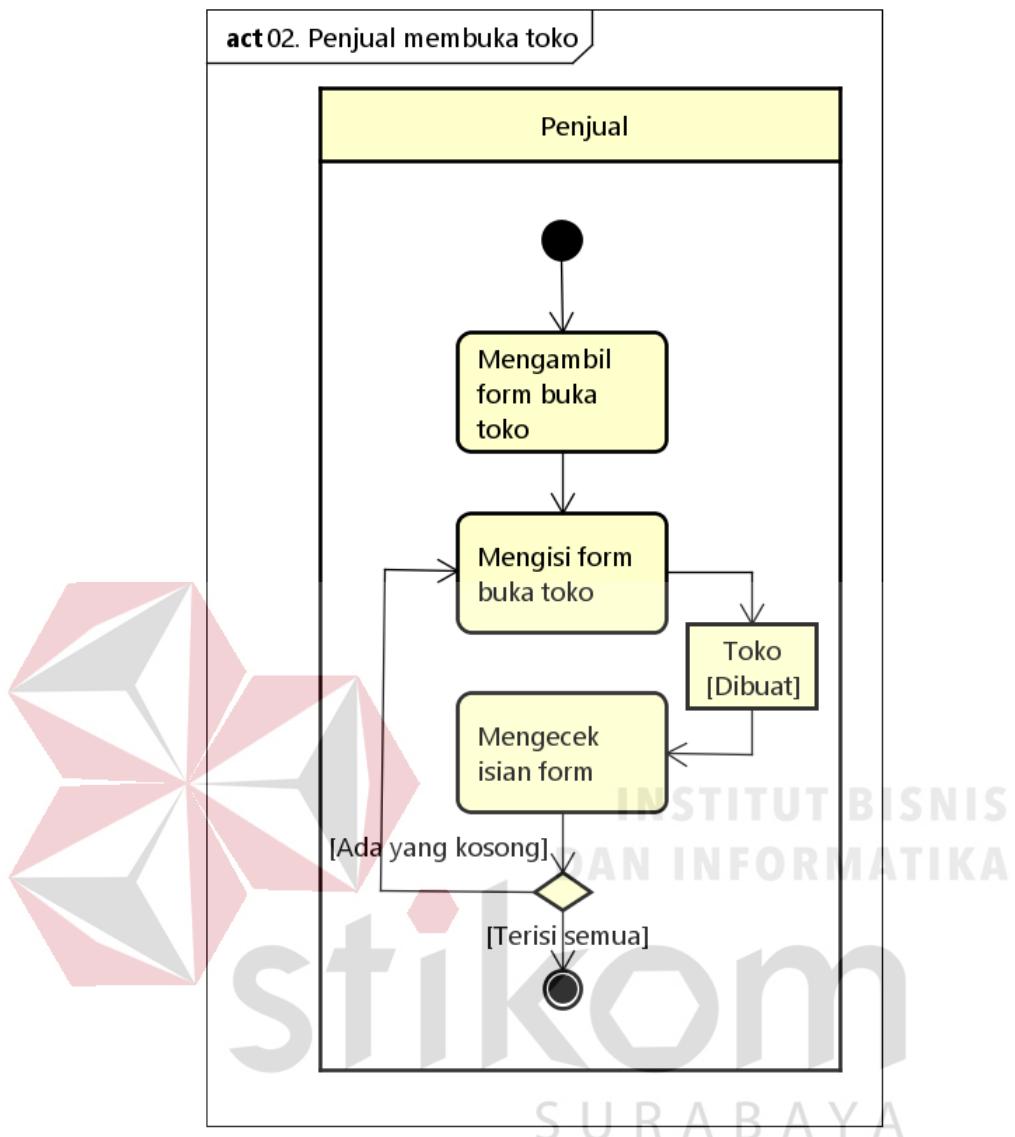
#### a. Pembeli Membuat Akun Baru



Gambar 3. 4 Pembeli membuat akun baru

Saat pembeli membuat akun baru, pembeli akan mengambil dan mengisi form pendaftaran akun baru. Kemudian pembeli mengecek form yang telah diisi. Jika form sudah benar dan tidak ada yang kosong maka, pembuatan akun baru sudah selesai. Apabila masih terdapat data yang kosong atau salah maka, pembeli harus mengisi form kembali.

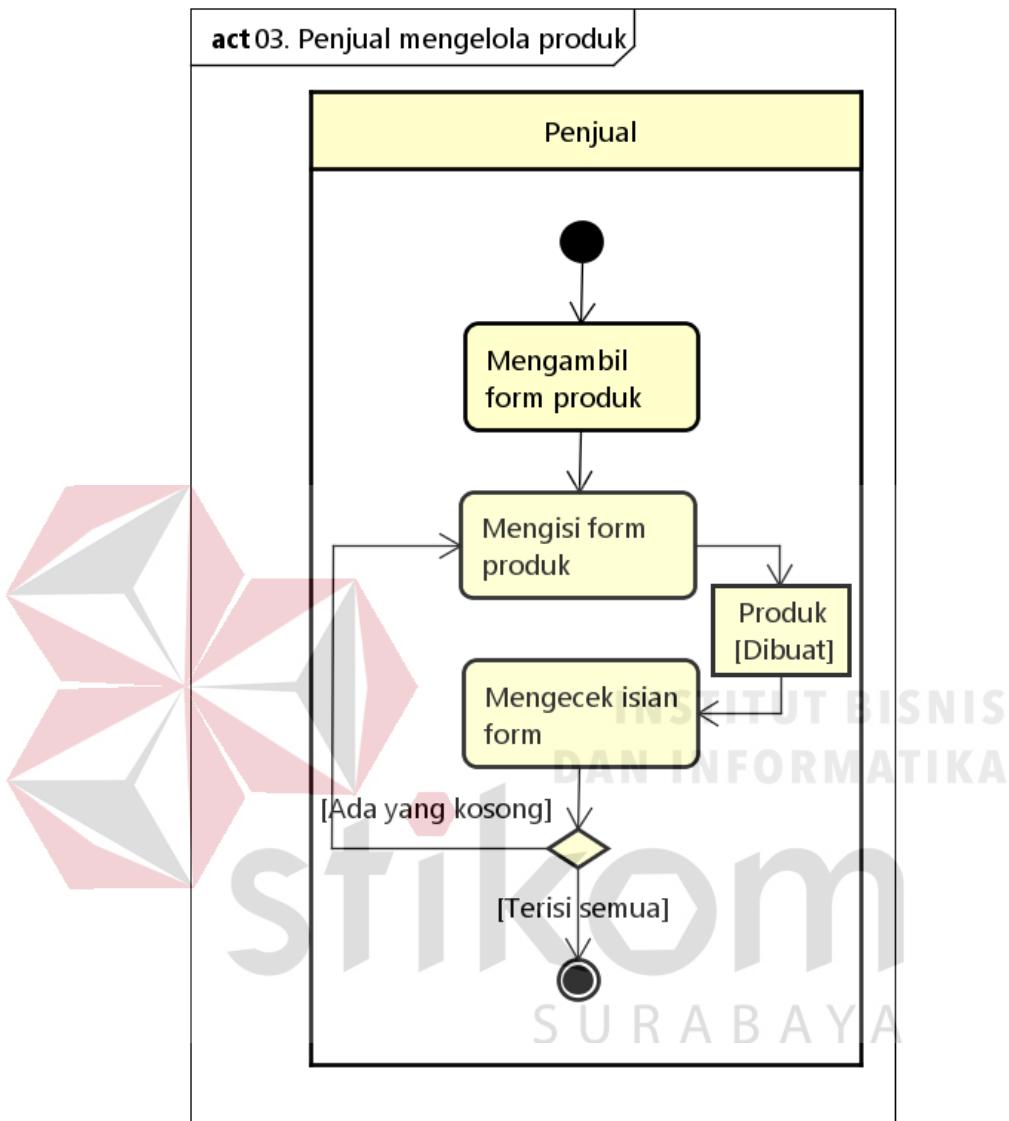
### b. Penjual Membuka Toko



Gambar 3. 5 Penjual membuka toko

Saat penjual ingin membuka toko, pemjual akan mengambil dan mengisi form untuk membuka toko. Kemudian penjual mengecek form yang telah diisi. Jika form sudah benar dan tidak ada yang kosong maka, proses pembuatan toko sudah selesai. Apabila masih terdapat data yang kosong atau salah maka, penjual harus mengisi form kembali.

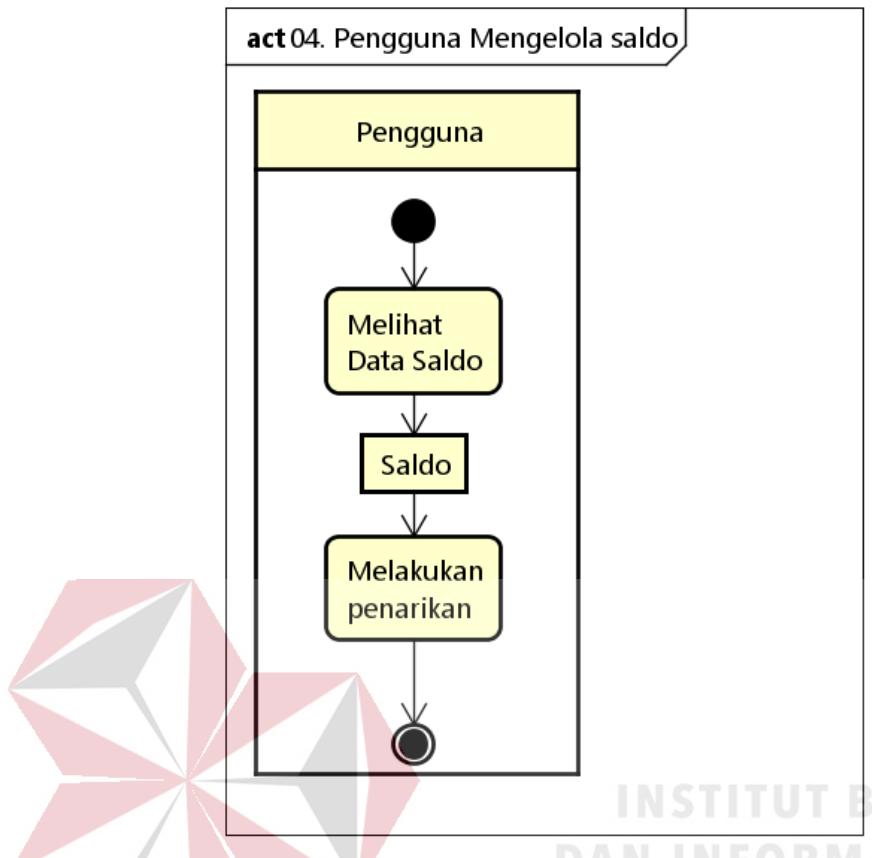
### c. Penjual Mengelola Produk



Gambar 3. 6 Penjual mengelola produk

Saat penjual ingin melakukan pengelolaan produk, penjual akan mengambil dan mengisi form produk. Kemudian penjual mengecek form yang telah diisi. Jika form sudah benar dan tidak ada yang kosong maka, proses pengelolaan produk sudah selesai. Apabila masih terdapat data yang kosong atau salah maka, penjual harus mengisi form kembali.

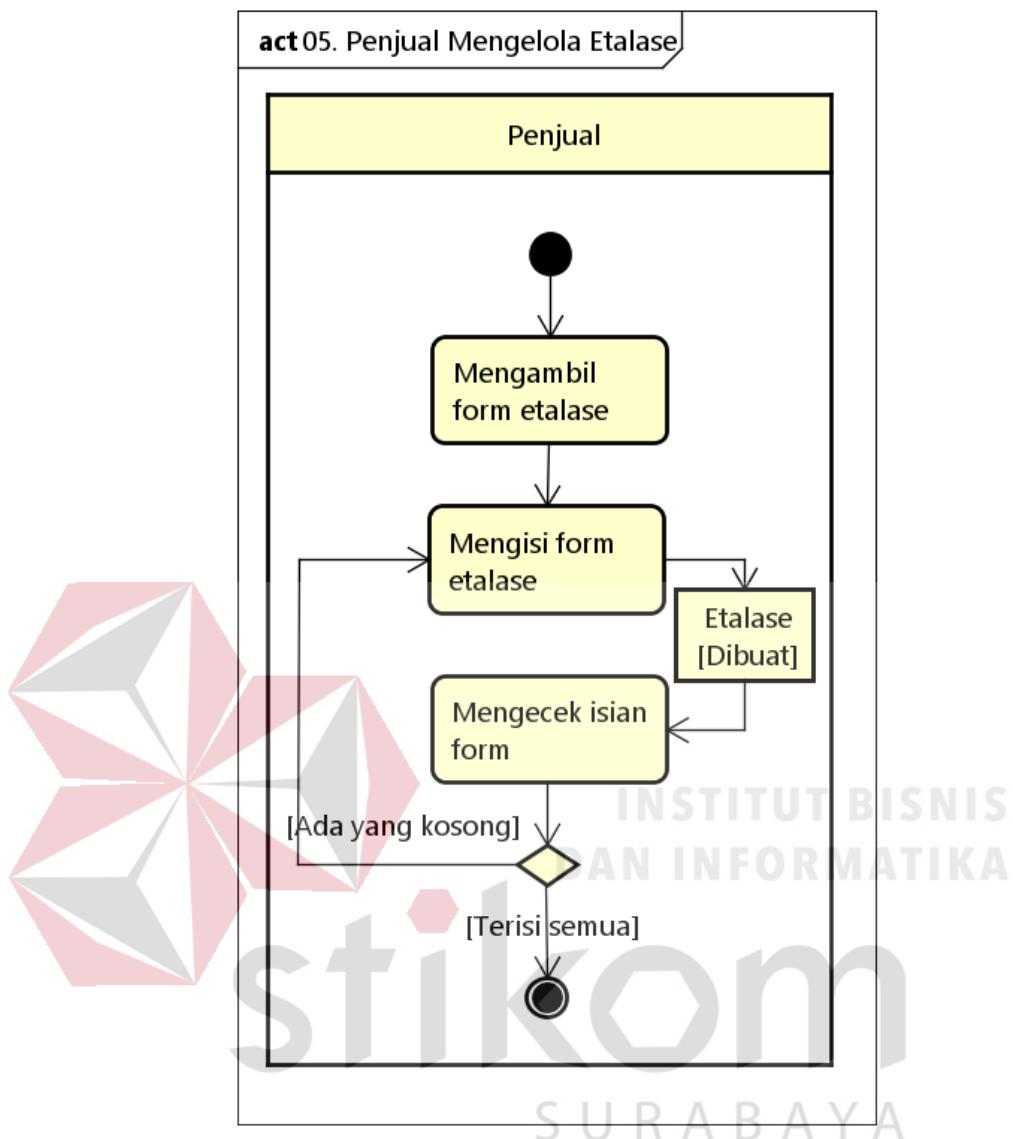
#### d. Pengguna Mengelola Saldo



Gambar 3. 7 Pengguna mengelola saldo

Saat pengguna ingin melakukan pengelolaan saldo, pengguna dapat melihat data saldo. Kemudian pengguna dapat melakukan permintaan penarikan dana sesuai dengan saldo yang dimiliki toko.

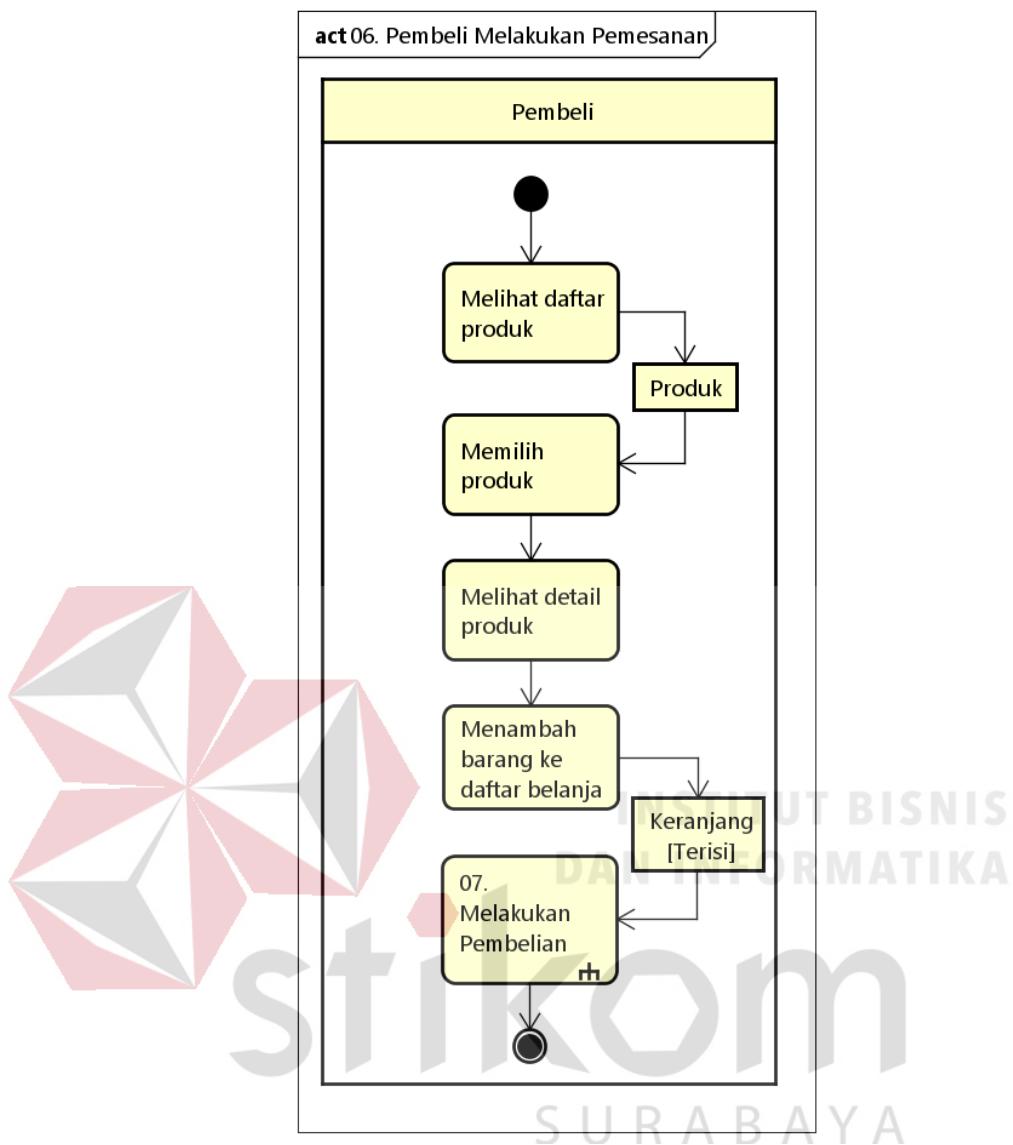
### e. Penjual Mengelola Etalase



Gambar 3. 8 Penjual mengelola etalase

Saat penjual akan melakukan pengelolaan etalase, penjual akan mengambil dan mengisi form etalase. Kemudian penjual mengecek form yang telah diisi. Jika form sudah benar dan tidak ada yang kosong maka, proses pengelolaan etalase sudah selesai. Apabila masih terdapat data yang kosong atau salah maka, penjual harus mengisi form kembali.

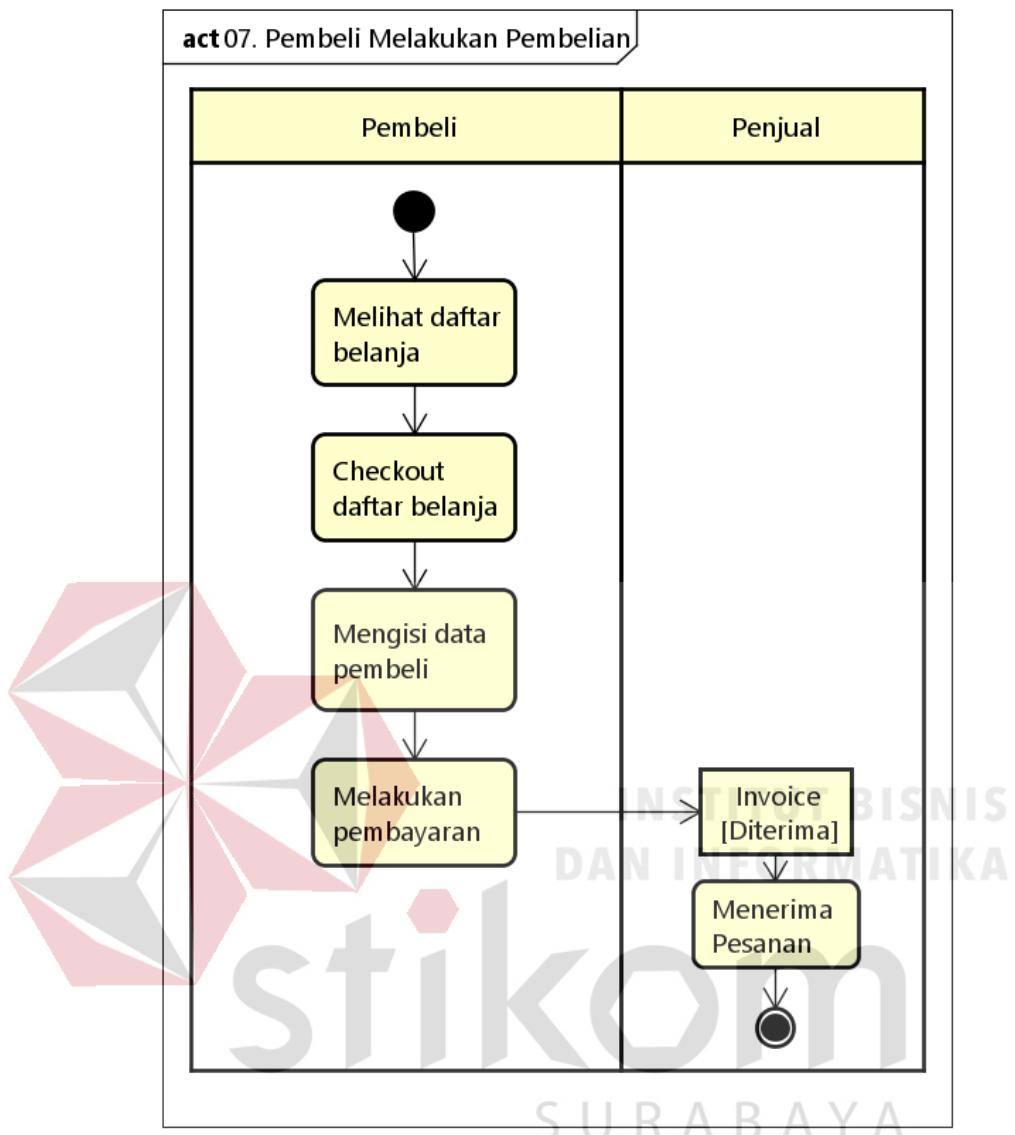
#### f. Pembeli Melakukan Pemesanan



Gambar 3. 9 Pembeli melakukan pemesanan

Saat pembeli akan melakukan pemesanan, penjual akan melihat daftar produk dan memilih produk. Setelah pembeli memilih produk kemudian pembeli akan melihat detail produk. Apabila pembeli tertarik maka, pembeli akan memasukkan barang ke daftar belanja dan kemudian melakukan pembelian.

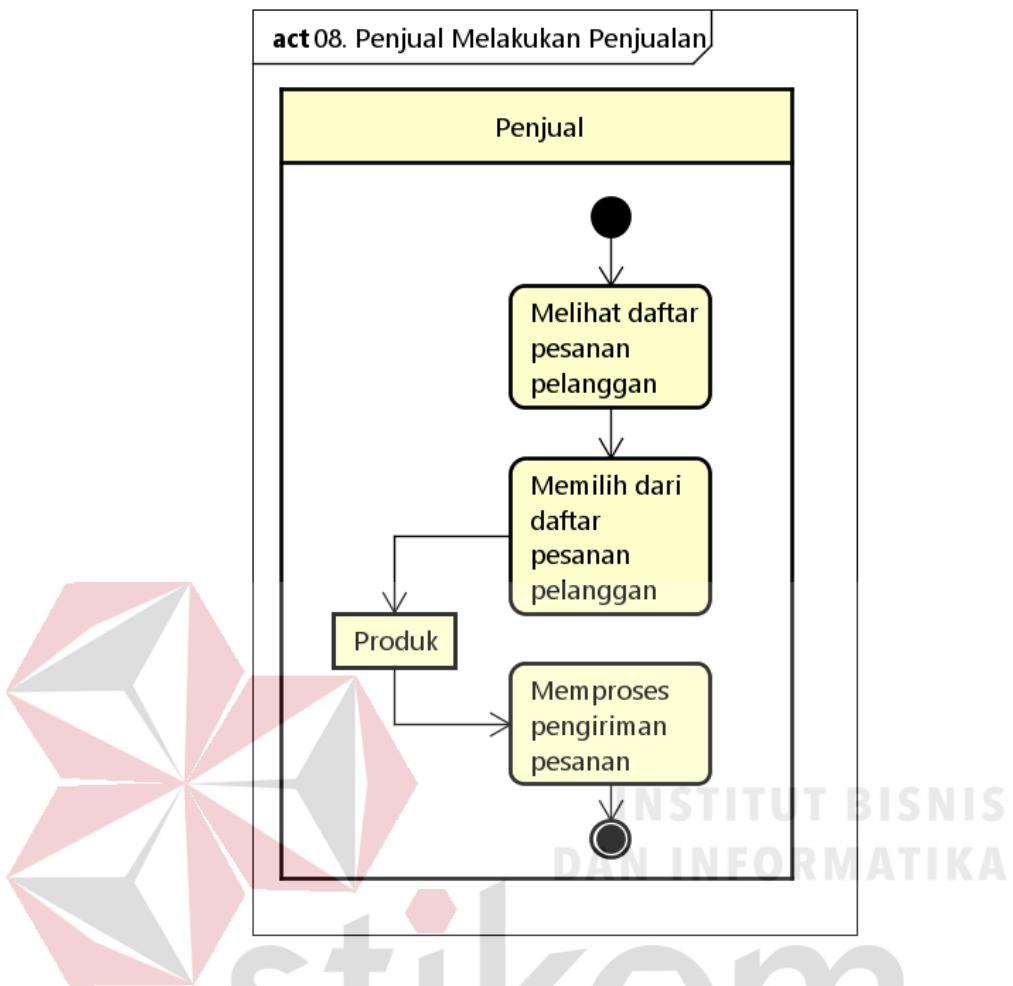
**g. Pembeli Melakukan Pembelian**



Gambar 3. 10 Pembeli melakukan pembelian

Saat pembeli melakukan pembelian, pembeli akan melihat daftar belanja dan melakukan checkout daftar belanja. Kemudian pembeli akan mengisi data pembelian dan melakukan pembayaran.

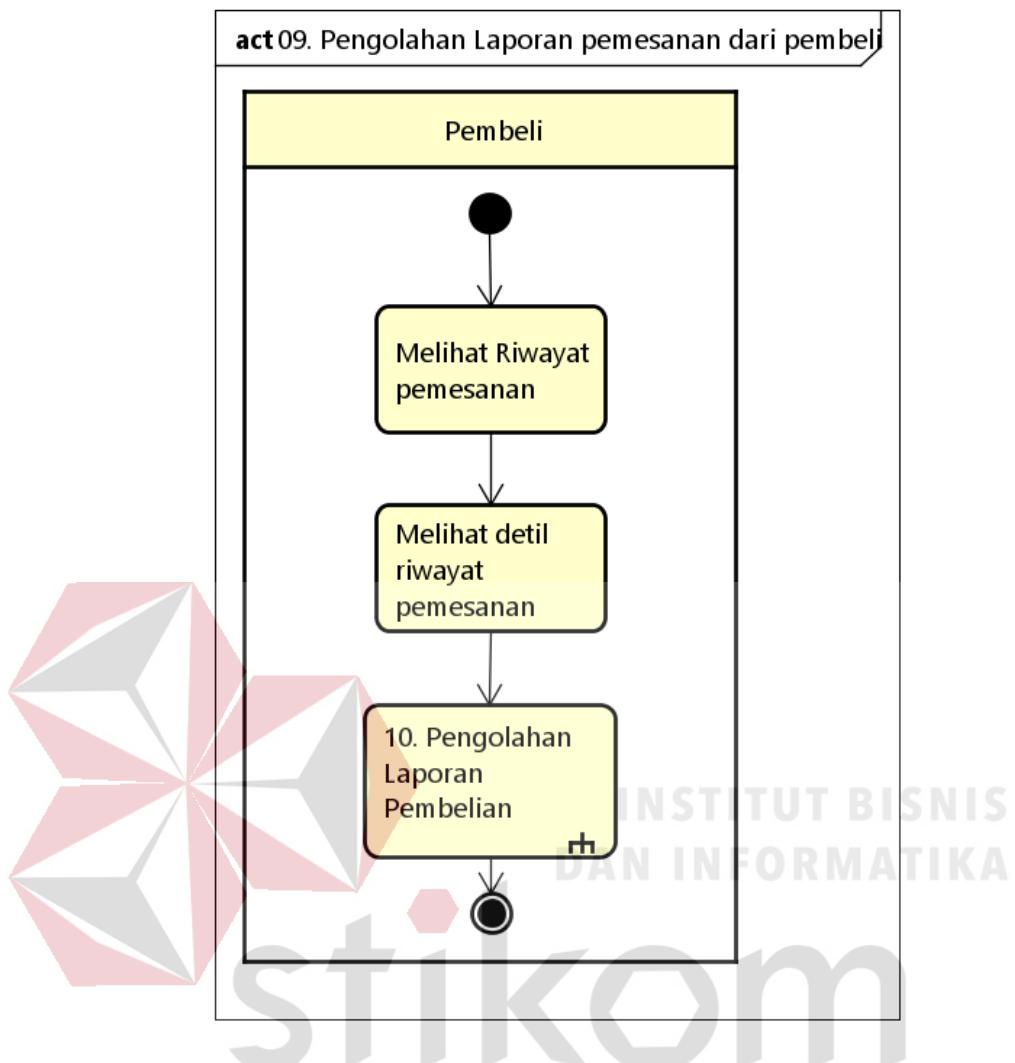
### h. Penjual Melakukan Penjualan



Gambar 3. 11 Penjual melakukan penjualan

Saat penjual akan melakukan penjualan, penjual akan melihat daftar pesanan pelanggan dan memilih daftar dari pesanan pelanggan. Kemudian penjual akan memproses pengiriman pesanan.

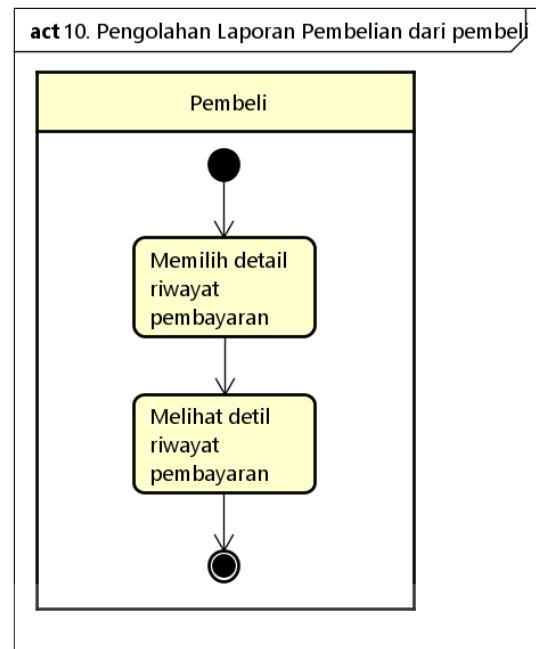
### i. Pengolahan Laporan Pemesanan dari Pembeli



Gambar 3. 12 Pengolahan laporan pemesanan dari pembeli

Saat pembeli melakukan pengolahan laporan pemesanan dari pembeli, pembeli akan melihat riwayat pemesanan dan melihat detail riwayat pemesanan. Kemudian akan dilakukan pengelolaan laporan pembelian.

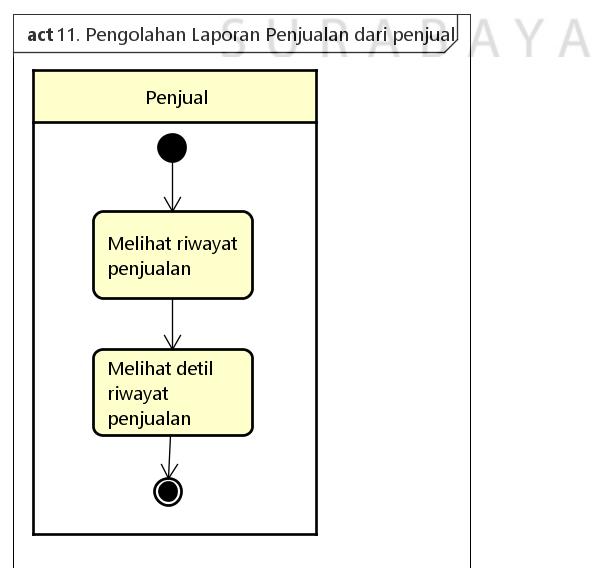
### j. Pengolahan Laporan Pembelian dari Pembeli



Gambar 3. 13 pengolahan laporan pembelian dari pembeli

Saat pembeli melakukan pengolahan laporan pembelian dari pembeli, pembeli akan memilih detail riwayat pembayaran dan melihat detail riwayat pembayaran.

### k. Pengolahan Laporan Penjualan dari Penjual

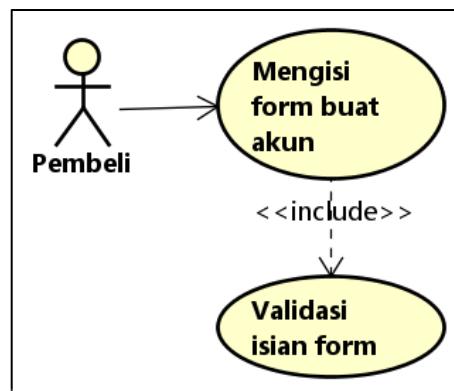


Gambar 3. 14 Pengolahan laporan penjualan dari penjual

Saat penjual akan melakukan pengolahan laporan penjualan dari penjual, penjual akan melihat riwayat penjualan dan melihat detail riwayat penjualan.

### 3.3.4 Usecase sistem

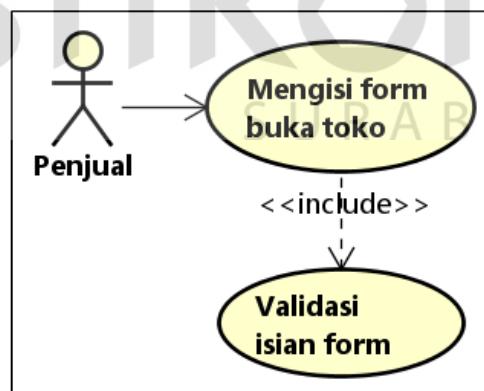
#### a. Pembeli Membuat Akun Baru



Gambar 3. 15 Pembeli membuat akun baru

Usecase sistem pembeli membuat akun baru dilakukan oleh pembeli yang akan mengisi form buat akun. Setiap pembeli mengisi form buat akun pembeli akan melakukan validasi isian form.

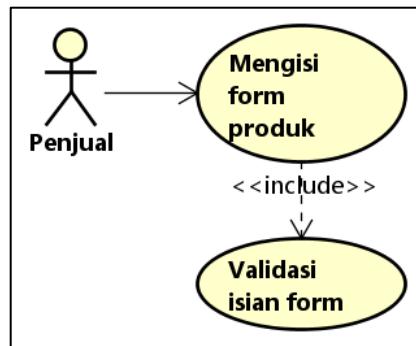
#### b. Penjual Membuka Toko



Gambar 3. 16 Penjual membuka toko

Usecase sistem penjual membuka toko dilakukan oleh penjual yang akan mengisi form buka toko. Setiap penjual mengisi form buka toko penjual akan melakukan validasi isian form.

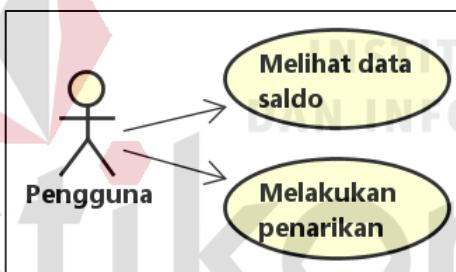
### c. Penjual Mengelola Produk



Gambar 3. 17 Penjual mengelola produk

Usecase sistem penjual mengelola produk dilakukan oleh penjual yang akan mengisi form produk. Setiap penjual mengisi form produk penjual akan melakukan validasi isian form.

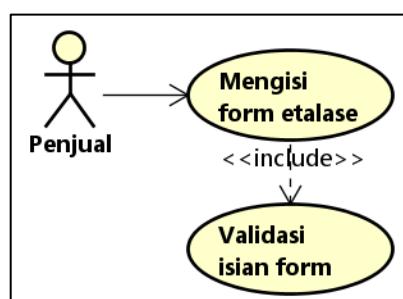
### d. Pengguna Mengelola Data Saldo



Gambar 3. 18 Pengguna mengelola data saldo

Usecase sistem pengguna mengelola data saldo dilakukan oleh pengguna. Pengguna dapat melihat data saldo dan melakukan penarikan.

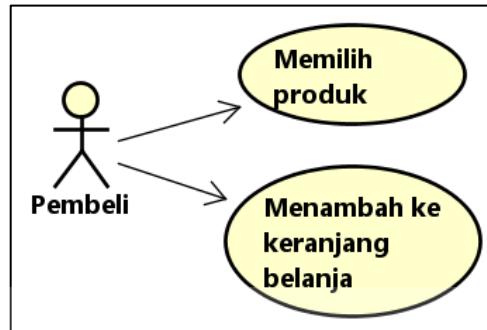
### e. Mengelola Etalase



Gambar 3. 19 Penjual mengelola etalase

Use case sistem mengelola etalase dilakukan oleh penjual yang akan mengisi form etalase. Setiap penjual mengisi form etalase penjual akan melakukan validasi isian form.

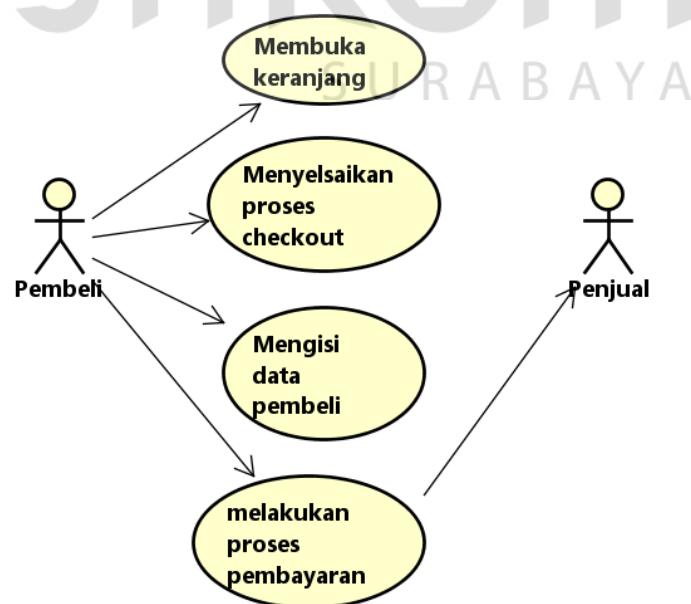
#### f. Pembeli Melakukan Pemesanan



Gambar 3. 20 Pembeli melakukan pemesanan

Use case sistem pembeli melakukan pemesanan dilakukan oleh pembeli. pembeli dapat memilih produk dan menambah ke keranjang belanja.

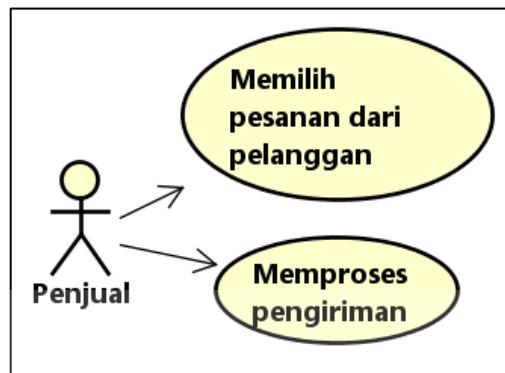
#### g. Pembeli Melakukan Pembelian



Gambar 3. 21 Pembeli melakukan pembelian

Use case sistem pembeli melakukan pembelian dilakukan oleh pembeli. pembeli dapat membuka keranjang, menyelesaikan proses checkout, mengisi data pembeli, dan melakukan proses pembayaran.

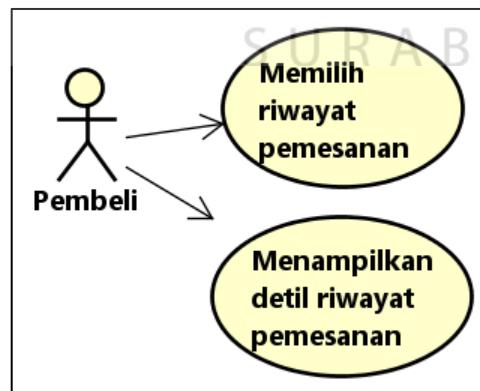
#### h. Penjual Melakukan Penjualan



Gambar 3. 22 Penjual melakukan penjualan

Use case sistem penjual melakukan penjualan dilakukan oleh penjual. penjual dapat memilih pesanan dari pelanggan dan memproses pengiriman.

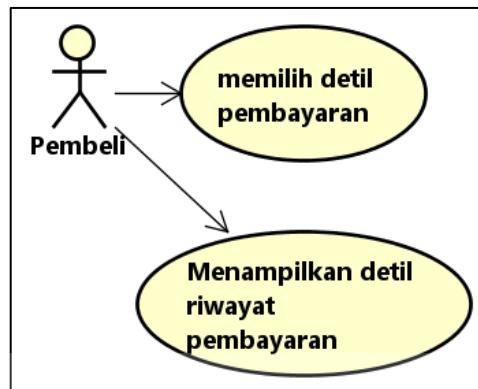
#### i. Pengolahan Laporan Pemesanan dari Pembeli



Gambar 3. 23 Pengolahan laporan pemesanan dari pembeli

Use case sistem pengolahan laporan pemesanan dari pembeli dilakukan oleh pembeli. pembeli dapat memilih riwayat pemesanan dan menampilkan detail riwayat pemesanan.

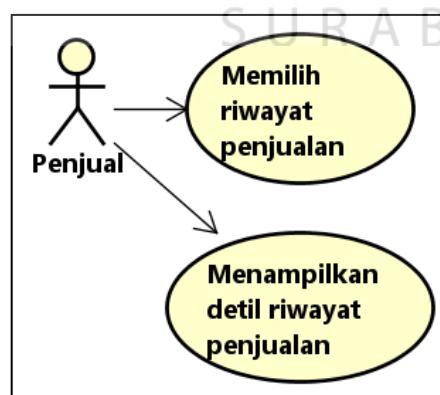
#### j. Pengolahan Laporan Pembelian dari Pembeli



Gambar 3. 24 Pengolahan laporan pembelian dari pembeli

Use case sistem pengolahan laporan pembelian dari pembeli dilakukan oleh pembeli. pembeli dapat memilih detail pembayaran dan menampilkan detail riwayat pembayaran.

#### k. Pengolahan Laporan Penjualan dari Penjual



Gambar 3. 25 Pengolahan laporan penjualan dari penjual

Use case sistem pengolahan laporan penjualan dari penjual dilakukan oleh penjual. penjual dapat memilih riwayat penjualan dan menampilkan detail riwayat penjualan.

### 3.3.5 Use Case Description

#### a. Membuat Akun Baru

Tabel 3. 5 Use Case Description Membuat Akun Baru

<b>Nama Use Case</b>	<b>Membuat Akun Baru</b>		
Aktor Primer	Pembeli		
Aktor Sekunder	-		
Aktor Terkait Kebutuhannya	<ul style="list-style-type: none"> <li>• Pembeli: Membuat akun baru.</li> </ul>		
Deskripsi Singkat	Pengguna ingin membuat akun sebagai pembeli.		
Pemicu	Pembeli memilih menu registrasi.		
Aliran Normal	<b>Langkah</b>	<b>Aktor</b>	<b>Aplikasi</b>
	1.	Aktor memilih menu registrasi.	Sistem menampilkan halaman registrasi.
	2.	Aktor mengisi data registrasi & menekan tombol simpan	Sistem menyimpan data registrasi
	3.		Use Case Berakhir
Sub Aliran	Tidak ada		
Aliran Kesalahan	<b>Langkah</b>	<b>Aktor</b>	<b>Aplikasi</b>
	2.1		Sistem menampilkan pesan data wajib diisi

#### b. Membuka Toko

Tabel 3. 6 Use Case Description Membuka Toko

<b>Nama Use Case</b>	<b>Membuka Toko</b>		
Aktor Primer	Penjual		
Aktor Sekunder	-		
Aktor Terkait Kebutuhannya	<ul style="list-style-type: none"> <li>• Penjual: Membuka Toko.</li> </ul>		
Deskripsi Singkat	Penjual ingin membuka toko.		
Pemicu	Penjual memilih menu buka toko.		
Aliran Normal	<b>Langkah</b>	<b>Aktor</b>	<b>Aplikasi</b>

	1.	Aktor memilih menu buka toko.	Sistem menampilkan form buka toko.
	2.	Aktor mengisi data toko & menekan tombol simpan	Sistem menyimpan data toko.
	3.		Use case selesai
Sub Aliran	Tidak ada		
Aliran Kesalahan	<b>Langkah</b>	<b>Aktor</b>	<b>Aplikasi</b>
	2.1		Sistem menampilkan pesan kesalahan validasi data toko

### c. Mengelola Produk

Tabel 3. 7 *Usecase Description* Mengelola Produk

<b>Nama Use Case</b>	<b>Mengelola Produk</b>		
Aktor Primer	Penjual		
Aktor Sekunder	-		
Aktor Terkait Kebutuhannya	<ul style="list-style-type: none"> <li>• Penjual: Mengelola Produk Toko.</li> </ul>		
Deskripsi Singkat	Penjual ingin mengelola produk pada toko.		
Pemicu	Penjual memilih menu produk.		
Aliran Normal	<b>Langkah</b>	<b>Aplikasi</b>	<b>Sistem</b>
	1.	Aktor memilih menu pengelolaan produk.	Sistem menampilkan halaman pengelolaan produk.
	2.	Jika Aktor menekan menambah produk masuk sub aliran 3.1, jika Aktor ingin mengupdate produk masuk sub aliran 3.2, jika Aktor ingin menghapus produk masuk sub aliran 3.3 .	
	3.	Use case selesai	
Sub Aliran	<b>Langkah</b>	<b>Aktor</b>	<b>Aplikasi</b>

	3.1	Aktor menekan tombol tambah data.	Sistem menampilkan halaman tambah produk.
		Aktor mengisi data produk yang diperlukan & menekan tombol simpan.	Sistem menyimpan ke database
	3.2	Aktor menekan update pada produk terkait	Sistem menampilkan halaman update produk.
		Aktor mengisi data produk yang diperlukan & menekan tombol simpan perubahan.	Sistem menyimpan perubahan ke database
Aliran Kesalahan	3.3	Aktor memilih produk yang ingin dihapus.	Sistem menampilkan konfirmasi hapus.
		Aktor menekan tombol hapus.	Sistem menghapus data dari database
Aliran Kesalahan	<b>Tidak ada</b>		

#### d. Membuka Toko

Tabel 3. 8 Mengelola saldo

<b>Nama Use Case</b>	<b>Mengelola</b>		
Aktor Primer	Penjual		
Aktor Sekunder	-		
Aktor Terkait Kebutuhannya	<ul style="list-style-type: none"> <li>• Penjual: Penarikan saldo.</li> </ul>		
Deskripsi Singkat	Penjual ingin melakukan penarikan saldo.		
Pemicu	Penjual memilih saldo penjual.		
Aliran Normal	<b>Langkah</b>	<b>Aktor</b>	<b>Aplikasi</b>
	1.	Aktor memilih saldo penjual.	Sistem menampilkan

			form penarikan saldo.
	2.	Aktor mengisi data penarikan & menekan tombol simpan	Sistem menyimpan data penarikan.
	3.		Use case selesai
Sub Aliran	Tidak ada		
Aliran Kesalahan	<b>Langkah</b>	<b>Aktor</b>	<b>Aplikasi</b>
	2.1		Sistem menampilkan pesan kesalahan validasi data penarikan

#### e. Mengelola Etalase

Tabel 3. 9 *Usecase Description* Mengelola Etalase

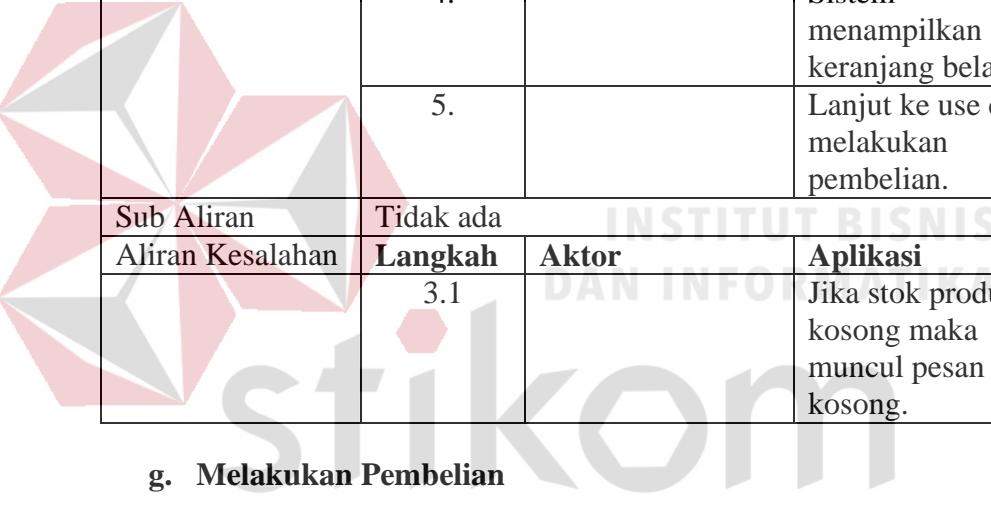
<b>Nama Use Case</b>	<b>Mengelola Etalase</b>		
Aktor Primer	Penjual		
Aktor Sekunder	-		
Aktor Terkait Kebutuhannya	<ul style="list-style-type: none"> <li>• Penjual: Mengelola Etalase.</li> </ul>		
Deskripsi Singkat	Penjual ingin mengelola etalase pada toko.		
Pemicu	Penjual memilih menu pengelolaan etalase.		
Aliran Normal	<b>Langkah</b>	<b>Aktor</b>	<b>Aplikasi</b>
	1.	Aktor memilih menu pengelolaan etalase.	Sistem menampilkan halaman pengelolaan etalase.
	2.	Jika Aktor menekan menambah etalase masuk sub aliran 3.1, jika Aktor ingin mengupdate etalase masuk sub aliran 3.2, jika Aktor ingin menghapus etalase masuk sub aliran 3.3 .	
	3.	Use case selesai	
Sub Aliran	<b>Langkah</b>	<b>Aktor</b>	<b>Aplikasi</b>

	3.1	Aktor menekan tombol tambah data.	Sistem menampilkan dialog tambah etalase.
		Aktor mengisi data etalase yang diperlukan & menekan tombol simpan.	Sistem menyimpan ke database.
	3.2	Aktor menekan update pada etalase terkait	Sistem menampilkan dialog update etalase.
		Aktor mengisi data etalase yang diperlukan & menekan tombol simpan perubahan.	Sistem menyimpan perubahan ke database.
Aliran Kesalahan	3.3	Aktor memilih etalase yang ingin dihapus. Sistem menampilkan konfirmasi hapus.	Aktor menekan tombol hapus(3.3.1). Sistem menghapus data dari database
<b>Aliran Kesalahan</b>	<b>Langkah</b>	<b>Aktor</b>	<b>Aplikasi</b>
	3.3.1		Jika etalase yang dihapus masih memiliki barang, maka muncul notifikasi “etalase masih memiliki barang”

#### f. Melakukan Pemesanan

Tabel 3. 10 *Use case Description* Melakukan Pemesanan

<b>Nama Use Case</b>	<b>Melakukan Pemesanan</b>
Aktor Primer	Pembeli
Aktor Sekunder	-



Aktor Terkait Kebutuhannya	<ul style="list-style-type: none"> <li>• Pembeli: Melakukan pemesanan barang.</li> </ul>		
Deskripsi Singkat	Pembeli ingin melakukan pembelian pemesanan barang.		
Pemicu	Pembeli membuka halaman utama produk.		
Aliran Normal	Langkah	Aksi	
	1.	Aktor memilih halaman utama produk.	Sistem menampilkan halaman utama produk.
	2.	Aktor memilih produk.	Sistem menampilkan detail produk.
	3.	Aktor memasukkan produk ke keranjang.	Sistem menyimpan produk terpilih ke keranjang.
	4.		Sistem menampilkan keranjang belanja.
	5.		Lanjut ke use case melakukan pembelian.
Sub Aliran	Tidak ada		
Aliran Kesalahan	Langkah	Aktor	Aplikasi
	3.1		Jika stok produk kosong maka muncul pesan stok kosong.

#### g. Melakukan Pembelian

Tabel 3. 11 *Use Case Description* Melakukan Pembelian

Nama Use Case	Melakukan Pembelian		
Aktor Primer	Pembeli		
Aktor Sekunder	Penjual		
Aktor Terkait Kebutuhannya	<ul style="list-style-type: none"> <li>• Pembeli: Melakukan pemesanan barang.</li> </ul>		
Deskripsi Singkat	Pembeli ingin melakukan pembayaran terhadap barang yang telah dipesan.		
Pemicu	Pembeli membuka keranjang belanja.		
Aliran Normal	Langkah	Aksi	
	1.	Aktor memilih keranjang belanja.	Sistem menampilkan keranjang belanja.
	2.	Aktor melanjutkan	Sistem menampilkan

		proses checkout produk.	halaman pembayaran.
3.	Aktor mengisi data pembeli.		
4.	Aktor menyelesaikan proses pembayaran.	Sistem menyimpan transaksi pembelian.	
5.	Penjual menerima pesanan.		
6.	Use case selesai.		
Sub Aliran	Tidak ada		
Aliran Kesalahan	Langkah	Aktor	Aplikasi
	1.1		Jika keranjang belanja kosong muncul pesan “keranjang belanja kosong”.
	4.1		Jika saldo tidak mencukupi muncul pesan “saldo tidak mencukupi”.

#### h. Melakukan Penjualan

Tabel 3. 12 *Usecase Description* Melakukan Penjualan

Nama Use Case	Melakukan penjualan		
Aktor Primer	Penjual		
Aktor Sekunder	-		
Aktor Terkait Kebutuhannya	<ul style="list-style-type: none"> <li>Penjual: Melakukan proses penjualan barang.</li> </ul>		
Deskripsi Singkat	Penjual ingin melakukan pembelian pemesanan barang.		
Pemicu	Penjual membuka halaman penjualan.		
Aliran Normal	Langkah	Aktor	Aplikasi
	1.	Aktor memilih halaman penjualan.	Sistem menampilkan halaman penjualan.
	2.	Aktor memilih data pesanan dari pelanggan.	Sistem menampilkan detail pesanan.
	3.	Aktor memproses pengiriman pesanan.	Sistem menyimpan dan update transaksi penjualan.
	4.		Use case selesai

Sub Aliran	Tidak ada		
Aliran Kesalahan	<b>Langkah</b>	<b>Aktor</b>	<b>Aplikasi</b>

### i. Pengolahan Laporan Pemesanan

Tabel 3. 13 *Usecase Description* Pengolahan Laporan Pemesanan

<b>Nama Use Case</b>	<b>Pengolahan Laporan Pemesanan</b>		
Aktor Primer	Pembeli		
Aktor Sekunder	-		
Aktor Terkait Kebutuhannya	<ul style="list-style-type: none"> <li>• Pembeli: Melihat riwayat pesanan.</li> </ul>		
Deskripsi Singkat	Penjual ingin melihat riwayat pesanan.		
Pemicu	Penjual membuka riwayat pesanan		
Aliran Normal	<b>Langkah</b>	<b>Aktor</b>	<b>Aplikasi</b>
	1.	Aktor memilih halaman riwayat pesanan.	Sistem menampilkan halaman riwayat pesanan.
	2.	Aktor memilih riwayat pesanan.	Sistem menampilkan detail riwayat pesanan.
	3.	Aktor memilih detail pembayaran.	
	4.	Lanjut ke use case pengolahan laporan pembelian.	
Sub Aliran	Tidak ada		
Aliran Kesalahan	<b>Langkah</b>	<b>Aktor</b>	<b>Aplikasi</b>
	1.1		Jika aktor tidak memiliki riwayat pesanan muncul pesan “belum memiliki pesanan”

### j. Pengolahan Laporan Pembelian

Tabel 3. 14 *Usecase Description* Pengolahan Laporan Pembelian

<b>Nama Use Case</b>	<b>Pengolahan Laporan Pembelian</b>		
Aktor Primer	Pembeli		

Aktor Sekunder	-		
Aktor Terkait Kebutuhannya	<ul style="list-style-type: none"> <li>Pembeli: Melihat riwayat pesanan.</li> </ul>		
Deskripsi Singkat	Pembeli ingin melihat detil pembayaran.		
Pemicu	Pembeli memilih detil pembayaran.		
Aliran Normal	Langkah	Aktor	Aplikasi
	1.	Aktor memilih detil riwayat pembayaran.	Sistem menampilkan detil riwayat pemesanan.
	2.		Use case selesai
Sub Aliran	Tidak ada		
Aliran Kesalahan	Tidak ada		

### k. Pengolahan Laporan Penjualan

Tabel 3. 15 *Usecase Description* Pengolahan Laporan Penjualan

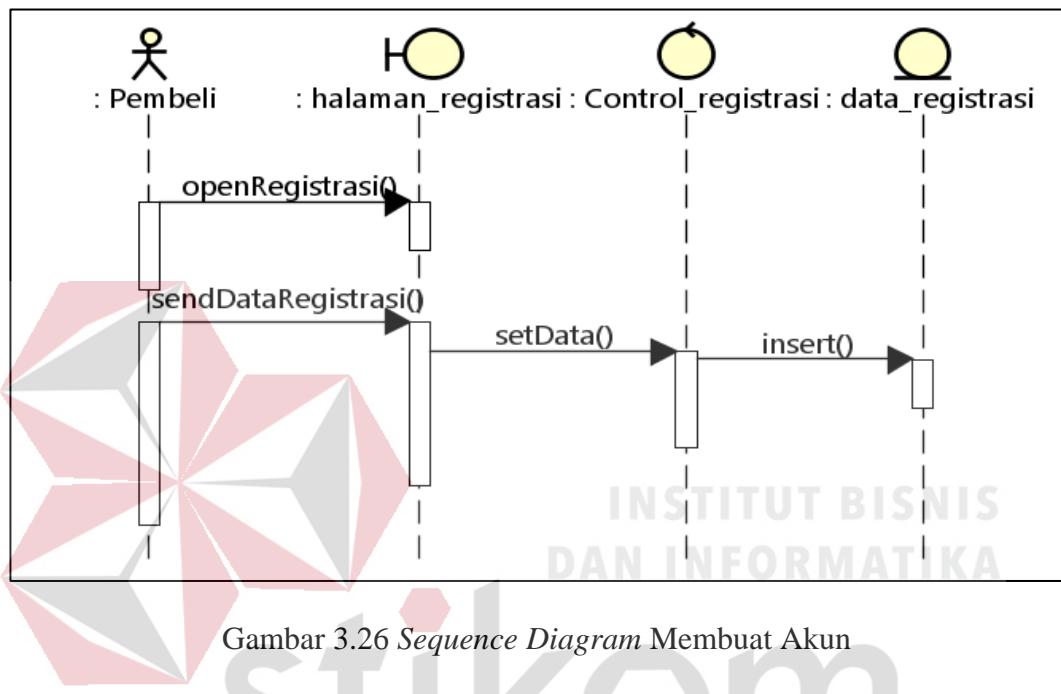
<b>Nama Use Case</b>	<b>Pengolahan Laporan Penjualan</b>		
Aktor Primer	Penjual		
Aktor Sekunder	-		
Aktor Terkait Kebutuhannya	<ul style="list-style-type: none"> <li>Penjual: Melihat riwayat penjualan.</li> </ul>		
Deskripsi Singkat	Penjual ingin melihat riwayat penjualan.		
Pemicu	Penjual membuka riwayat penjualan		
Aliran Normal	Langkah	Aksi	
	1.	Aktor memilih halaman riwayat penjualan.	Sistem menampilkan halaman riwayat penjualan.
	2.	Aktor memilih riwayat penjualan.	Sistem menampilkan detil riwayat penjualan.
	3.		Use case selesai
Sub Aliran	Tidak ada		
Aliran Kesalahan	Langkah	Aktor	Aplikasi
	1.1		Jika aktor tidak memiliki riwayat pesanan muncul pesan “belum memiliki pesanan”

### 3.3.6 Sequence Diagram

#### a. Pembeli Membuat Akun

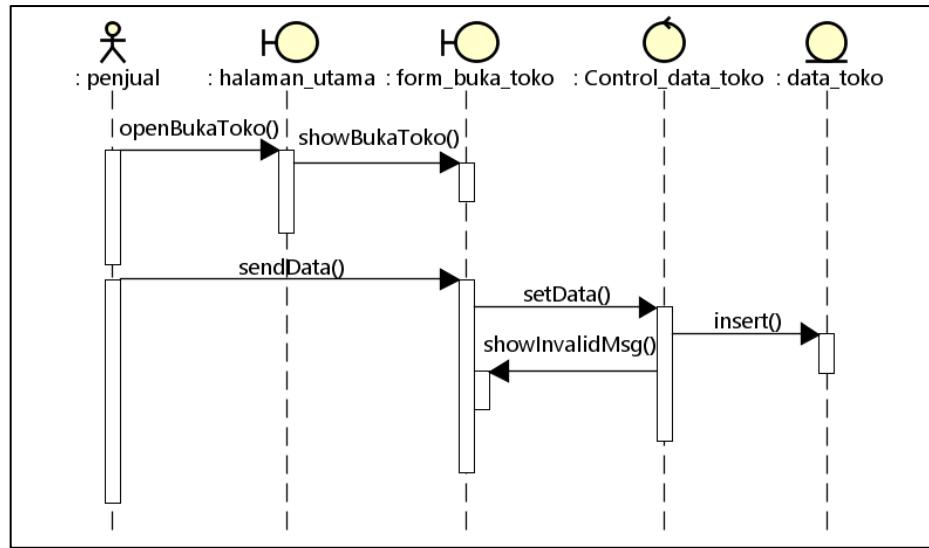
Sequence diagram pembeli membuat akun akan menjelaskan proses pembuatan akun. Pembuatan akun dimulai saat pembeli membuka halaman

registrasi, pada saat itu aplikasi akan mengirim data untuk membuka halaman registrasi. Kemudian halaman registrasi akan mengirim pesan kepada control register, control register akan menyimpan data yang telah diisi oleh pembeli pada database data registrasi. *Sequence Diagram* Membuat Akun dapat dilihat pada gambar 2.26.



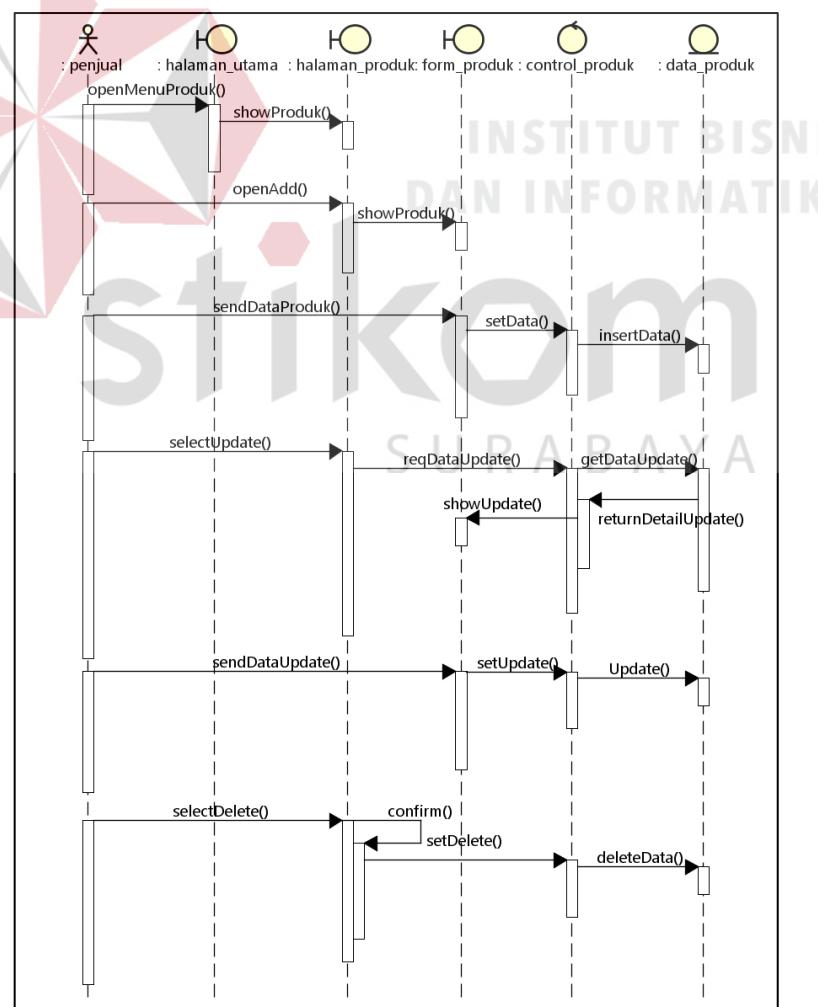
### b. Penjual Membuka Toko

Sequence diagram penjual membuka toko akan menjelaskan proses penjual membuka toko. Penjual membuka halaman utama, pada saat itu aplikasi akan mengirim data untuk membuka form buka toko dan penjual akan membuka form buka toko. Kemudian form buka toko akan mengirim pesan kepada control data toko, control data toko akan mengecek apakah data yang diisi sudah valid atau belum, jika belum penjual harus mengisi lagi data yang belum valid, jika sudah valid data toko yang telah diisi akan disimpan pada database data toko. *Sequence Diagram* Membuka Toko dapat dilihat pada gambar 3.27.



Gambar 3.27 Sequence Diagram Membuka Toko

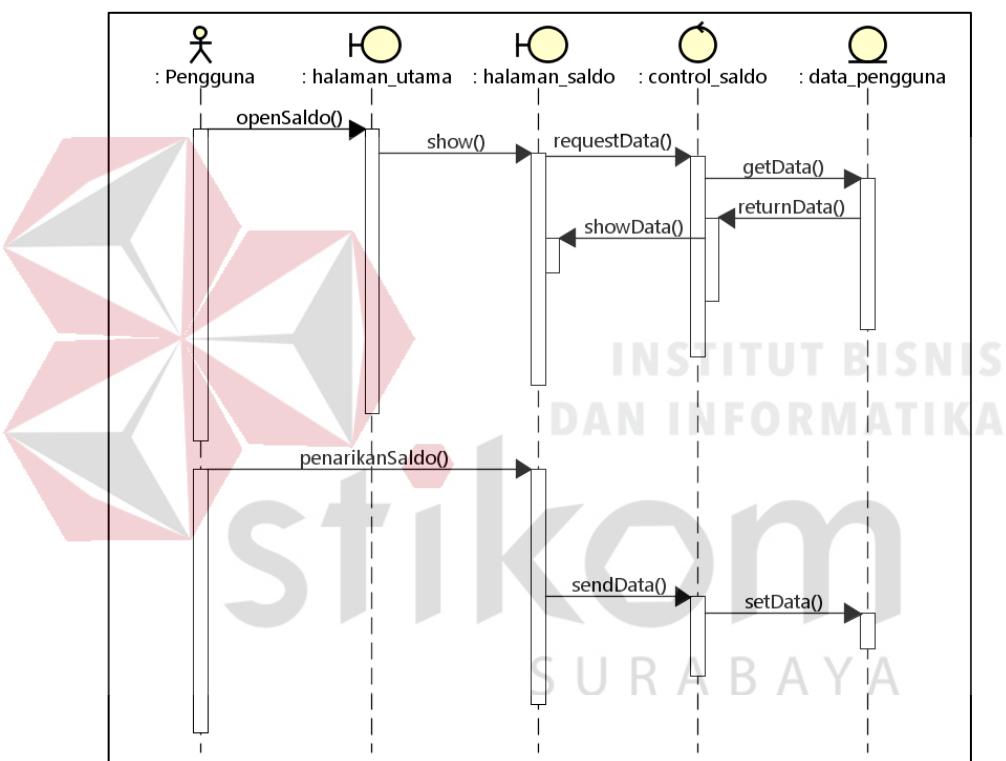
### c. Penjual Mengelola Produk



Gambar 3.28 Sequence Diagram Mengelola Produk

Sequence diagram penjual mengelola produk akan menjelaskan proses pengelolaan produk penjual. Penjual akan membuka halaman utama dan akan memilih menu produk kemudian akan menampilkan halaman produk. Setelah membuka halaman produk aplikasi akan menampilkan form produk. Kemudian form produk akan mengirim pesan kepada control produk, control produk akan menyimpan data yang telah diisi oleh penjual pada database data produk.

#### d. Pengguna Mengelola Saldo

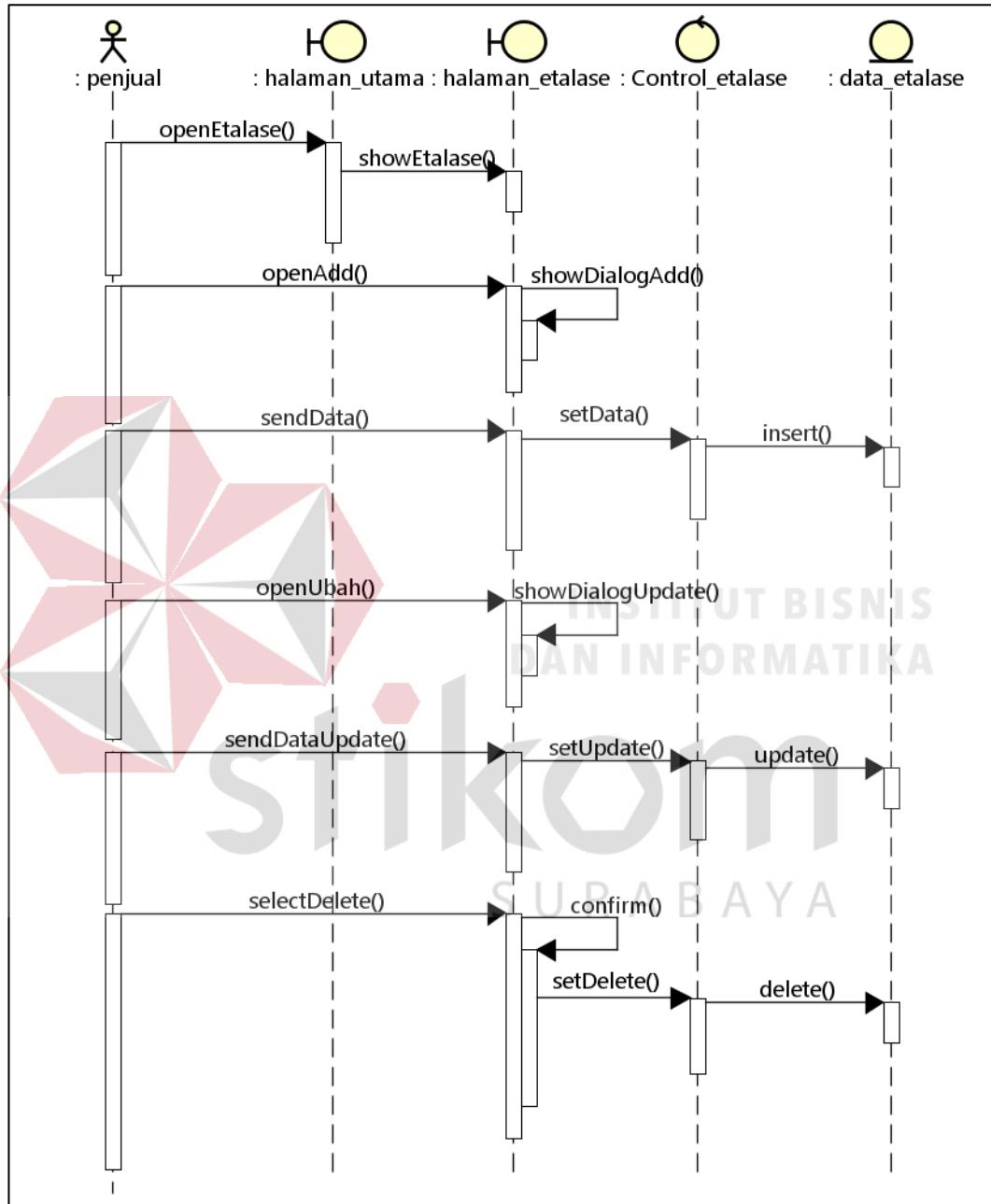


Gambar 3.29 Mengelola saldo

Sequence diagram pengguna mengelola saldo akan menjelaskan proses pengelolaan saldo pengguna. Pengguna akan membuka halaman utama dan akan memilih menu saldo kemudian akan membuka halaman saldo. Kemudian aplikasi akan mengambil data pengguna pada database data pengguna, data pengguna akan mengirim pesan pada control saldo dan akan menampilkan data pengguna pada halaman saldo. Setelah itu pengguna akan mendepositkan saldo pada halaman

saldo, halaman saldo akan mengirim pesan kepada control saldo, control saldo akan menyimpan data yang telah diisi oleh pengguna pada database data pengguna.

### e. Penjual Mengelola Etalase

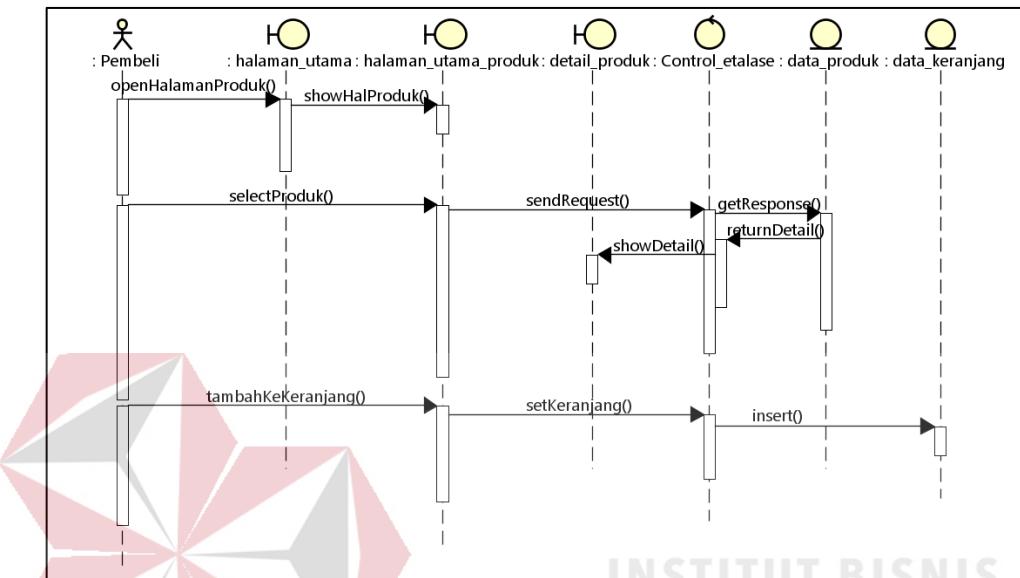


Gambar 3.30 Sequence Diagram Mengelola Etalase

Sequence diagram penjual mengelola etalase akan menjelaskan proses pengelolaan etalase oleh penjual. Penjual membuka halaman utama dan memilih

menu etalase kemudian akan membuka halaman etalase. Penjual dapat melakukan proses penambahan, perubahan, dan penghapusan etalase yang akan diterima oleh control etalase dan di proses terhadap data etalase.

#### f. Pembeli Melakukan Pemesanan



Gambar 3.31 Sequence Diagram Pembeli Melakukan Pemesanan

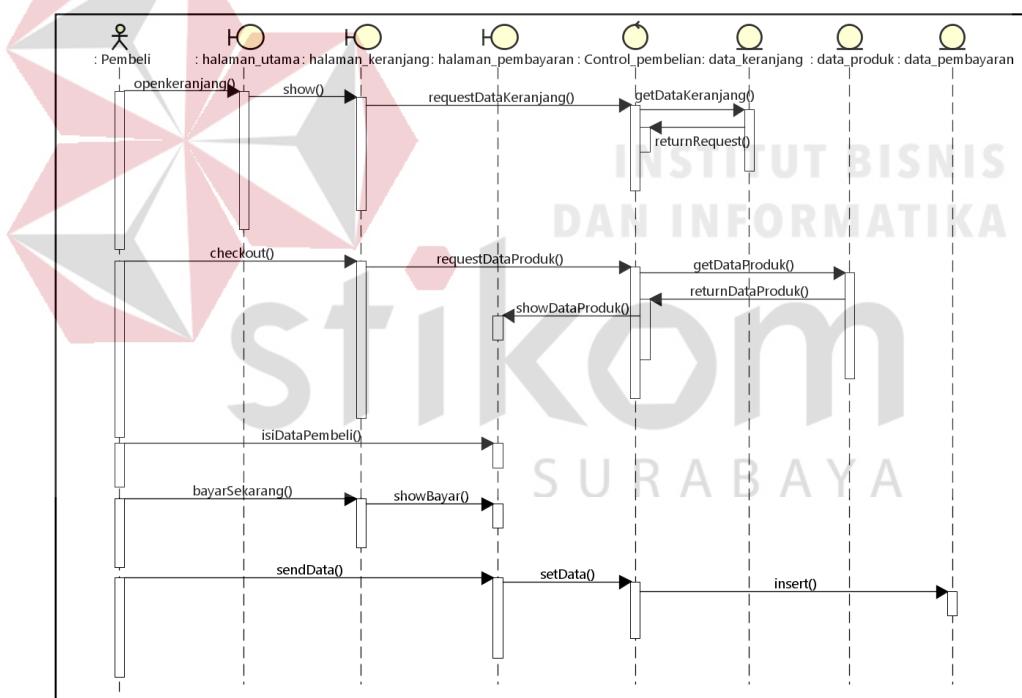
Sequence diagram pembeli melakukan pemesanan akan menjelaskan proses pemesanan yang dilakukan pembeli. pembeli membuka halaman utama dan memilih menu halaman produk kemudian akan membuka halaman utama produk. Pembeli akan memilih produk pada halaman utama produk. Kemudian halaman utama produk akan mengirimkan pesan pada control etalase. Aplikasi akan mengambil data detail produk pada database data produk, data produk akan mengirim pesan pada control etalase dan akan menampilkan data detail produk pada halaman detail produk.

#### g. Pembeli Melakukan Pembelian

Sequence diagram pembeli melakukan pembelian akan menjelaskan proses pembelian. Pembeli akan membuka menu keranjang pada halaman utama dan akan

menampilkan halaman keranjang. Pembeli melakukan checkout pada halaman keranjang. Aplikasi akan mengambil data keranjang dan data produk pada database data keranjang dan database data produk, data keranjang akan mengirim pesan pada control pembelian dan akan menampilkan data keranjang dan data produk pada halaman keranjang. Pembeli akan mengisi data pembelian pada halaman pembayaran, kemudian pembeli akan memilih menu bayar sekarang pada halaman keranjang dan akan menampilkan halaman pembayaran. Pengguna akan mengirim data pada halaman pembayaran, halaman pembayaran akan mengirim pesan pada control pembelian untuk memasukkan pembayaran pada data pembayaran. Gambar

*Sequence Diagram* Pembeli Melakukan Pembelian dapat dilihat pada gambar 3.32.



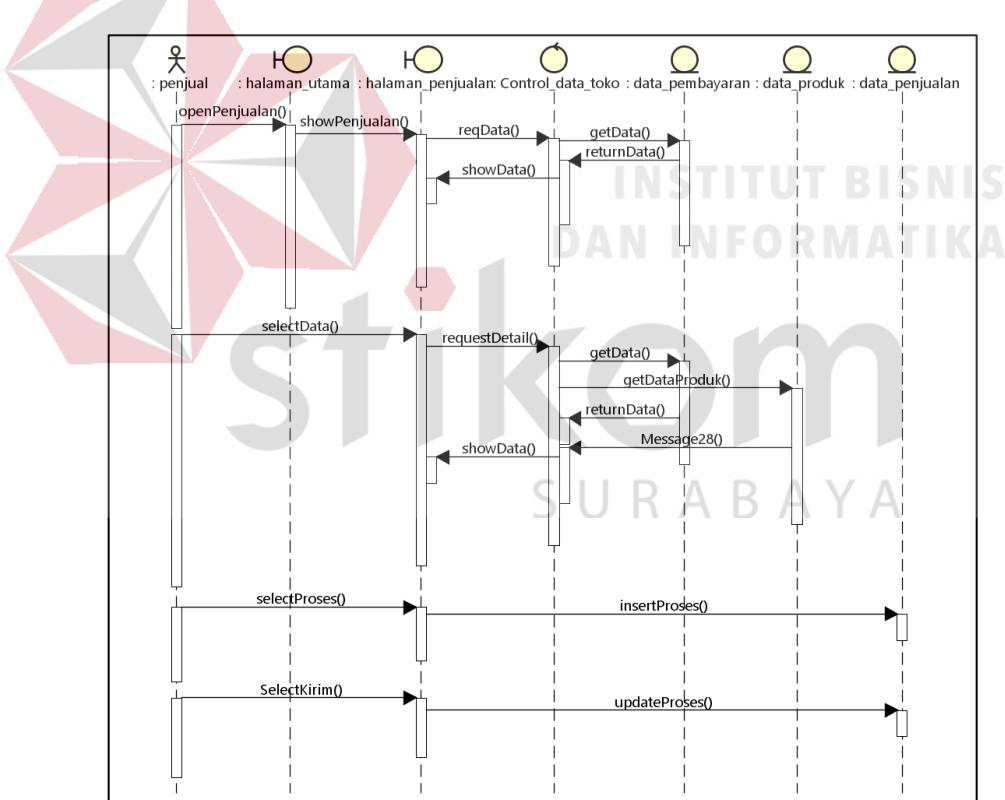
Gambar 3.32 *Sequence Diagram* Pembeli Melakukan Pembelian

#### **h. Penjual Melakukan Penjualan**

Sequence diagram penjual melakukan penjualan akan menjelaskan proses penjual dalam melakukan penjualan. Penjual akan membuka menu penjualan pada halaman utama dan akan menampilkan halaman penjualan. Halaman penjualan

akan mengirim pesan pada control data toko, kemudian control data toko akan mengambil data pembayaran pada database data pembayaran dan akan ditampilkan pada halaman penjualan. Penjual akan memilih data pada halaman penjualan, halaman penjualan akan mengirimkan pesan pada control data toko, kemudian control data toko akan mengambil data pembayaran dan data produk pada database data pembayaran dan database data produk yang akan ditampilkan pada halaman penjualan. Penjual memilih proses pada halaman penjualan dan akan dimasukkan pada database penjualan. Pengguna juga dapat memilih kirim pada halaman penjualan dan akan di update pada database data penjualan. *Sequence Diagram*

Penjual Melakukan Penjualan dapat dilihat pada gambar 3.33.

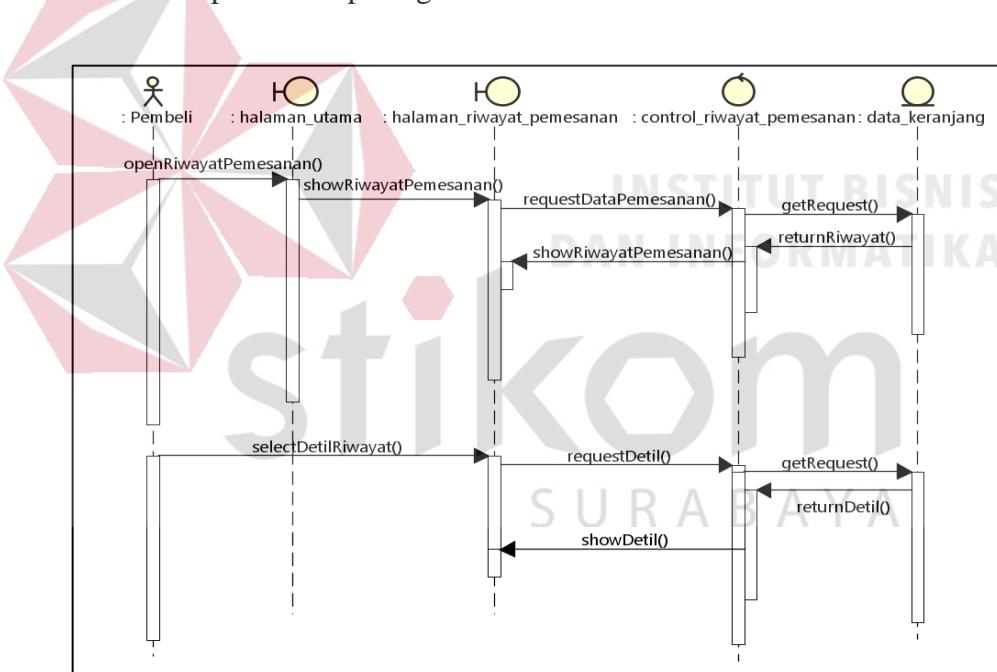


Gambar 3.33 *Sequence Diagram* Penjual Melakukan Penjualan

#### i. Pengolahan Laporan Pemesanan dari Pembeli

Sequence diagram pengolahan laporan pemesanan dari pembeli akan menjelaskan tentang pengolahan laporan pemesanan dari pembeli. Pembeli akan

membuka menu riwayat pemesanan pada halaman utama dan akan menampilkan halaman riwayat pemesanan. Halaman riwayat pemesanan akan mengambil data riwayat pada database keranjang dan akan dikirim pada control riwayat pemesanan, control riwayat pemesanan akan menampilkan data riwayat pemesanan pada halaman riwayat pemesanan. Pembeli akan memilih detail riwayat pada halaman riwayat pemesanan, kemudian halaman riwayat pemesanan akan mengirimkan pesan pada control riwayat pemesanan dan halaman riwayat pemesanan akan mengambil data detail pada database keranjang yang akan menampilkan detail pada halaman riwayat pemesanan. *Sequence Diagram Pengolahan Laporan Pemesanan* dari Pembeli dapat dilihat pada gambar 3.34.



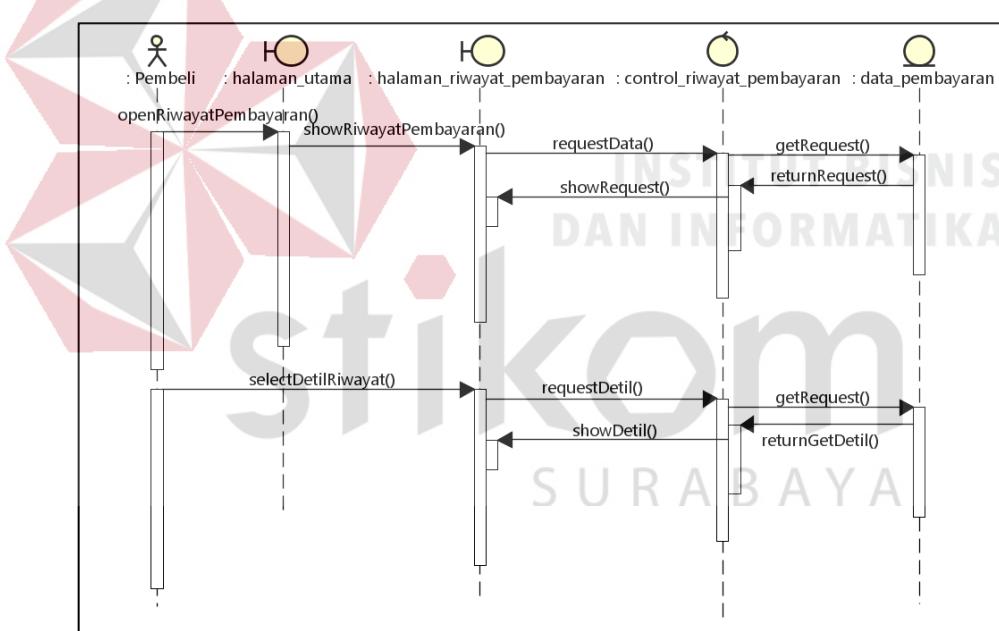
Gambar 3.34 *Sequence Diagram Pengolahan Laporan Pemesanan* dari Pembeli

#### j. Pengolahan Laporan Pembayaran dari Pembeli

Sequence diagram pengolahan laporan pembayaran dari pembeli akan menjelaskan tentang pengolahan laporan pembayaran dari pembeli. Pembeli akan membuka menu riwayat pembayaran pada halaman utama dan akan menampilkan

halaman riwayat pembayaran. Halaman riwayat pembayaran akan mengambil data riwayat pada database data penjualan dan akan dikirim pada control riwayat pembayaran, control riwayat pembayaran akan menampilkan data riwayat pembayaran pada halaman riwayat pembayaran. Pembeli akan memilih detail riwayat pada halaman riwayat pembayaran, kemudian halaman riwayat pembayaran akan mengirimkan pesan pada control riwayat pembayaran dan halaman riwayat pembayaran akan mengambil data detail pada database data penjualan yang akan menampilkan detail pada halaman riwayat pembayaran.

*Sequence Diagram* Pengolahan Laporan Pembayaran dari Pembeli dapat dilihat pada gambar 3.35.

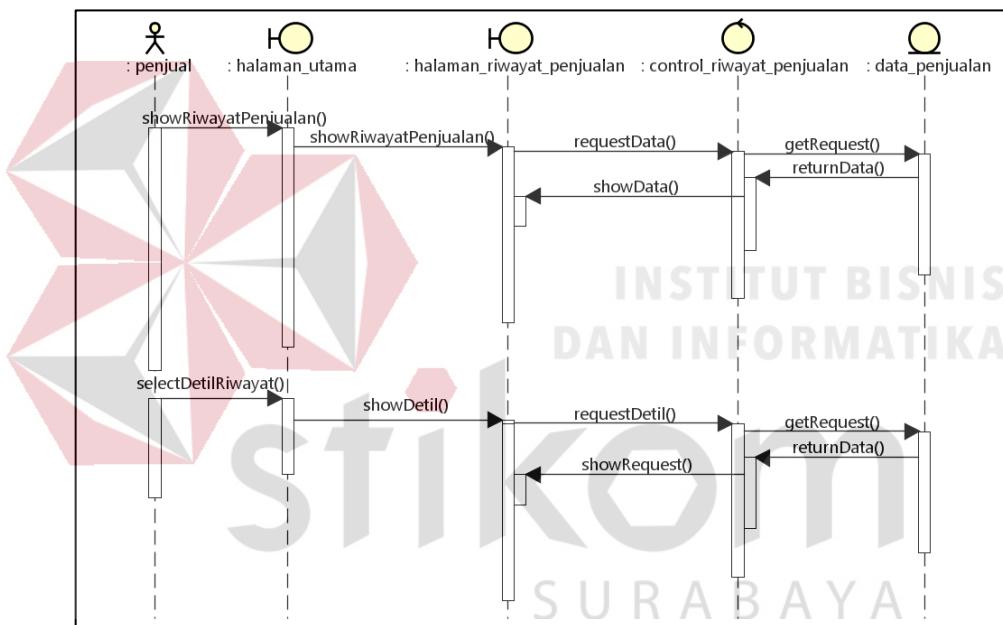


Gambar 3.35 *Sequence Diagram* Pengolahan Laporan Pembayaran dari Pembeli

#### k. Pengolahan Laporan Penjualan dari Penjual

Sequence diagram pengolahan laporan penjualan dari penjual akan menjelaskan tentang pengolahan laporan penjualan dari penjual. Penjual akan membuka menu riwayat penjualan pada halaman utama dan akan menampilkan halaman riwayat penjualan. Halaman riwayat penjualan akan mengambil data

riwayat penjualan pada database data penjualan dan akan dikirim pada control riwayat penjualan yang akan menampilkan riwayat penjualan pada halaman riwayat penjualan. Penjual akan memilih menu detail riwayat pada halaman utama dan halaman utama akan me request detail pada halaman riwayat penjualan. Halaman riwayat penjualan akan mengambil data detail pada database data penjualan dan akan dikirim pada control riwayat penjualan yang akan menampilkan detail pada halaman riwayat penjualan. *Sequence Diagram* Pengolahan Laporan Penjualan dari Penjual dapat dilihat pada gambar 3.36.

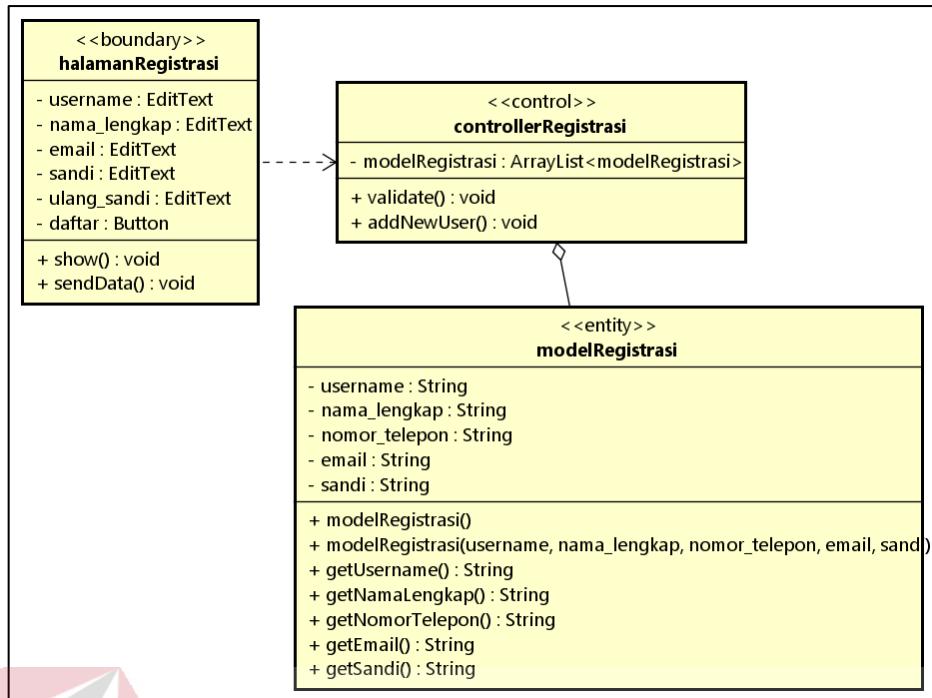


Gambar 3.36 *Sequence Diagram* Pengolahan Laporan Penjualan dari Penjual

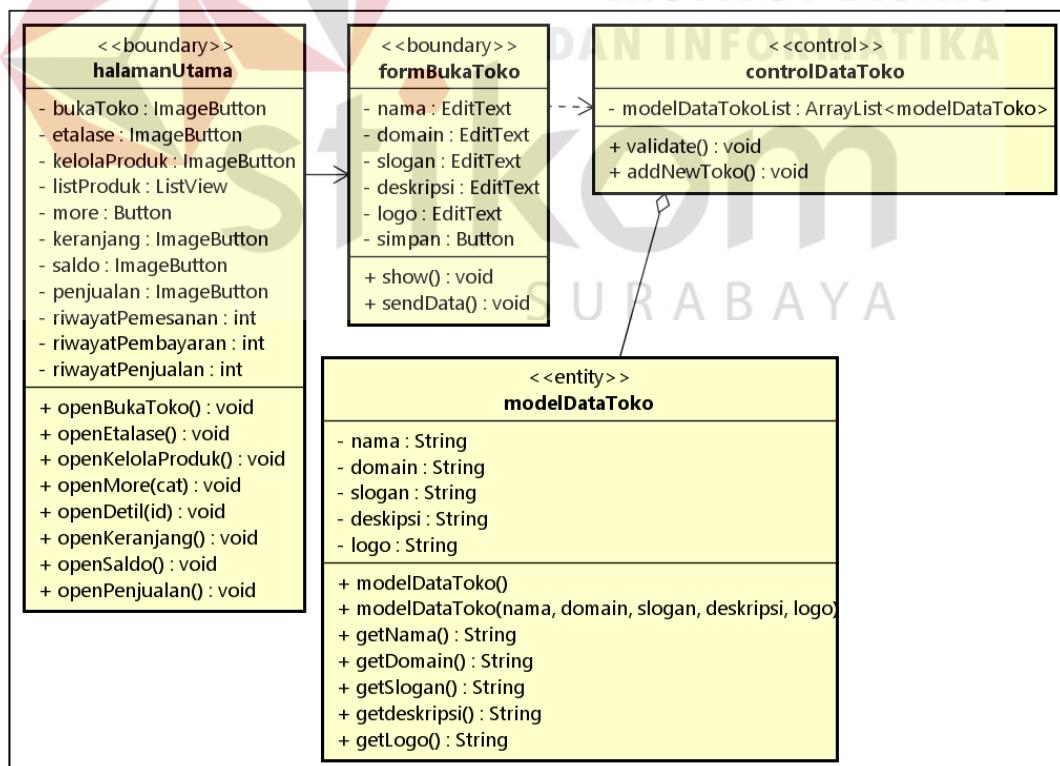
### 3.3.7 Class diagram

#### a. *Class Diagram* Membuat Akun Baru

Class diagram membuat akun baru akan menjelaskan tentang hubungan antara modelRegistrasi yang memiliki hubungan agregasi terhadap controllerRegistrasi. Entitas modelRegistrasi akan digunakan pada halamanRegistrasi. *Class Diagram* Membuat Akun Baru dapat dilihat pada gambar 3.37.

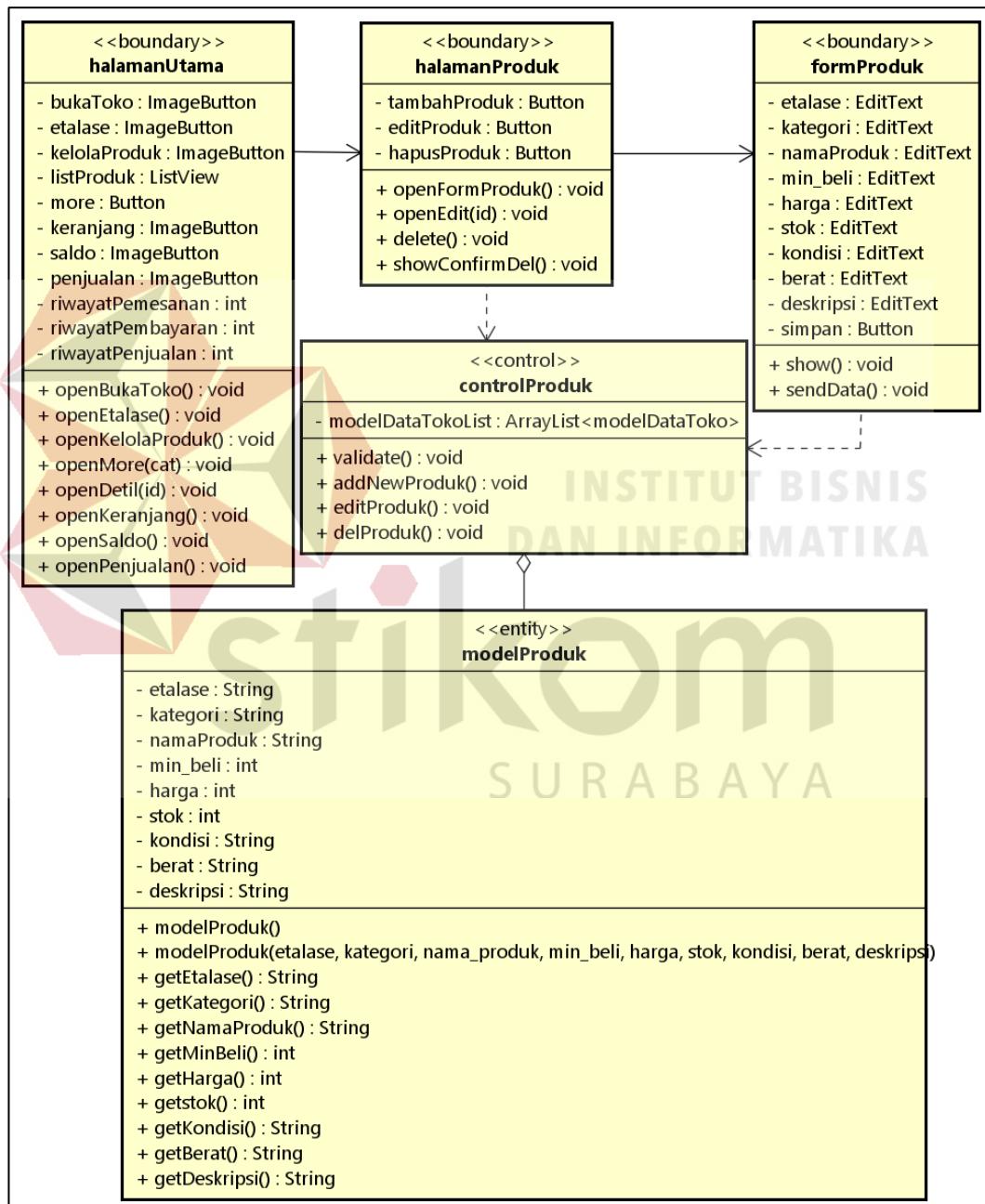
Gambar 3.37 *Class Diagram* Membuat Akun Baru

### b. *Class Diagram* Penjual Membuat Toko

Gambar 3.38 *Class diagram* penjual membuat toko

Class diagram penjual membuat toko akan menjelaskan tentang hubungan antara modelDataToko yang memiliki hubungan agregasi terhadap controlDataToko. Entitas yang berhubungan dengan entitas controlDataToko akan ditampilkan pada halaman utama.

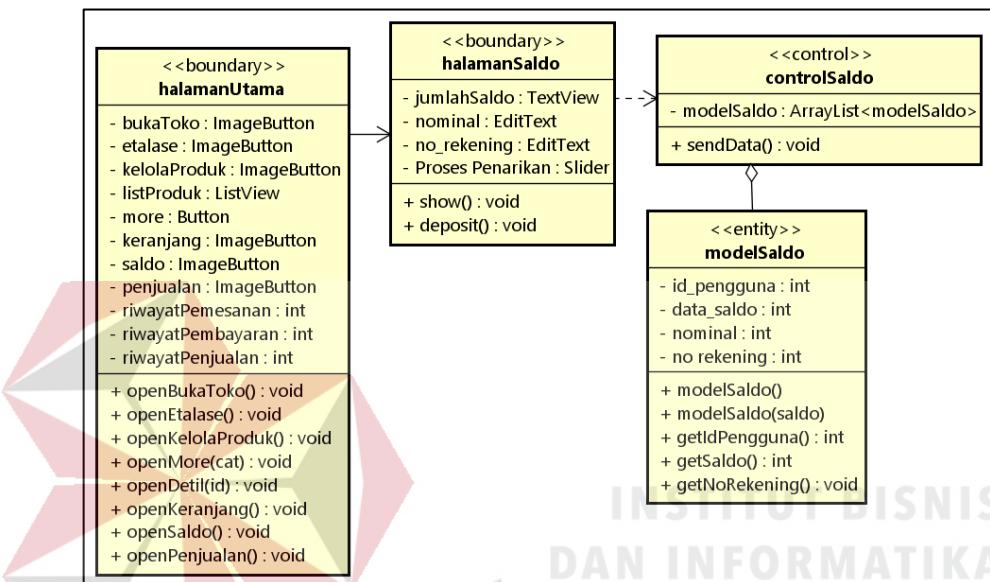
### c. *Class Diagram Penjual Mengelola Produk*



Gambar 3.39 *Class diagram penjual mengelola produk*

Class diagram penjual mengelola produk akan menjelaskan tentang hubungan antara modelProduk yang memiliki hubungan agregasi dengan controlProduk. Entitas yang berhubungan dengan controlProduk akan ditampilkan pada halamanProduk.

#### d. *Class Diagram Pengguna Mengelola Saldo*

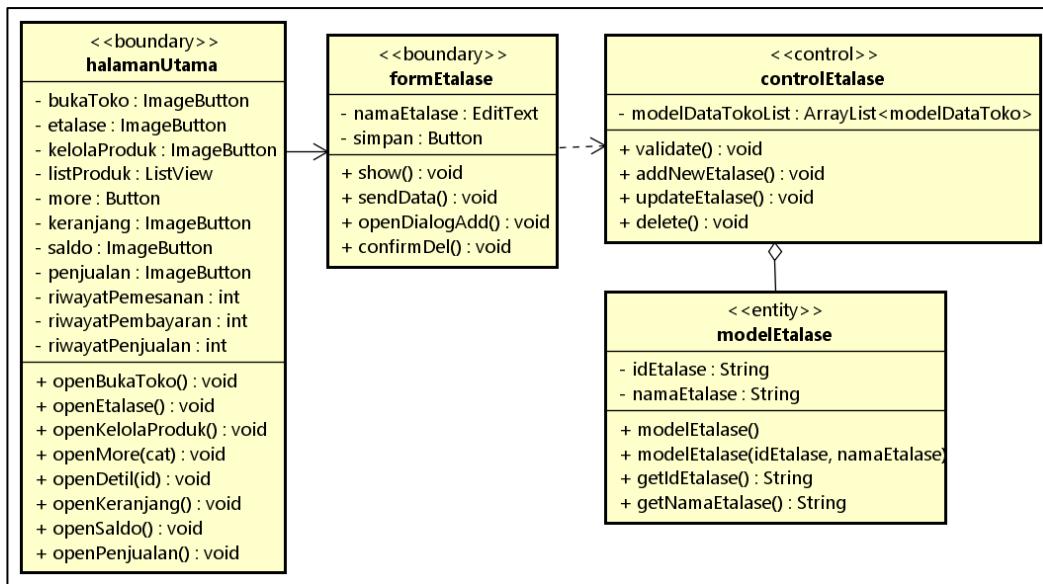


Gambar 3.40 *Class diagram* mengelola saldo

Class diagram pengguna mengelola saldo akan menjelaskan tentang hubungan antara modelSaldo yang memiliki hubungan agregasi dengan controlSaldo. Entitas yang berhubungan dengan controlSaldo akan ditampilkan pada halamanSaldo.

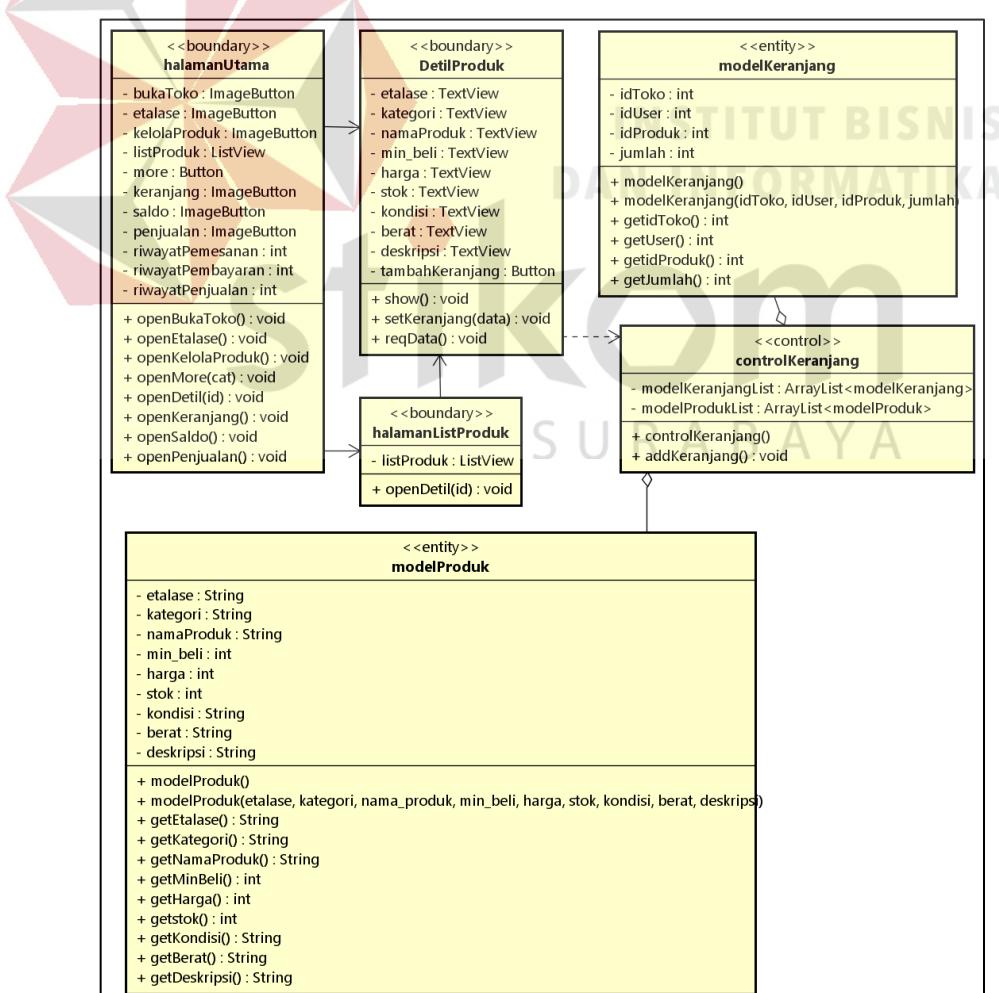
#### e. *Class Diagram Penjual Mengelola Etalase*

Class diagram penjual mengelola etalase akan menjelaskan tentang hubungan antara modelEtalase yang memiliki hubungan agregasi dengan controlEtalase. Entitas yang berhubungan dengan controlEtalase akan ditampilkan pada halaman utama. *Class diagram* penjual mengelola etalase dapat dilihat pada gambar 3.41.



Gambar 3.41 Class diagram penjual mengelola etalase

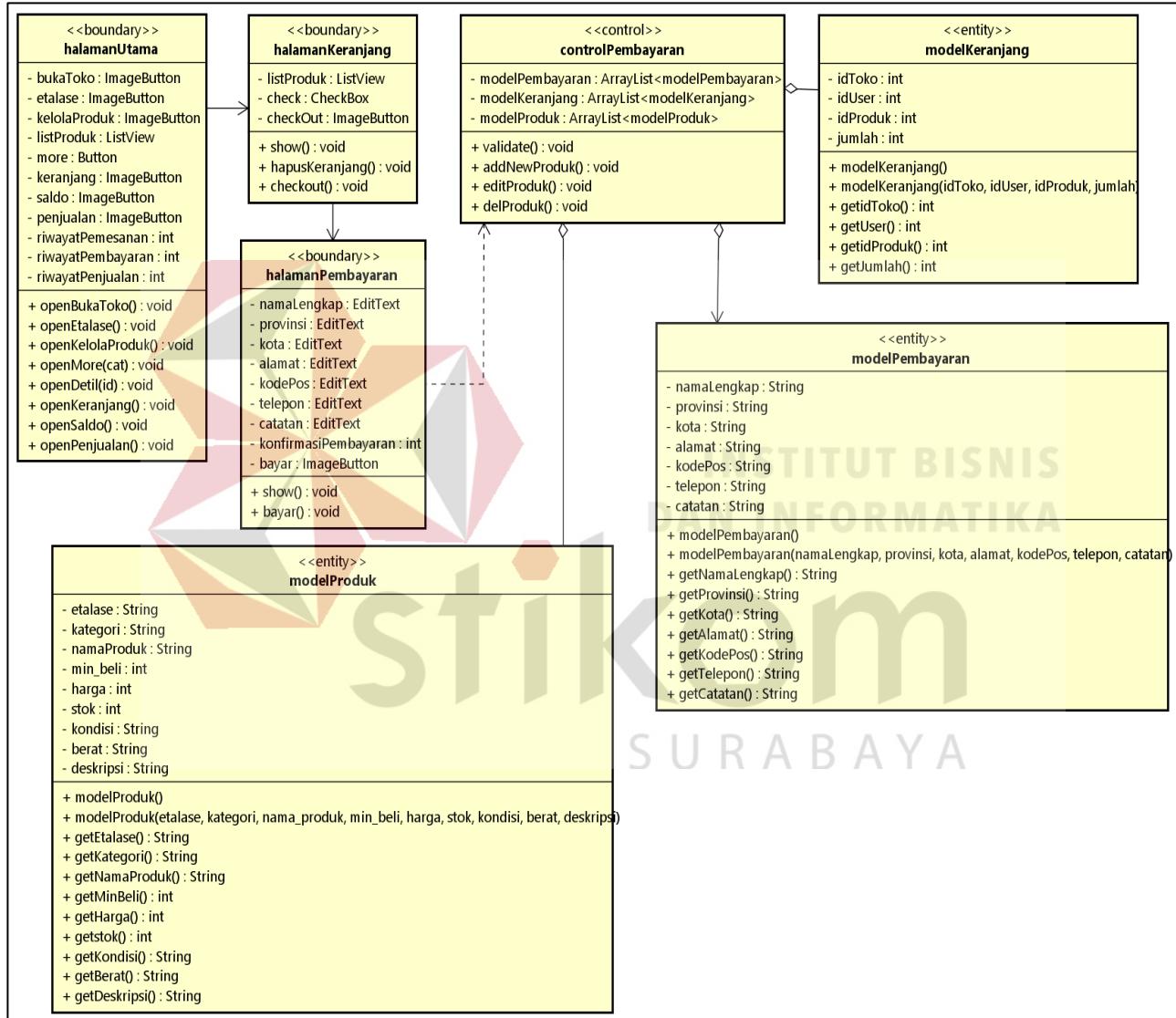
#### f. Class Diagram Penjual Melakukan Pemesanan



Gambar 3.42 Class diagram penjual melakukan pemesanan

Class diagram penjual melakukan pemesanan akan menjelaskan tentang hubungan antara modelProduk dan modelKeranjang yang memiliki hubungan agregasi dengan controlProduk. Entitas yang berhubungan dengan controlProduk akan ditampilkan pada halamanListProduk.

### g. Class Diagram Pembeli Melakukan Pembelian

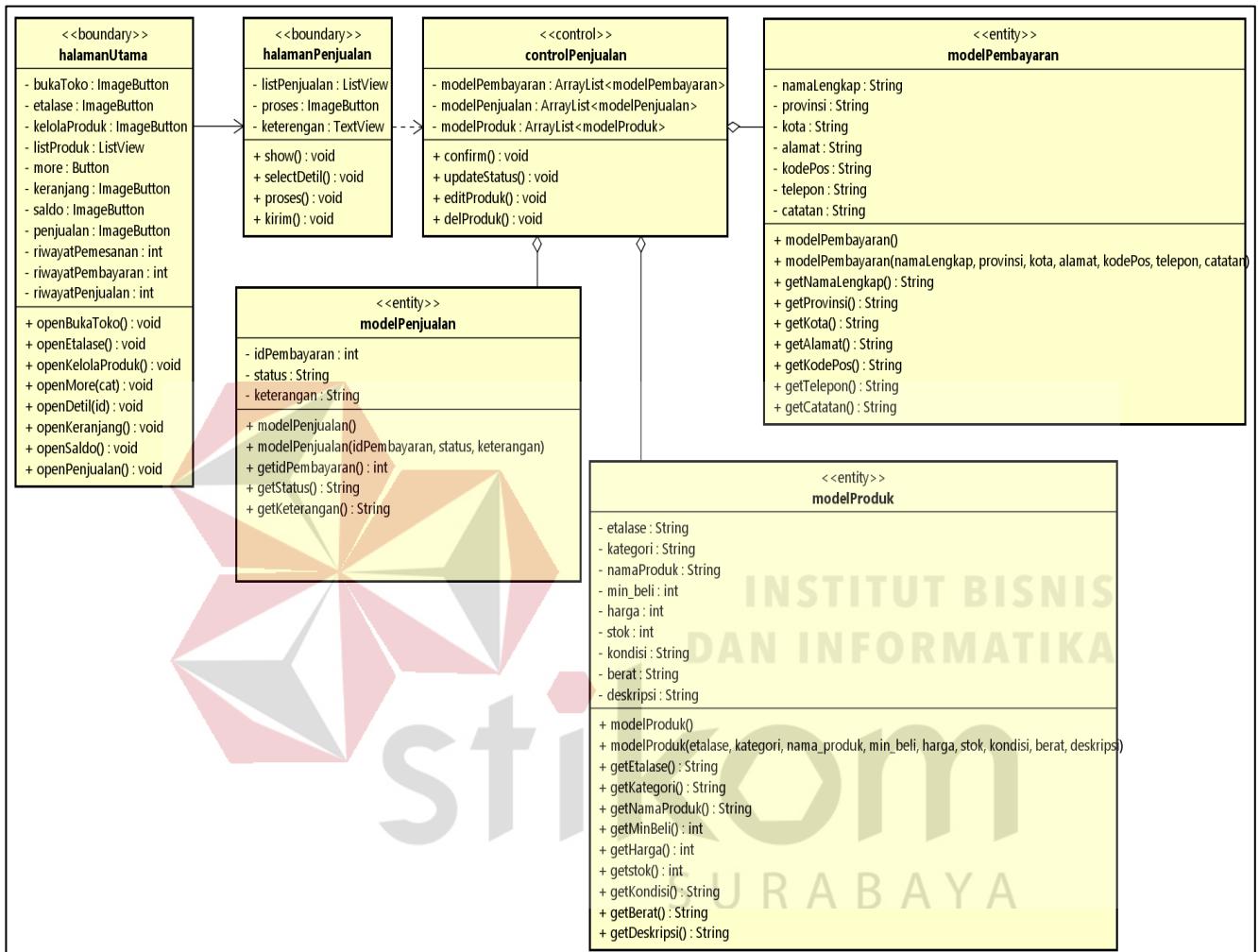


Gambar 3.43 class diagram penjual melakukan pembelian

Class diagram penjual melakukan pembelian akan menjelaskan tentang hubungan antara modelProduk yang memiliki hubungan agregasi dengan

controlPembayaran. Entitas yang berhubungan dengan controlPembayaran akan ditampilkan pada halamanPembayaran.

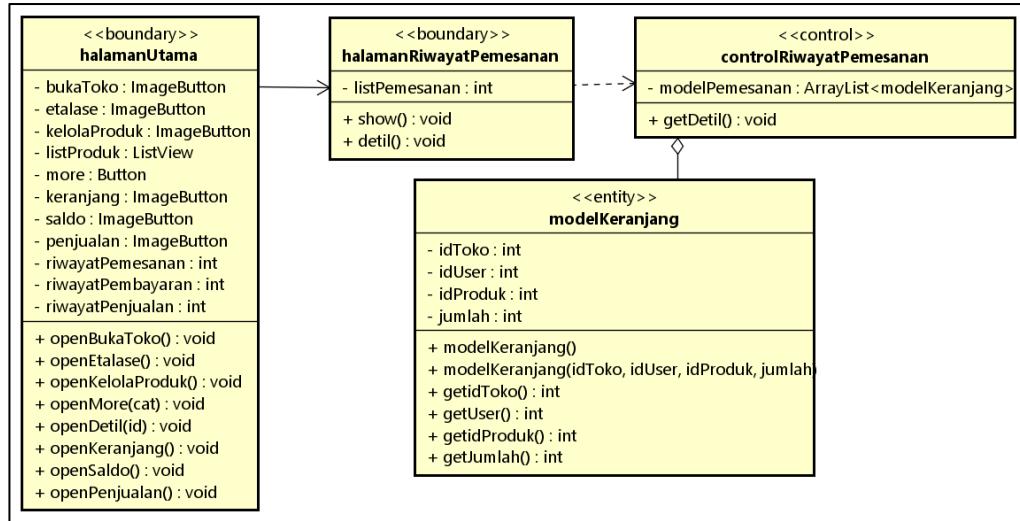
### 3.3.1 Class Diagram Penjual Melakukan Penjualan



Gambar 3.44 class diagram penjual melakukan penjualan

Class diagram penjual melakukan penjualan akan menjelaskan tentang hubungan antara model keranjang yang memiliki hubungan agregasi dengan controlPenjualan. Entitas yang berhubungan dengan controlPenjualan akan ditampilkan pada halamanPenjualan.

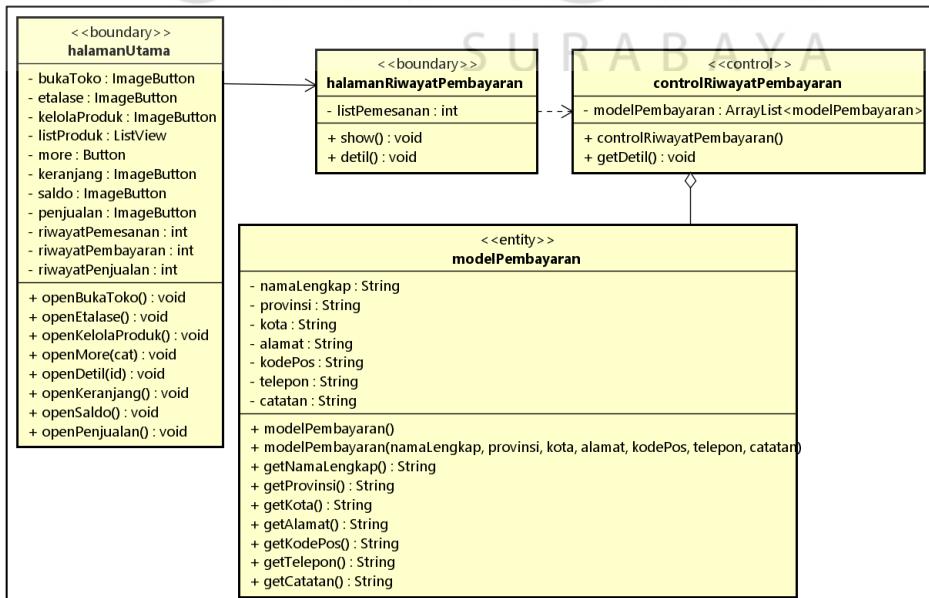
### i. Class diagram pengolahan laporan pemesanan



Gambar 3.45 class diagram pengolahan laporan pemesanan

Class diagram pengolahan laporan pemesanan akan menjelaskan tentang hubungan antara modelKeranjang yang memiliki hubungan agregasi dengan controlRiwayatPemesanan. Entitas yang berhubungan dengan controlRiwayatPemesanan akan ditampilkan pada halamanRiwayatPemesanan.

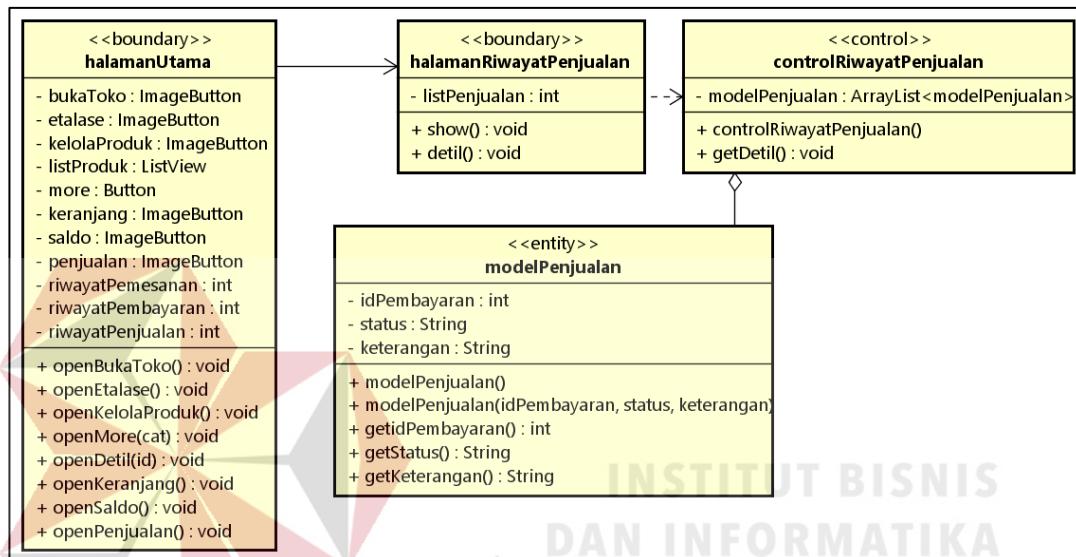
### j. Class diagram pengolahan laporan pembayaran



Gambar 3.46 class diagram pengolahan laporan pembayaran

Class diagram pengolahan laporan pembayaran akan menjelaskan tentang hubungan antara modelPembayaran yang memiliki hubungan agregasi dengan controlRiwayatPembayaran. Entitas yang berhubungan dengan controlRiwayatPembayaran akan ditampilkan pada halamanRiwayatPembayaran.

#### k. *Class diagram pengolahan laporan penjualan*



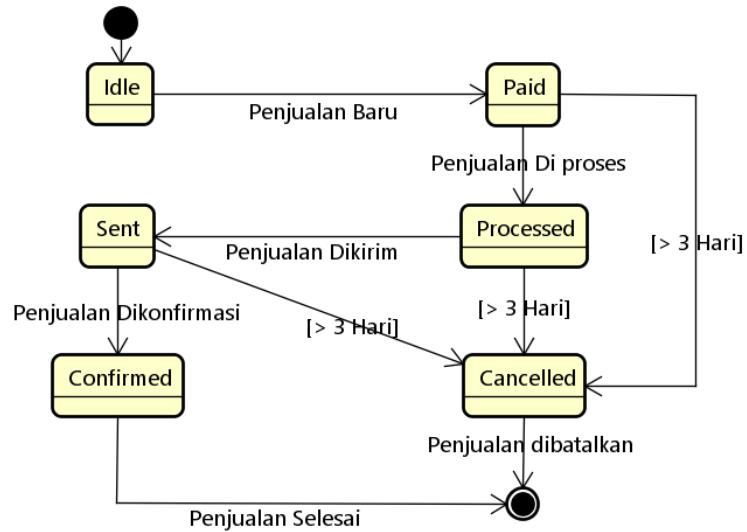
Gambar 3.47 *class diagram* pengolahan laporan penjualan

Class diagram pengolahan laporan penjualan akan menjelaskan tentang hubungan antara modelPenjualan yang memiliki hubungan agregasi dengan controlRiwayatPenjualan. Entitas yang berhubungan dengan controlRiwayatPenjualan akan ditampilkan pada halamanRiwayatPenjualan.

#### 3.3.8 Statechart diagram

##### 1. *Statechart diagram class modelPenjualan*

Berdasarkan *Statechart diagram* class modelPenjualan menunjukkan kondisi suatu class objek dimulai dari *paid* (dibayar), *processed* (diproses), *sent* (dikirim), *confirmed* (dikonfirmasi). *Statechart diagram* class modelPenjualan dapat dilihat pada gambar 3.48.

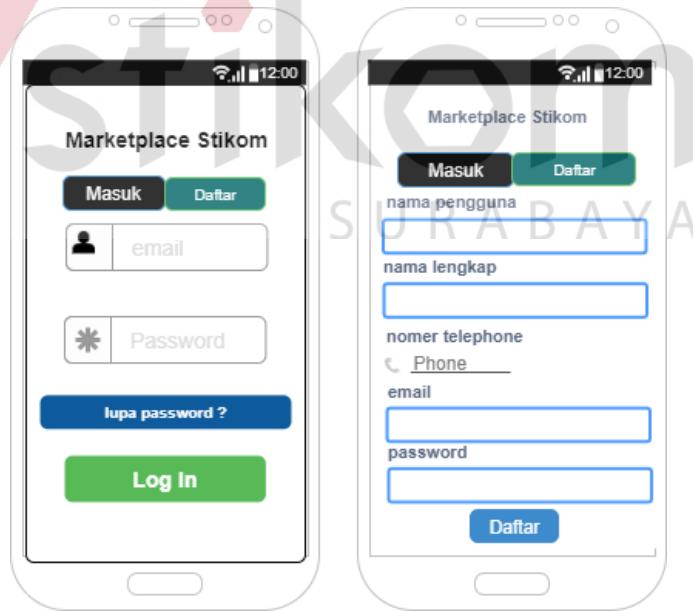


Gambar 3. 48 Statechart diagram class modelPenjualan.

### 3.3.9 Desain antarmuka pengguna

Desain antarmuka pengguna adalah desain aplikasi yang berfokus pada pengalaman pengguna. Tujuannya adalah untuk membuat interaksi pengguna menjadi sederhana dan efisien dalam mencapai tujuan pengguna.

#### a. Login dan Membuat Akun Baru



Gambar 3.43 Desain antarmuka pengguna Login dan Membuat Akun Baru

Desain antarmuka pengguna login dan membuat akun baru digunakan oleh penjual dan pembeli, pada halaman ini akan dilakukan validasi akses menggunakan

email dan password. Untuk pengguna yang belum memiliki akun dapat membuat akun baru dengan mengisi form pada halaman pendaftaran untuk akun baru.

### b. Halaman Utama



Gambar 3.44 Desain antarmuka pengguna Halaman Utama

Desain antarmuka pengguna halaman utama digunakan oleh penjual dan pembeli. Pada halaman ini akan ditampilkan seluruh produk yang ada pada marketplace stikom. Pengguna dapat melihat detail produk, harga produk, dan dapat melakukan pembelian produk, pengguna juga dapat mencari produk yang diinginkan.

### c. Penjual Membuka Toko

Desain antarmuka pengguna penjual membuka toko digunakan oleh penjual untuk membuka toko. Sebelum membuat toko penjual dipastikan harus login terlebih dahulu. Selanjutnya penjual harus mengisi form untuk membuat toko, setelah toko dibuat penjual akan mendapatkan akses untuk menjual produk pada

marketplace stikom. Penjual dapat melihat produk, etalase, pesanan tahap di proses, dan pesanan yang telah selesai. Desain antarmuka pengguna Penjual Membuka Toko dapat dilihat pada gambar 3.45.



Gambar 3.45 Desain antarmuka pengguna Penjual Membuka Toko

#### d. Penjual Mengelola Produk

Desain antarmuka pengguna penjual mengelola produk digunakan oleh penjual untuk menambah produk baru untuk dijual. Sebelum dapat menambahkan produk, penjual harus membuat etalase dahulu baru dapat memasukkan produk ke

toko yang dimiliki. Desain antarmuka penjual mengelola produk dapat dilihat pada gambar 3.48.



Gambar 3.49 Desain antarmuka penjual mengelola produk

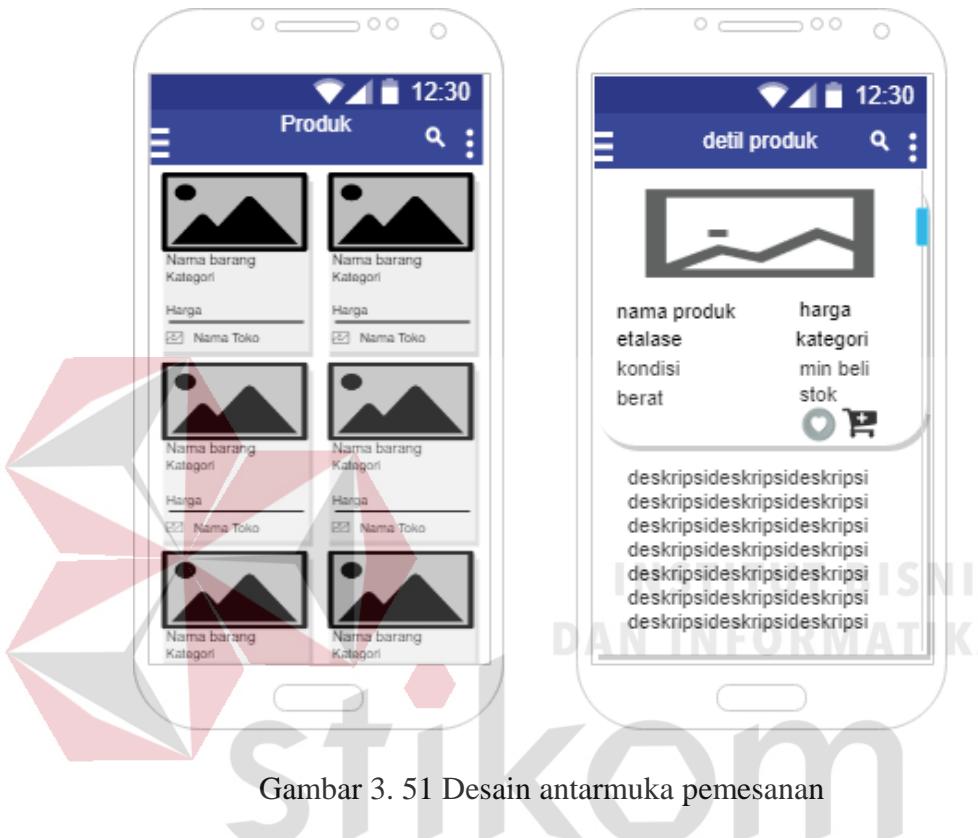
#### e. Penjual Mengelola Etalase



Gambar 3.50 Desain antarmuka mengelola etalase

Desain antarmuka pengguna penjual mengelola etalase digunakan oleh penjual untuk menambah etalase baru. Setelah membuat etalase pengguna dapat menambahkan produk sesuai dengan etalase yang telah tersedia.

#### f. Pembeli Melakukan Pemesanan

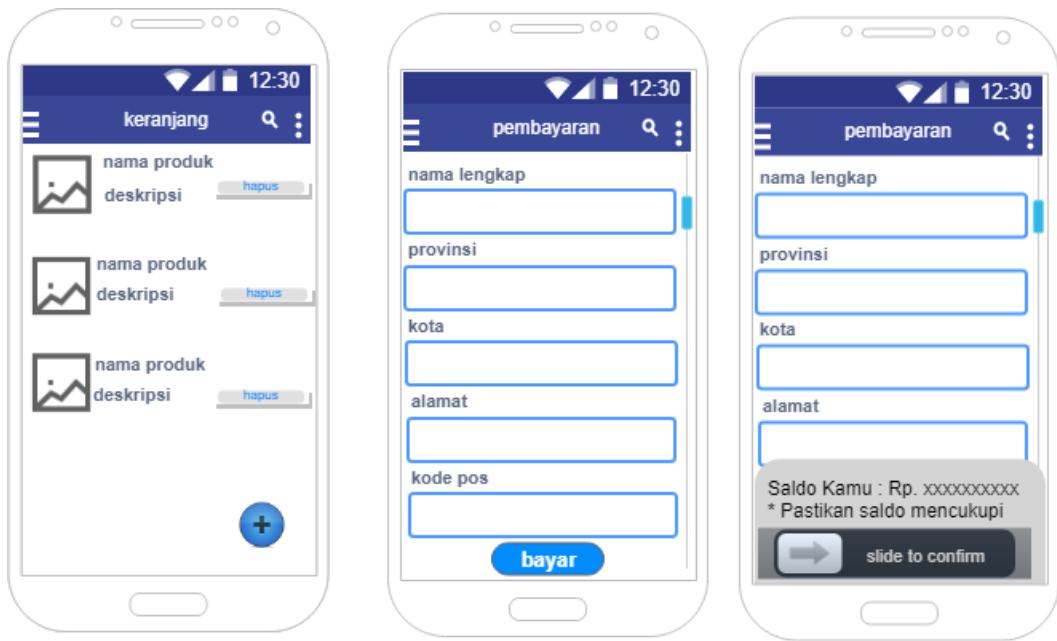


Gambar 3. 51 Desain antarmuka pemesanan

Desain antarmuka pengguna pembeli melakukan pemesanan digunakan oleh pembeli untuk menambahkan produk ke dalam keranjang dan nantinya akan dilakukan proses *checkout* agar bisa melakukan pembelian barang.

#### g. Pembeli Melakukan Pembelian dan Pembayaran

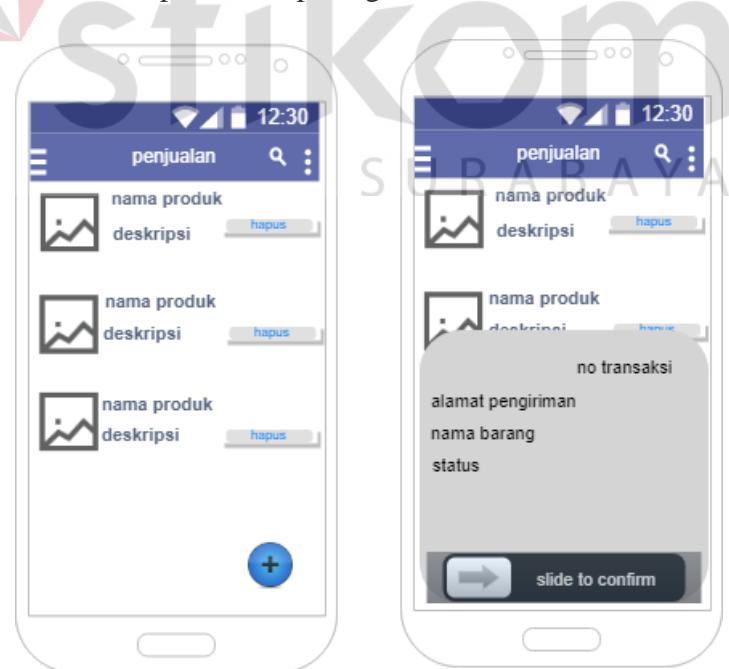
Desain antarmuka pembeli melakukan pembelian dan pembayaran digunakan oleh pembeli untuk mengkonfirmasi barang yang akan dibeli. Pembeli memasukkan alamat pengiriman barang dan setelah alamat pembeli sudah lengkap pembeli dapat melakukan proses pembayaran. Desain antarmuka pembelian dan pembayaran dapat dilihat pada gambar 3.51.



Gambar 3.52 Desain antarmuka pembelian dan pembayaran

#### **h. Penjual Melakukan Penjualan**

Desain antarmuka penjual melakukan penjualan digunakan penjual untuk memproses transaksi yang telah dilakukan pembeli. Desain antarmuka penjual melakukan pembelian dapat dilihat pada gambar 3.52



Gambar 3.53 Desain antarmuka penjualan

### i. Pengolahan Laporan Pemesanan Dari Pembeli

Desain antarmuka pengolahan laporan pemesanan dari pembeli digunakan pembeli untuk melihat riwayat transaksi pemesanan. Desain antarmuka pengolahan laporan pemesanan dari pembeli dapat dilihat pada gambar 3.53.



Gambar 3.54 Desain antarmuka laporan pemesanan

### j. Pengolahan Laporan Pembayaran Dari Pembeli

Desain antarmuka pengolahan laporan pembayaran dari pembeli digunakan pembeli untuk melihat riwayat transaksi pembayaran. Desain antarmuka pengolahan laporan pembayaran dari pembeli dapat dilihat pada gambar 3.54.



Gambar 3.55 Desain antarmuka laporan pembayaran

#### k. Pengolahan Laporan Penjualan Dari Penjual

Desain antarmuka pengolahan laporan penjualan dari penjual digunakan penjual untuk melihat riwayat transaksi penjualan. Desain antarmuka pengolahan laporan penjualan dari penjual dapat dilihat pada gambar 3.54.



Gambar 3. 56 Desain antarmuka laporan penjualan

### 3.3.10 Desain uji coba

Pengujian dilakukan dengan pembuatan test plan berdasarkan fungsi serta urutan dari penyelesaian masalah pada aplikasi. Pengujian dilakukan menggunakan metode black box testing dengan mengamati hasil keluaran tiap fungsi yang diuji coba.

#### a. Login Dan Membuat Akun Baru

Tabel 3. 16 Desain uji coba login dan membuat akun baru

No	Tujuan	Input	Output yang diharapkan
<b>Login</b>			
1	Menguji validasi isian user yang terdaftar	Email dan password benar	Menampilkan halaman utama
2	Menguji validasi isian yang tidak terdaftar	Email dan password salah	Menampilkan pesan kesalahan
<b>Daftar</b>			
4	Menguji validasi isian	Text box isian sudah terisi semua	Menampilkan pesan apabila terdapat isian yang belum diisi
5	Menguji tombol simpan	Tombol simpan	Data akun baru tersimpan pada database

#### b. Pembeli Melakukan Pemesanan

Tabel 3. 17 Desain uji coba pembeli melakukan pemesanan

No	Tujuan	Input	Output yang diharapkan
1	Menguji tombol detail	Tombol detail	Menampilkan halaman detail produk yang dipesan
2	Menguji tombol tambah	Tombol tambah	Menampilkan halaman form tambah produk
3	Menguji tombol keranjang	Tombol keranjang	Data tersimpan pada database keranjang dan akan tampil pada halaman keranjang

#### c. Pembeli Melakukan Pembelian

Tabel 3. 18 Desain uji coba pembeli melakukan pembelian

No	Tujuan	Input	Output yang diharapkan
1	Menampilkan data keranjang		Data keranjang tampil sesuai dengan yang telah diinputkan
2	Menguji tombol bayar	Tombol bayar	Data pembayaran tersimpan pada database

#### d. Penjual Melakukan Penjualan

Tabel 3. 19 Desain uji coba penjual melakukan penjualan

No	Tujuan	Input	Output yang diharapkan
1	Menampilkan data penjualan		Data penjualan tampil sesuai dengan database
2	Menguji tombol detail	Tombol detail	Menampilkan halaman detail transaksi yang dilakukan
3	Menguji tombol slide to slide to confirm	Tombol slide to confirm	Data transaksi tersimpan pada database

#### e. Penjual Membuka Toko

Tabel 3. 20 Desain uji coba penjual membuka toko

No	Tujuan	Input	Output yang diharapkan
1	Menguji validasi isian	Text box isian sudah terisi semua	Menampilkan pesan apabila terdapat isian yang belum diisi
2	Menguji tombol simpan	Tombol simpan	Data buka toko tersimpan pada database

#### f. Penjual Mengelola Etalase

Tabel 3. 21 Desain uji coba penjual mengelola etalase

No	Tujuan	Input	Output yang diharapkan
1	Menampilkan data etalase		Data etalase tampil sesuai dengan yang telah disimpan pada database
2	Menguji tombol edit	Tombol edit	Data etalase teredit dan muncul form data etalase yang akan diedit
3	Menguji tombol hapus	Tombol hapus	Data produk terhapus
4	Menguji tombol simpan	Tombol simpan	Data etalase tersimpan pada database dan tampil pada halaman etalase

#### g. Penjual Mengelola Produk

Tabel 3. 22 Desain uji coba penjual mengelola produk

No	Tujuan	Input	Output yang diharapkan
1	Menampilkan data produk		Data produk tampil sesuai dengan yang telah diinputkan
2	Menguji tombol edit	Tombol edit	Muncul form data produk yang akan diedit
3	Menguji tombol hapus	Tombol hapus	Data produk terhapus

4	Menguji tombol tambah	Tombol tambah	Menampilkan halaman form tambah produk
5	Menguji validasi isian	Text box isian sudah terisi semua	Menampilkan pesan apabila terdapat isian yang belum diisi
6	Menguji tombol simpan	Tombol simpan	Data buka toko tersimpan pada database dan tampil pada halaman produk

#### **h. Pengolahan Laporan Pemesanan Dari Pembeli**

Tabel 3. 23 Desain uji coba pengolahan laporan pemesanan dari pembeli

No	Tujuan	Input	Output yang diharapkan
1	Menampilkan data laporan pemesanan		Data laporan pemesanan tampil sesuai dengan database
2	Menguji tombol detail	Tombol detail	Menampilkan halaman detail laporan penjualan yang dilakukan

#### **i. Pengolahan Laporan Pembayaran Dari Pembeli**

Tabel 3. 24 Desain uji coba pengolahan laporan pembayaran dari pembeli

No	Tujuan	Input	Output yang diharapkan
1	Menampilkan data laporan pembayaran		Data laporan pembayaran tampil sesuai dengan database
2	Menguji tombol detail	Tombol detail	Menampilkan halaman detail laporan pembayaran yang dilakukan

#### **j. Pengolahan Laporan Penjualan Dari Penjual**

Tabel 3. 25 Desain uji coba laporan penjualan dari penjual

No	Tujuan	Input	Output yang diharapkan
1	Menampilkan data laporan penjualan		Data laporan penjualan tampil sesuai dengan database
2	Menguji tombol detail	Tombol detail	Menampilkan halaman detail laporan penjualan yang dilakukan

## **BAB IV**

### **IMPLEMENTASI DAN EVALUASI**

#### **4.1. Implementasi**

##### **4.1.1. Implementasi Kebutuhan Software dan Hardware**

Kebutuhan software dan hardware yang akan digunakan pada pengaplikasian rancang bangun aplikasi marketplace stikom surabaya berbasis android dari sisi client maupun server dapat dilihat pada tabel 4.1

Tabel 4. 1 Kebutuhan *Software* dan *Hardware*

No	Kebutuhan	Client	Server
1.	Software	1. Minimal windows 7/8 2. Android studio 3. SQLyog 4. Minimal OS Android Lollipop (API 21)	1. Minimal windows 7 2. Database MySQL 3. Web Server Apache 4. Programming Language PHP
2.	Hardware	1. Minimal RAM 4 (Disarankan RAM 8) 2. Minimal hardisk 2 GB (Disarankan 4 GB) 3. Minimal RAM 1 GB	1. Minimal RAM 2 GB (Disarankan RAM 4) 2. Minimal Kapasitas Harddisk yang tersedia 10 GB
3.	Network	Gigabit Network	

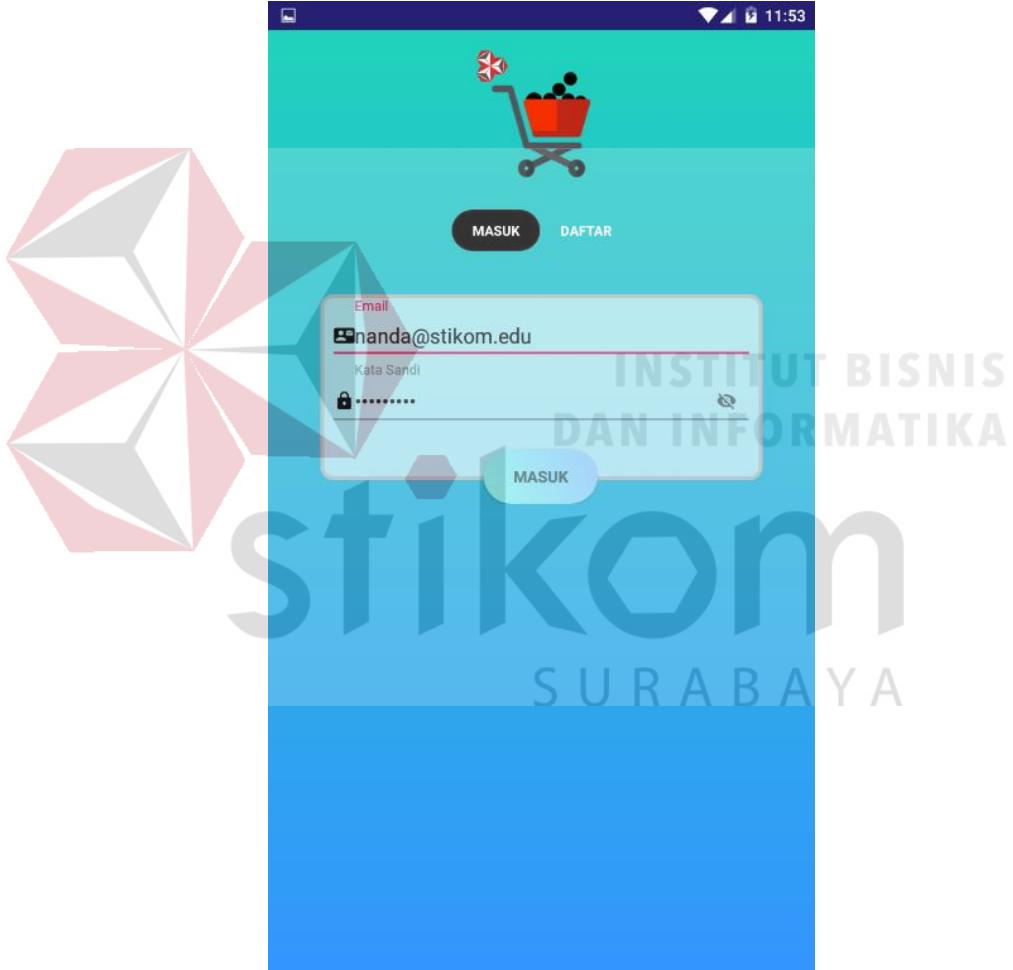
##### **4.1.2. Implementasi Sistem**

Pada bab ini akan menjelaskan implementasi dan evaluasi dari rancang bangun aplikasi marketplace marketplace stikom berbasis android. Antarmuka/interface akan dijelaskan untuk mempermudah pengguna mengenal fitur, cara kerja, dan alur dari aplikasi marketplace stikom surabaya berbasis android.

#### 4.1.3. Tampilan halaman login dan membuat akun baru

##### a. Tampilan login

Tampilan login muncul pada saat user mencoba mengakses tampilan yang memerlukan login. Halaman login berfungsi untuk mengidentifikasi pengguna dengan memasukkan email dan kata sandi yang telah terdaftar sebelumnya. Tampilan halaman login dapat dilihat pada gambar 4.1.

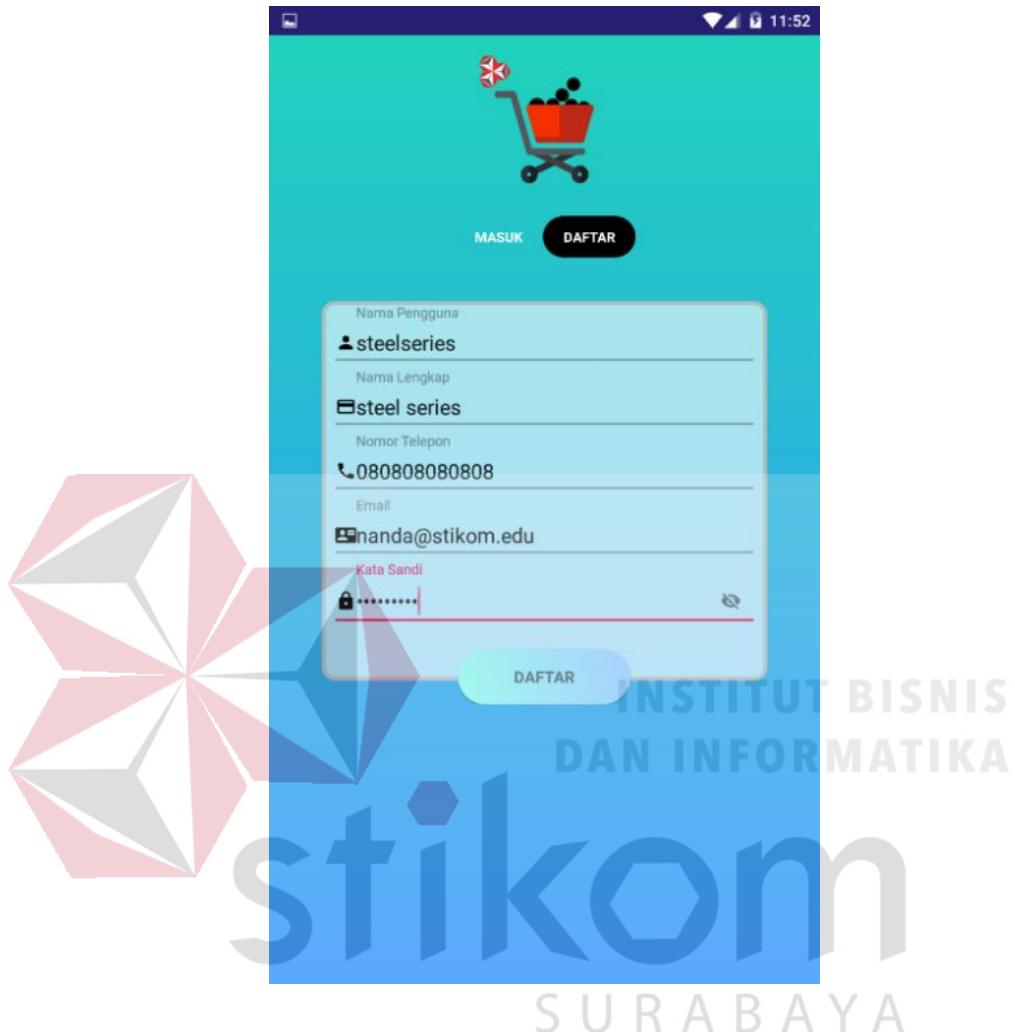


Gambar 4.1 Tampilan halaman login

##### b. Tampilan membuat akun baru

Tampilan membuat akun baru digunakan untuk mendaftarkan pengguna baru agar dapat melakukan transaksi pada aplikasi. Form membuat akun baru berisi

inputan seperti nama pengguna nama lengkap, nomor telepon, email, dan kata sandi seperti pada gambar 4.2.



Gambar 4.2 Tampilan halaman membuat akun baru

#### 4.1.4. Tampilan halaman utama

##### a. Tampilan halaman utama aplikasi

Halaman utama aplikasi adalah halaman yang menampilkan banner dari aplikasi dan juga list produk terbaru. Halaman ini bisa diakses tanpa harus login. Tampilan halaman utama dapat dilihat pada gambar 4.3.



Gambar 4.3 Tampilan halaman utama aplikasi

#### b. Tampilan detail produk

Tampilan detail produk adalah tampilan yang muncul ketika pengguna menekan salah satu produk yang tersedia. Halaman ini menampilkan gambar dari produk, nama lengkap produk, harga produk, dan deskripsi produk. Pada halaman ini pengguna juga dapat menyukai produk dengan menekan tombol hati, menambahkan produk ke keranjang, dan melihat informasi toko pemilik produk.

Tampilan detail produk dapat dilihat pada halaman 4.4.



Gambar 4.4 Tampilan detail produk

### c. Tampilan informasi toko

Tampilan informasi toko muncul saat pengguna menekan tombol informasi toko pada tampilan detail produk. Tampilan informasi produk menampilkan informasi toko, produk apa saja yang dimiliki toko, dan etalase apa saja yang dimiliki toko. Tampilan informasi toko dapat dilihat pada gambar 4.5.



Gambar 4.5 Tampilan informasi toko

#### 4.1.5. Tampilan halaman membuka toko

Tampilan halaman membuka toko muncul ketika pengguna mencoba mengakses toko saya pada tombol navigasi dengan kondisi pengguna belum memiliki toko di marketplace stikomart.

##### a. Tampilan pop up informasi belum punya toko

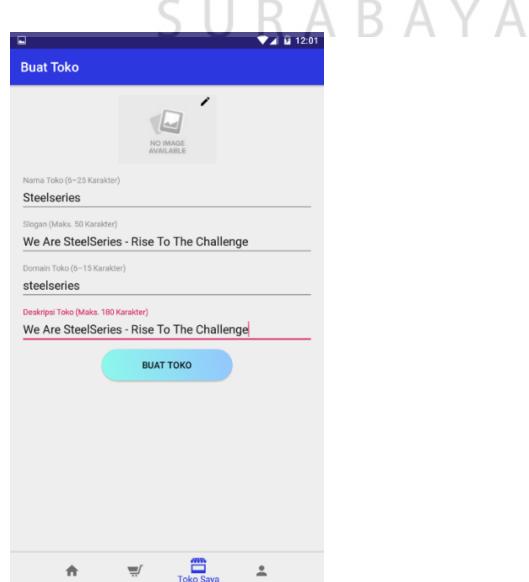
Tampilan pop up informasi belum punya toko muncul ketika pengguna mencoba mengakses halaman toko saya. Tampilan pop up dapat dilihat pada gambar 4.6.



Gambar 4.6 Tampilan *pop up* belum buka toko

##### b. Tampilan form buka toko

Tampilan form buka toko digunakan pengguna untuk membuat toko yang dapat digunakan untuk transaksi. Tampilan form buka toko dapat terlihat pada gambar 4.7.



Gambar 4.7 Tampilan form buka toko

#### 4.1.6. Tampilan dashboard penjual

##### a. Tampilan utama dashboard penjual

Tampilan utama dashboard penjual dapat muncul ketika pengguna telah membuat toko sebelumnya. Pada tampilan ini terdapat beberapa informasi mengenai nama toko dan slogan toko, jumlah produk, jumlah etalase, jumlah pesanan yang sedang tahap proses, dan total seluruh pesanan. Pada tampilan ini juga terdapat list pesanan baru sehingga penjual dapat memproses pesanan lebih mudah.



Gambar 4.8 Tampilan dashboard penjual

##### b. Tampilan list pesanan

Tampilan list pesanan dapat dilihat ketika pengguna menekan pada bagian pesanan tahap proses ataupun pada pesanan. Pada tampilan list pesanan terdapat sub menu seperti semua, baru, proses, kirim dan selesai sesuai dengan status pada pesanan tersebut. Tampilan list pesanan dapat dilihat pada gambar 4.9.



Gambar 4.9 Tampilan list pesanan

### c. Tampilan terima pesanan

Dengan menekan satu kali pada salah satu list akan menampilkan detail dari pesanan tersebut. Isi dari detail pesanan akan menyesuaikan status dari pesanan tersebut. Seperti pada gambar 4.10 menunjukkan detail dari pesanan baru yang menunggu proses dari penjual. Penjual dapat memproses pesanan dengan cara memilih terima dan konfirmasi proses terima pesanan dengan cara menggeser slider.



Gambar 4.10 Tampilan terima pesanan

#### d. Tampilan tolak pesanan

Penjual juga dapat menolak pesanan yang diterima dengan memilih batalkan dan konfirmasi batal dengan cara menggeser *slider* seperti pada gambar 4.11.



Gambar 4.11 Tampilan tolak pesanan

#### e. Tampilan kirim pesanan

Hampir sama dengan tampilan sebelumnya, ketika penjual telah memproses pesanan penjual dapat melakukan proses pengiriman kepada pembeli seperti pada gambar 4.12



Gambar 4.12 Tampilan kirim pesanan

#### 4.1.7. Tampilan halaman penjual mengelola etalase

##### a. Tampilan list etalase

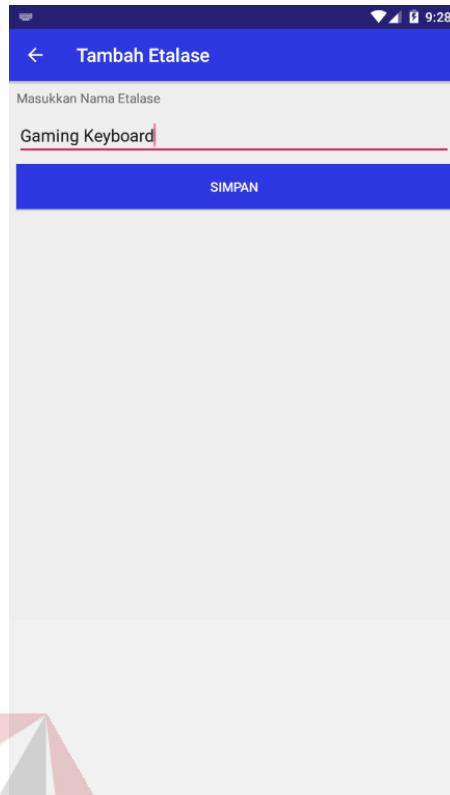
Tampilan list etalase muncul ketika pengguna menekan pada bagian etalase di dashboard penjual. Tampilan list etalase menampilkan data etalase yang dimiliki oleh penjual, serta pada masing masing list menunjukkan nama etalase dan tanggal pembuatan etalase. Penjual dapat menambahkan etalase baru dengan menekan tombol tambah pada bagian kanan bawah seperti pada gambar 4.13.



Gambar 4.13 Tampilan list etalase

##### b. Tampilan tambah etalase

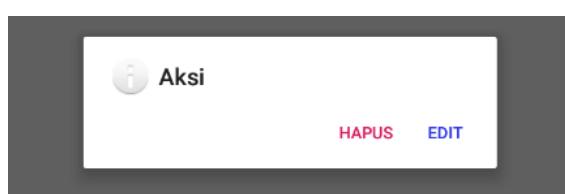
Tampilan tambah etalase muncul pada saat pengguna menekan tombol tambah pada tampilan list etalase. Pada tampilan ini terdapat satu isian yaitu nama etalase yang nantinya akan digunakan pada saat menambah produk. Tampilan tambah etalase dapat dilihat pada gambar 4.14.



Gambar 4.14 Tampilan tambah etalase

**c. Tampilan pop up aksi hapus dan edit**

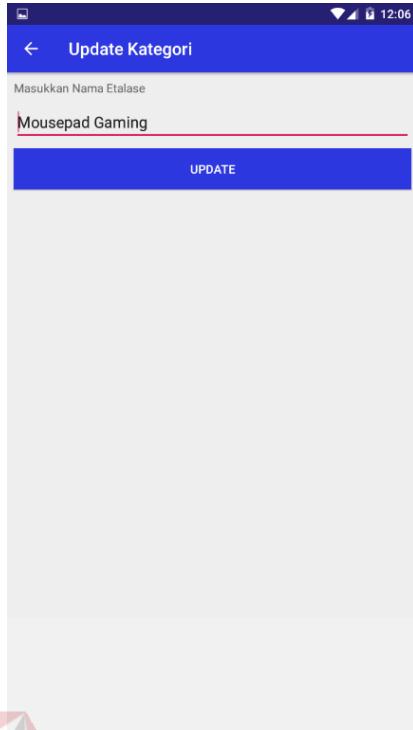
Tampilan pop up aksi hapus dan edit muncul ketika pengguna menekan lama pada salah satu list. Tampilan pop up aksi hapus dan edit dapat dilihat pada gambar 4.15.



Gambar 4. 15 Tampilan *pop up* aksi hapus dan edit

**d. Tampilan edit etalase**

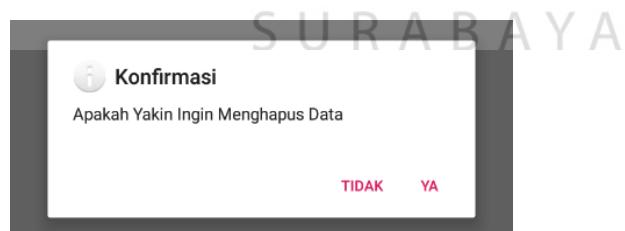
Tampilan edit etalase muncul ketika pengguna memilih edit pada tampilan pop up aksi hapus dan edit. Tampilan form edit etalase dapat dilihat pada gambar 4.16.



Gambar 4.16 Tampilan edit etalase

**e. Tampilan konfirmasi hapus etalase**

Tampilan konfirmasi hapus etalase muncul ketika pengguna memilih hapus pada tampilan pop up aksi hapus dan edit. Tampilan konfirmasi hapus etalase dapat dilihat pada gambar 4.17.

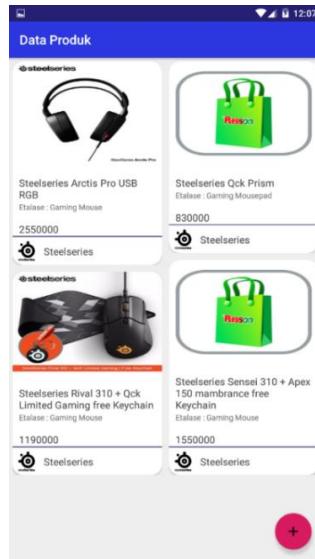


Gambar 4.17 Tampilan konfirmasi hapus etalase

**4.1.8. Tampilan halaman penjual mengelola produk**

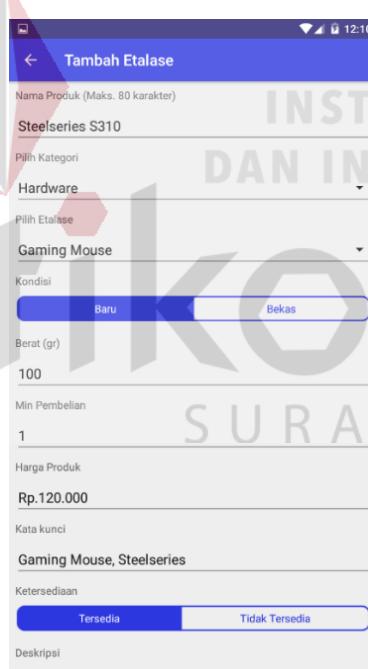
**a. Tampilan list produk**

Tampilan list produk penjual dapat dimunculkan ketika pengguna menekan pada bagian produk di tampilan dashboard penjual. Tampilan list produk penjual menampilkan list produk yang dimiliki oleh penjual seperti pada gambar 4.18.



Gambar 4.18 Tampilan list produk

### b. Tampilan tambah produk



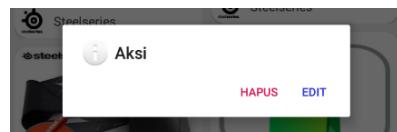
Gambar 4.19 Tampilan tambah produk

Tampilan tambah produk muncul ketika pengguna menekan tombol tambah pada tampilan list produk. Pada tampilan tambah produk terdapat inputan seperti nama produk, kategori produk, etalase produk sesuai dengan yang dimiliki toko, kondisi ketersediaan produk, berat dari produk, minimal pembelian produk, harga

produk, kata kunci produk, ketersediaan produk, dan deskripsi dari produk yang dijual.

#### c. Tampilan pop up aksi hapus dan edit

Tampilan konfirmasi hapus produk muncul ketika pengguna memilih hapus pada tampilan pop up aksi hapus dan edit. Tampilan konfirmasi hapus produk dapat dilihat pada gambar 4.20.



Gambar 4. 20 Tampilan pop up aksi hapus dan edit

#### d. Tampilan edit produk

Tampilan edit produk muncul ketika pengguna memilih edit pada tampilan pop up aksi hapus dan edit. Tampilan form edit produk dapat dilihat pada gambar 4.21.

Nama Produk (Maks. 80 karakter)	Steelseries Sensei 310 + Apex 150 mambrance free Keychain
Pilih Kategori	Komputer
Pilih Etalase	Gaming Mouse
Kondisi	Baru
Berat (gr)	3000
Min Pembelian	1
Harga Produk	Rp.1.550.000
Kata kunci	mouse, gaming, gear, sensei 310
Ketersediaan	Tersedia
Deskripsi	

Gambar 4. 21 Tampilan edit produk

### e. Tampilan konfirmasi hapus produk

Tampilan konfirmasi hapus produk muncul ketika pengguna memilih hapus pada tampilan pop up aksi hapus dan edit. Tampilan konfirmasi hapus produk dapat dilihat pada gambar 4.22.

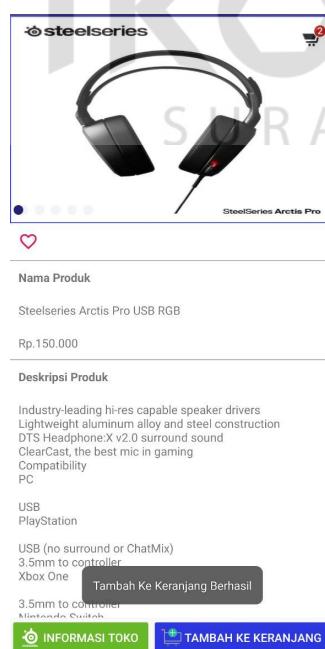


Gambar 4.22 Tampilan konfirmasi hapus produk

### 4.1.9. Tampilan halaman pembeli melakukan pemesanan

#### a. Tampilan menambahkan ke keranjang

Pengguna dapat menambahkan barang ke keranjang dengan melihat detail dari produk yang ingin ditambahkan ke keranjang. Lalu, pada detail produk pengguna dapat menekan tombol tambah ke keranjang untuk menambahkan barang ke keranjang. Tampilan dapat dilihat pada gambar 4.23.



Gambar 4.23 Tampilan Menambahkan Barang Ke Keranjang

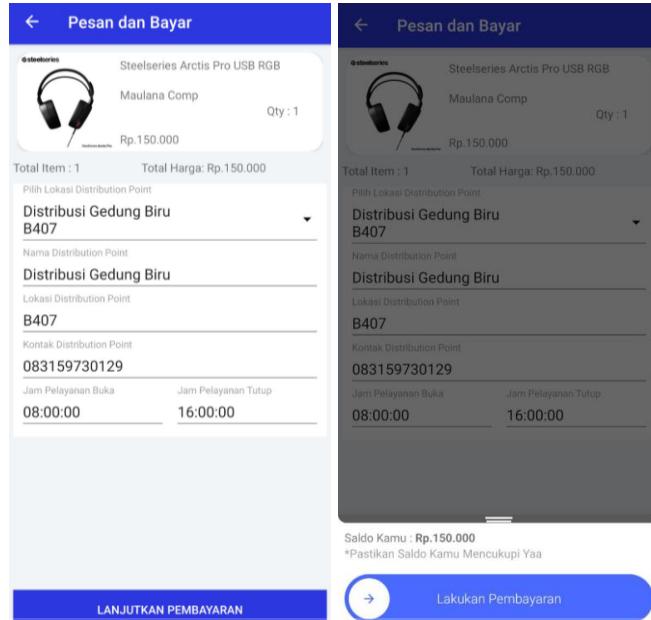
### b. Tampilan keranjang

Untuk melakukan proses pemesanan pengguna dapat memilih barang apa saja yang akan dibayarkan. Tampilan keranjang dapat dilihat pada gambar 4.23.



#### 4.1.10. Tampilan halaman pembeli melakukan pembelian

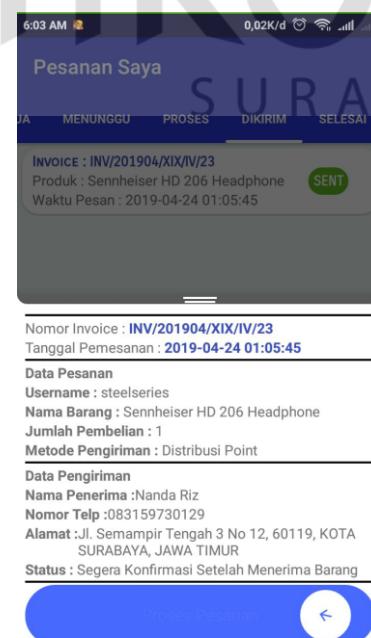
Tampilan pembeli melakukan pembelian dilakukan setelah pembeli melakukan checkout barang apa saja yang akan dipilih kemudian pengguna harus melengkapi data pada form yang tersedia. Setelah semua data terisi pengguna dapat melakukan proses pembayaran jika saldo rfid pengguna mencukupi. Tampilan halaman pembeli melakukan pembelian dapat dilihat pada gambar 4.25



Gambar 4.25 Tampilan halaman pembeli melakukan pembelian

#### 4.1.11. Tampilan halaman pembeli konfirmasi barang diterima

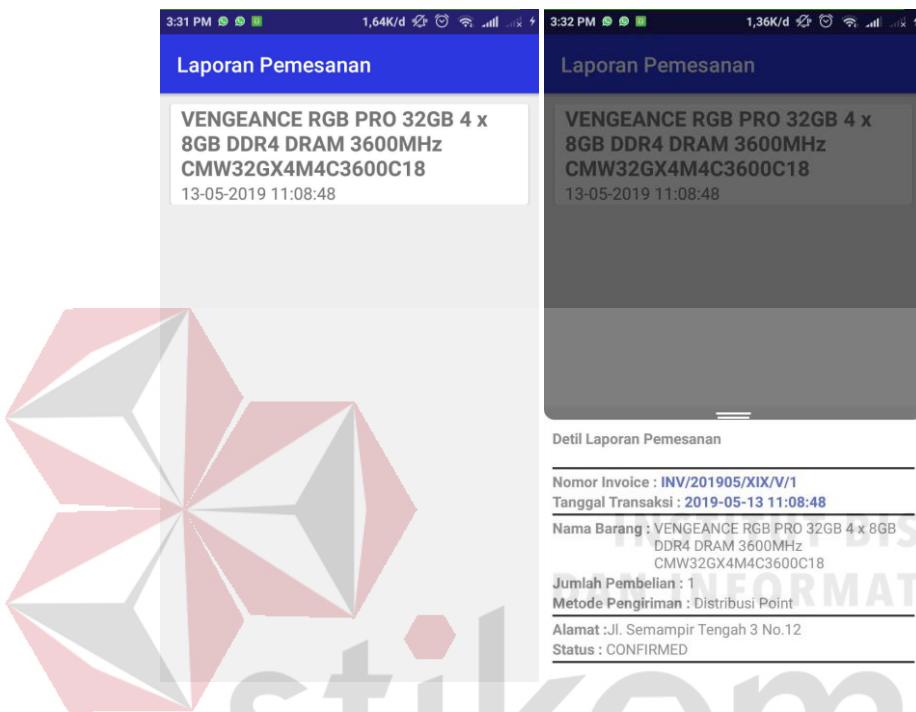
Tampilan pembeli konfirmasi barang diterima dilakukan setelah penjualan barang yang dipesan sedang dalam proses dikirim. Tampilan konfirmasi barang telah diterima dapat dilihat pada gambar 4.26.



Gambar 4.26 Tampilan konfirmasi barang diterima

#### 4.1.12. Tampilan halaman laporan riwayat pemesanan

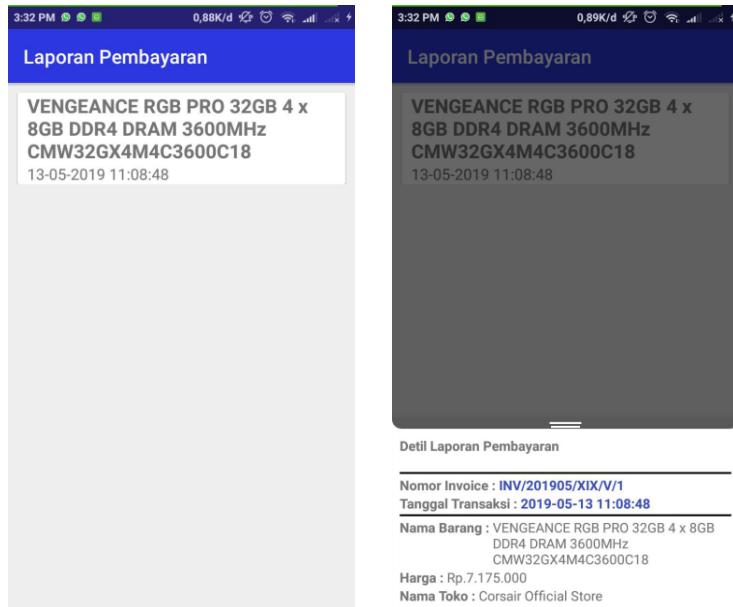
Tampilan laporan riwayat pemesanan digunakan untuk melihat riwayat pemesanan yang telah dilakukan pengguna sebagai pembeli. Tampilan laporan riwayat pemesanan dapat dilihat pada gambar 4.27.



Gambar 4.27 Tampilan laporan riwayat pemesanan

#### 4.1.13. Tampilan halaman laporan riwayat pembayaran

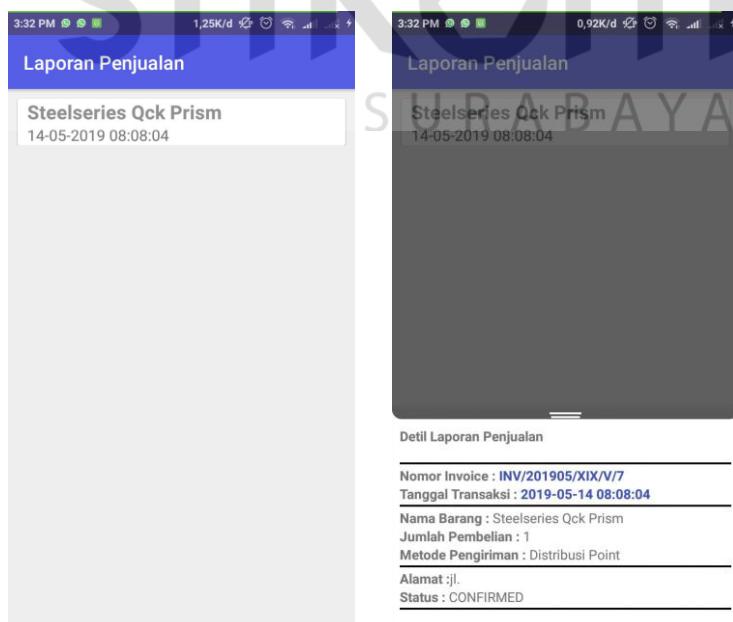
Tampilan laporan riwayat pembayaran digunakan untuk melihat riwayat pembayaran yang telah dilakukan pengguna sebagai pembeli. Tampilan laporan riwayat pembayaran dapat dilihat pada gambar 4.28.



Gambar 4.28 Tampilan laporan riwayat pembayaran

#### 4.1.14. Tampilan halaman laporan riwayat penjualan

Tampilan laporan riwayat penjualan digunakan untuk melihat riwayat penjualan yang telah dilakukan pengguna sebagai penjual. Tampilan laporan riwayat penjualan dapat dilihat pada gambar 4.29.



Gambar 4.29 Tampilan laporan riwayat penjualan

## 4.2. Evaluasi Sistem

### 4.2.1. Uji coba fungsi perangkat lunak

Berdasarkan pada perancangan uji coba, aplikasi ini akan menggunakan black box testing dalam pengujinya dan menggunakan data yang sebenarnya. Uji coba dilakukan untuk memastikan aplikasi telah dibuat dengan benar sesuai kebutuhan atau tujuan yang diharapkan.

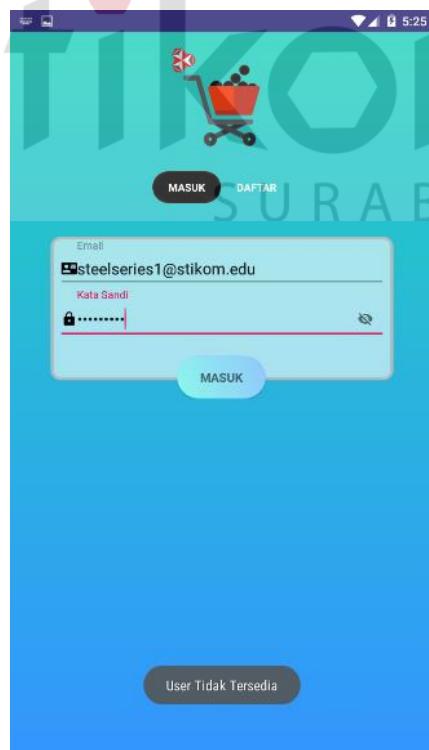
### 4.2.2. Uji coba login dan membuat akun baru

Tabel 4.2 Uji coba login dan membuat akun

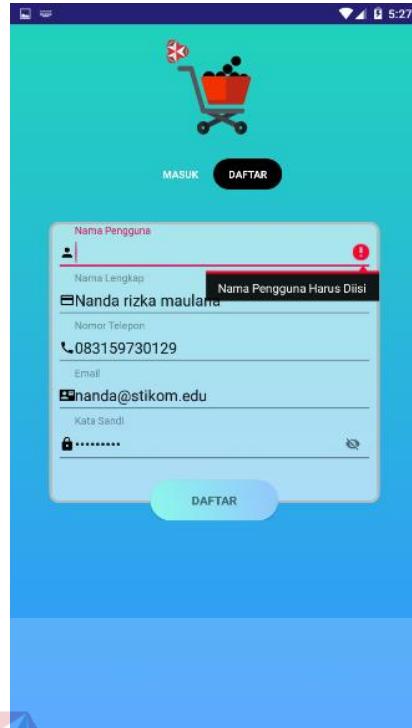
No	Proses	Input	Output yang diharapkan	Hasil
<b>Login</b>				
1	Menguji validasi isian user yang terdaftar	Email dan password benar	Menampilkan halaman utama	Uji berhasil (Gambar 2.30)
2	Menguji validasi isian user yang tidak terdaftar	Email dan password salah	Menampilkan pesan kesalahan	Uji berhasil (Gambar 2.31)
<b>Daftar</b>				
3	Menguji validasi isian	Text box isian sudah terisi semua	Menampilkan pesan apabila terdapat isian yang belum diisi	Uji berhasil (Gambar 2.32)
4	Menguji tombol simpan	Tombol simpan	Data akun baru tersimpan pada database	Uji berhasil (Gambar 2.33)



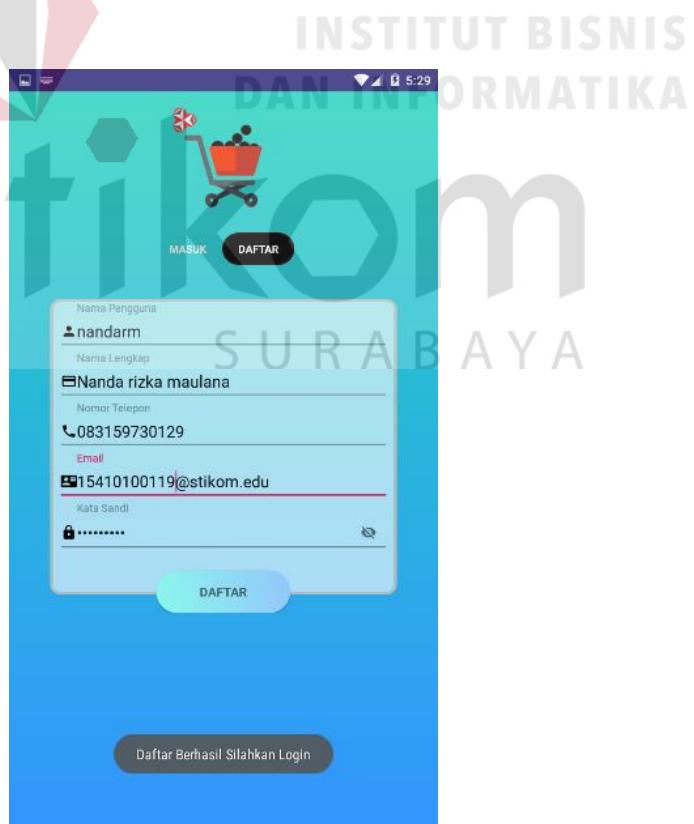
Gambar 4.30 Hasil Uji Coba Login User terdaftar



Gambar 4.31 Hasil Uji Coba Login User Tidak Terdaftar



Gambar 4.32 Hasil Uji Coba Validasi Isian Kosong



Gambar 4.33 Hasil Uji Coba Mendaftarkan Akun Baru

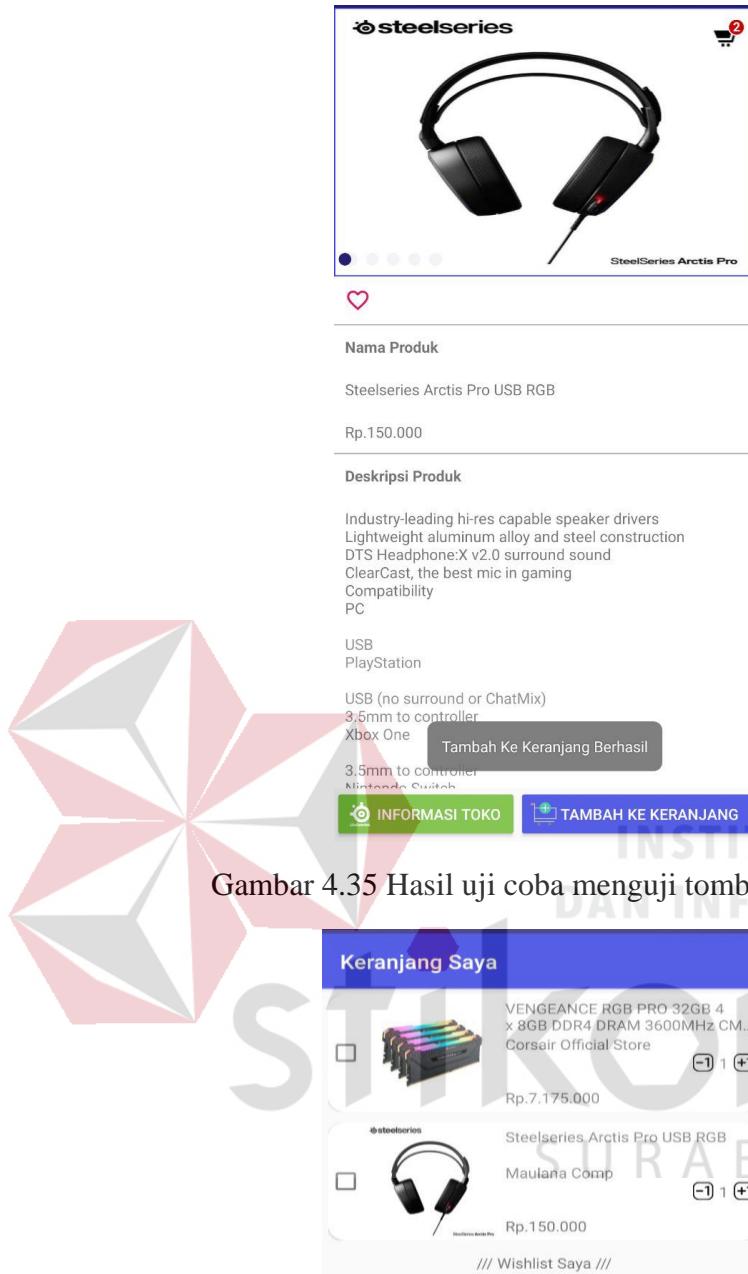
#### 4.2.3. Uji coba pembeli melakukan pemesanan

Tabel 4. 3 Uji coba pembeli melakukan pemesanan

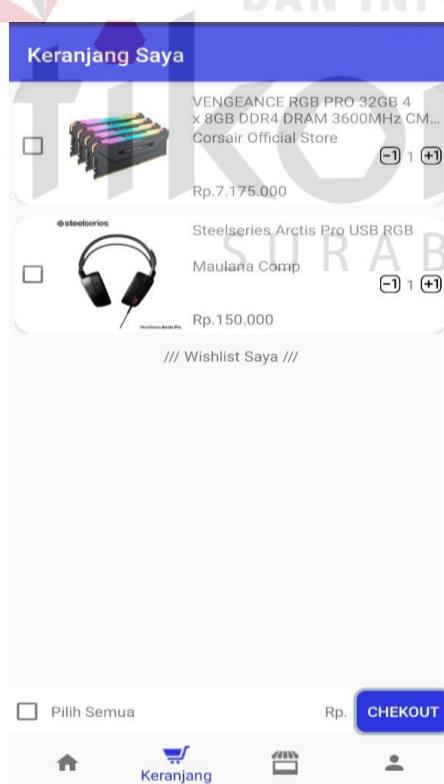
No	Tujuan	Input	Output yang diharapkan	Hasil
1	Menguji tombol detail	Tombol detail	Menampilkan halaman detail produk yang dipesan	Gambar 4.34
2	Menguji tombol tambah	Tombol tambah	Menampilkan halaman form tambah produk	Gambar 4.35
3	Menguji tombol keranjang	Tombol keranjang	Menampilkan list barang yang telah ditambahkan ke keranjang	Gambar 4.36



Gambar 4.34 Hasil uji coba menguji tombol detail



Gambar 4.35 Hasil uji coba menguji tombol tambah

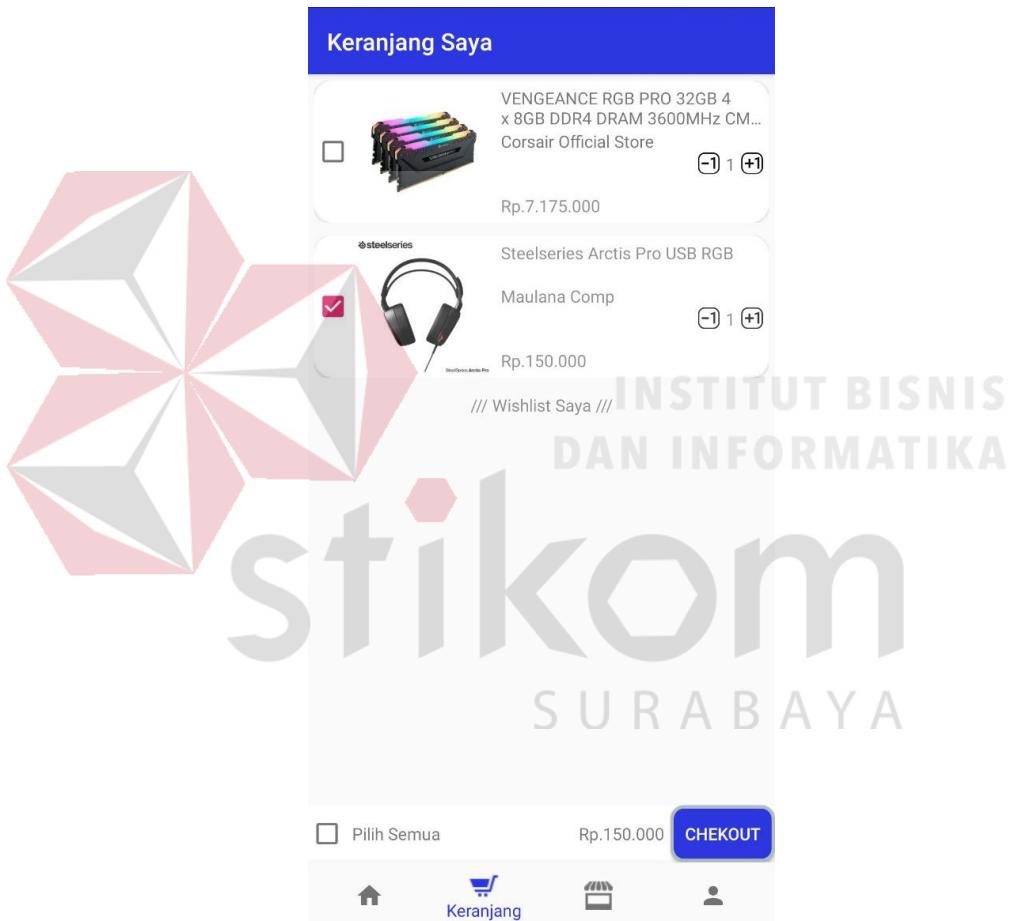


Gambar 4.36 Hasil uji coba tombol keranjang

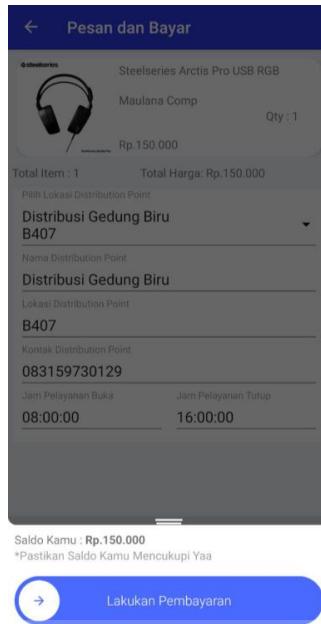
#### 4.2.4. Uji coba pembeli melakukan pembelian

Tabel 4. 4 Uji coba pembeli melakukan pembelian

No	Tujuan	Input	Output yang diharapkan	Hasil
1	Menampilkan data keranjang		Data keranjang tampil sesuai dengan yang telah diinputkan	Gambar 4.37
2	Menguji tombol bayar	Tombol bayar	Data pembayaran tersimpan pada database	Gambar 4.38



Gambar 4.37 Hasil uji coba menampilkan data keranjang

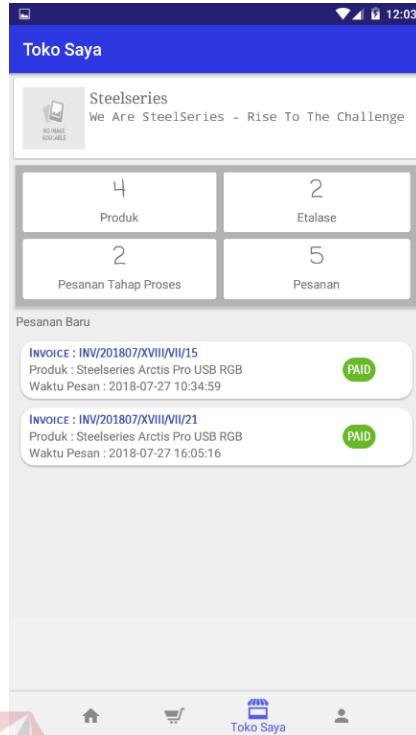


Gambar 4.38 Hasil uji coba menguji tombol bayar

#### 4.2.5. Uji coba penjual melakukan penjualan

Tabel 4. 5 Uji coba penjual melakukan penjualan

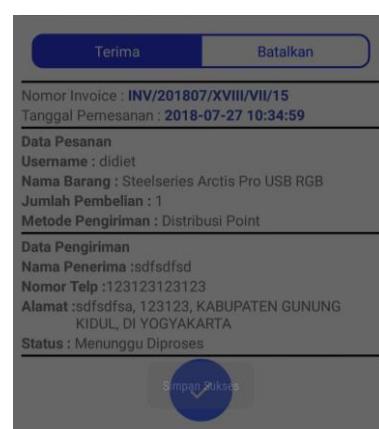
No	Tujuan	Input	Output yang diharapkan	Hasil
1	Menampilkan data penjualan		Data penjualan tampil sesuai dengan database	Gambar 4.39
2	Menguji tombol detail	Tombol detail	Menampilkan halaman detail transaksi yang dilakukan	Gambar 4.40
3	Menguji tombol slide to confirm	Tombol slide to confirm	Data transaksi tersimpan pada database	Gambar 4.41



Gambar 4.39 Hasil uji coba list pesanan



Gambar 4.40 Hasil uji coba detail pesanan



Gambar 4.41 Hasil uji coba tombol slide

#### 4.2.6. Uji coba membuka toko

Tabel 4. 6 Uji coba membuka toko

No	Proses	Input	Output yang diharapkan	Hasil
1	Menguji validasi isian	Text box isian sudah terisi semua	Menampilkan pesan apabila terdapat isian yang belum diisi	Uji Berhasil (Gambar 4.42)
2	Menguji tombol simpan	Tombol simpan	Data buka toko tersimpan pada database	Uji Berhasil (Gambar 4.43)



Gambar 4.42 Hasil Uji Coba Validasi Isian

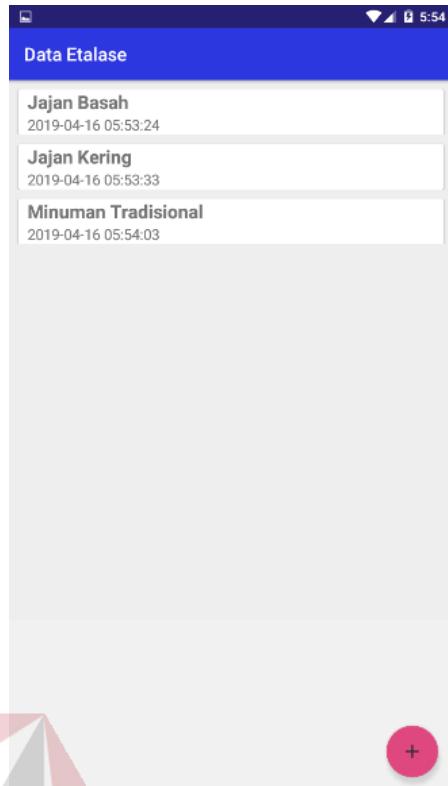


Gambar 4.43 Hasil Uji Tombol Simpan

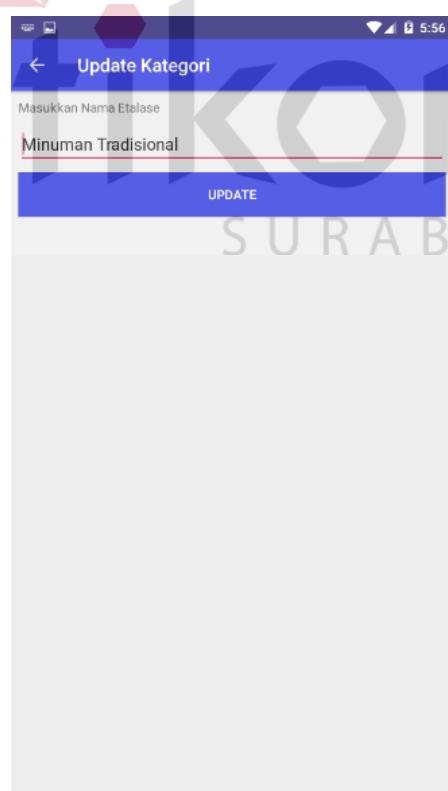
#### 4.2.7. Uji coba penjual mengelola etalase

Tabel 4. 7 Uji coba penjual mengelola etalase

No	Proses	Input	Output yang diharapkan	Hasil
1	Menampilkan data etalase		Data etalase tampil sesuai dengan yang telah disimpan pada database	Uji Berhasil (Gambar 4.44)
2	Menguji tombol edit	Tombol edit	Data etalase teredit dan muncul form data etalase yang akan diedit	Uji Berhasil (Gambar 4.45)
3	Menguji tombol hapus	Tombol hapus	Data produk terhapus	Uji Berhasil (Gambar 4.46)
4	Menguji tombol simpan	Tombol simpan	Data etalase tersimpan pada database dan tampil pada halaman etalase	Uji Berhasil (Gambar 4.47)



Gambar 4.44 Hasil uji coba menampilkan data etalase



Gambar 4.45 Hasil uji coba edit etalase

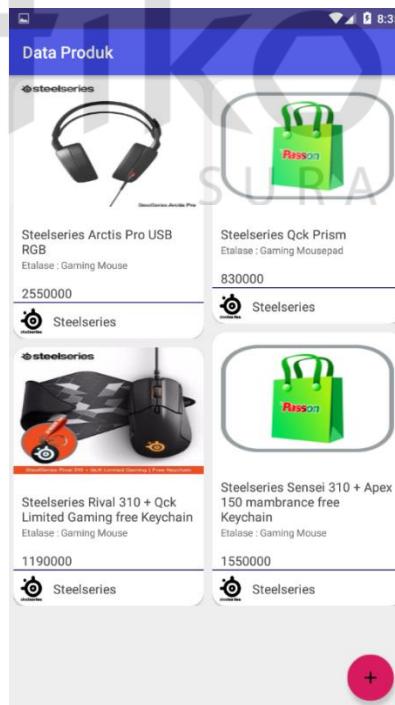


Gambar 4.46 Hasil uji coba hapus etalase

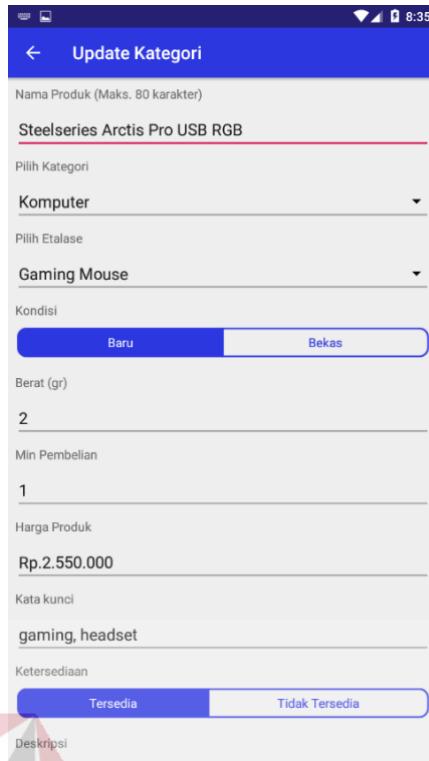
#### 4.2.8. Uji coba penjual mengelola produk

Tabel 4. 8 Uji coba penjual mengelola produk

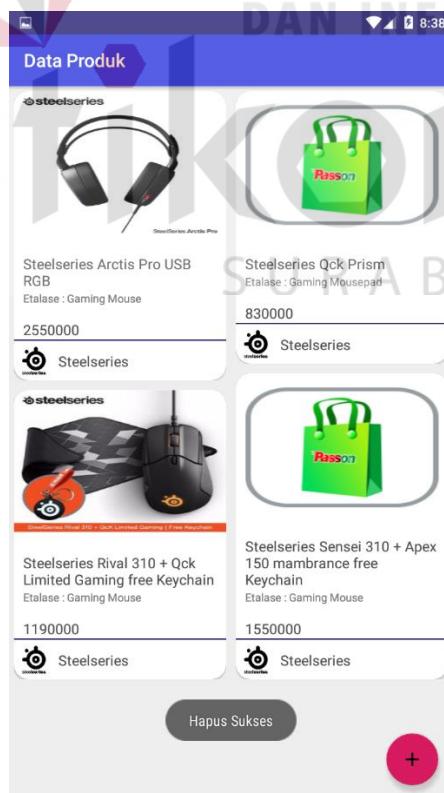
No	Proses	Input	Output yang diharapkan	Hasil
1	Menampilkan data produk		Data produk tampil sesuai dengan yang telah diinputkan	Gambar 4.48
2	Menguji tombol edit	Tombol edit	Muncul form data produk yang akan diedit	Gambar 4.49
3	Menguji tombol hapus	Tombol hapus	Data produk terhapus	Gambar 4.50
4	Menguji tombol tambah	Tombol tambah	Menampilkan halaman form tambah produk	Gambar 4.51
5	Menguji validasi isian	Text box isian sudah terisi semua	Menampilkan pesan apabila terdapat isian yang belum diisi	Gambar 4.52
6	Menguji tombol simpan	Tombol simpan	Data buka toko tersimpan pada database dan tampil pada halaman produk	Gambar 4.53



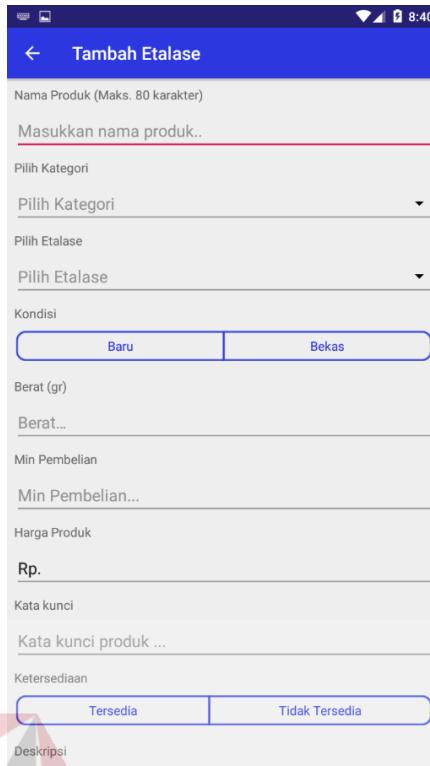
Gambar 4.48 Hasil uji coba menampilkan data produk



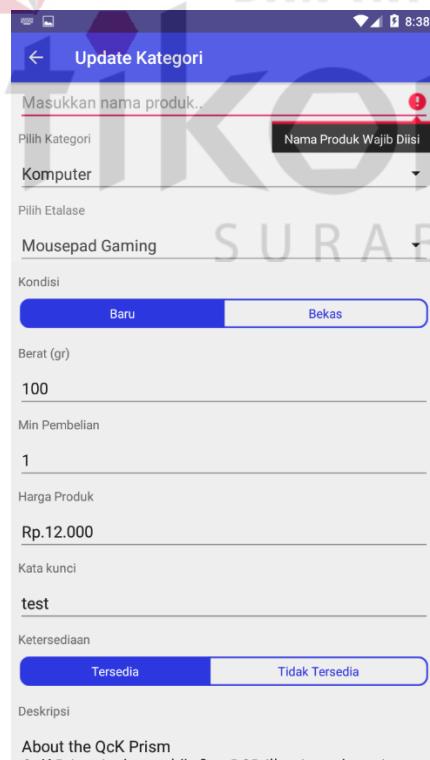
Gambar 4.49 Hasil uji coba edit produk



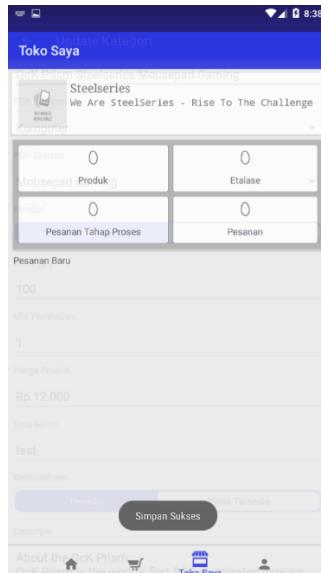
Gambar 4. 50 Hasil uji coba hapus produk



Gambar 4.51 Hasil uji coba tambah produk



Gambar 4.52 Hasil uji coba validasi isian tambah produk



Gambar 4.53 Hasil uji coba simpan produk

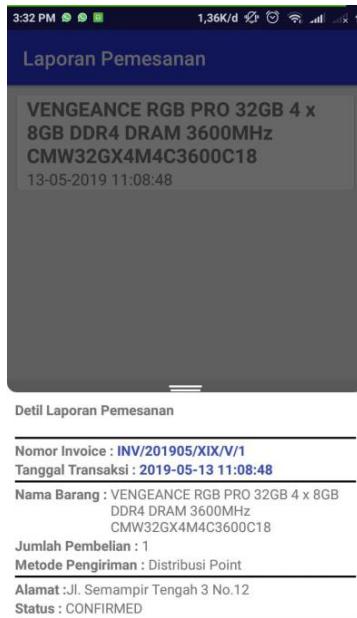
#### 4.2.9. Uji coba laporan riwayat pemesanan

Tabel 4. 9 Uji coba laporan riwayat pemesanan

No	Tujuan	Input	Output yang diharapkan	Hasil
1	Menampilkan data laporan pemesanan		Data penjualan tampil sesuai dengan database	Gambar 4.55
2	Menguji tombol detail	Tombol detail	Menampilkan detail riwayat pemesanan	Gambar 4.56



Gambar 4. 54 Hasil uji coba menampilkan riwayat pemesanan

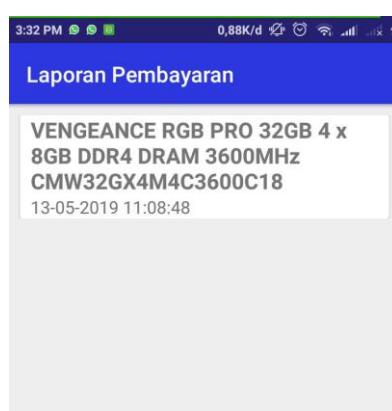


Gambar 4. 55 Hasil uji coba detail riwayat pemesanan

#### 4.2.10. Uji coba laporan riwayat pembayaran

Tabel 4. 10 Uji coba laporan riwayat pembayaran

No	Tujuan	Input	Output yang diharapkan	Hasil
1	Menampilkan data laporan pembayaran		Data penjualan tampil sesuai dengan database	Gambar 4.57
2	Menguji tombol detail	Tombol detail	Menampilkan detail riwayat pembayaran	Gambar 4.58



Gambar 4.56 Hasil uji coba menampilkan riwayat pembayaran



Detail Laporan Pembayaran

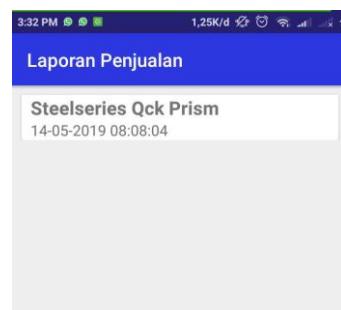
Nomor Invoice : INV/201905/XIX/V/1  
 Tanggal Transaksi : 2019-05-13 11:08:48  
 Nama Barang : VENGEANCE RGB PRO 32GB 4 x 8GB  
 DDR4 DRAM 3600MHz  
 CMW32GX4M4C3600C18  
 Harga : Rp.7.175.000  
 Nama Toko : Corsair Official Store

Gambar 4.57 Hasil uji coba detail riwayat pembayaran

#### 4.2.11. Uji coba laporan riwayat penjualan

Tabel 4. 11 Uji coba laporan riwayat penjualan

No	Tujuan	Input	Output yang diharapkan	Hasil
1	Menampilkan data laporan penjualan		Data penjualan tampil sesuai dengan database	Gambar 4.58
2	Menguji tombol detail	Tombol detail	Menampilkan detail riwayat penjualan	Gambar 4.59



Gambar 4. 58 Hasil uji coba menampilkan riwayat penjualan



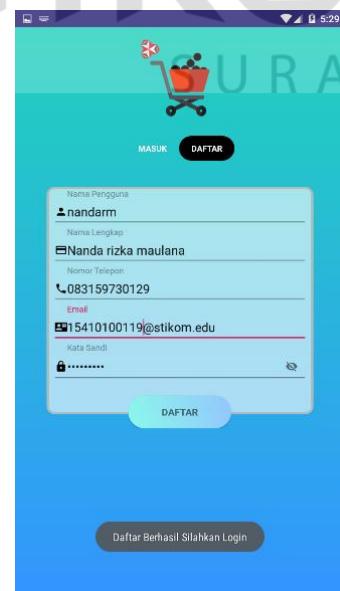
Gambar 4. 59 Hasil uji coba menampilkan riwayat penjualan

### 4.3 Evaluasi Kesesuaian Akhir

Evaluasi kesesuaian akhir sistem dilakukan untuk memastikan aplikasi yang dibangun sudah memenuhi semua kebutuhan yang terdapat pada proses bisnis. Hasil evaluasi yang telah didapatkan yaitu :

#### 4.3.1 Membuat Akun Baru

- 1) Pengguna mengisi form pendaftaran



Gambar 4. 60 Hasil evaluasi mengisi form pendaftaran

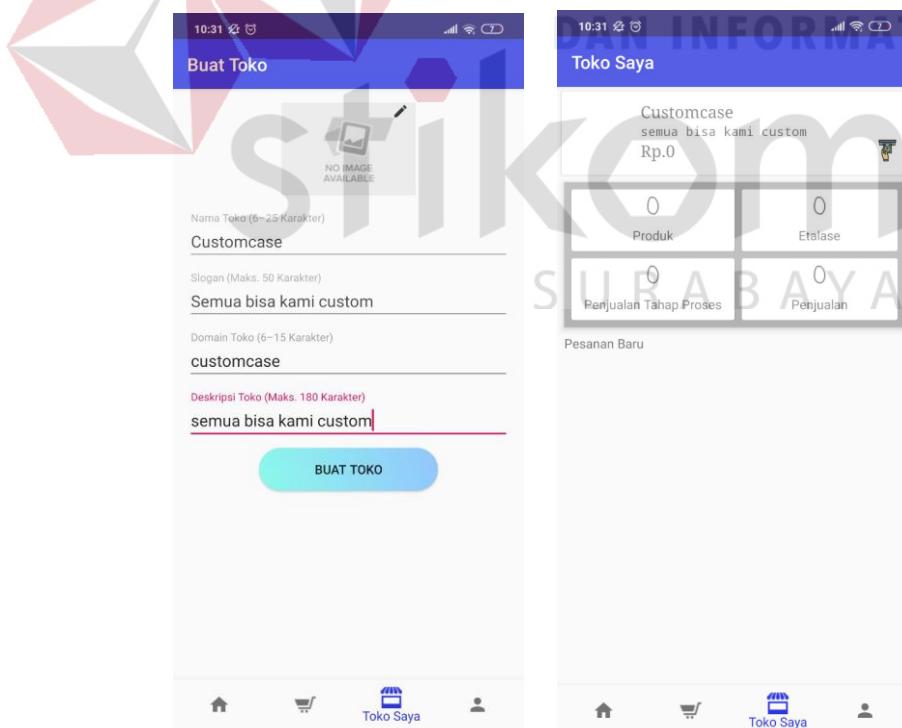
2) Pengguna menggunakan akun terdaftar untuk login



Gambar 4. 61 Hasil evaluasi menggunakan akun terdaftar

### 4.3.2 Membuka Toko

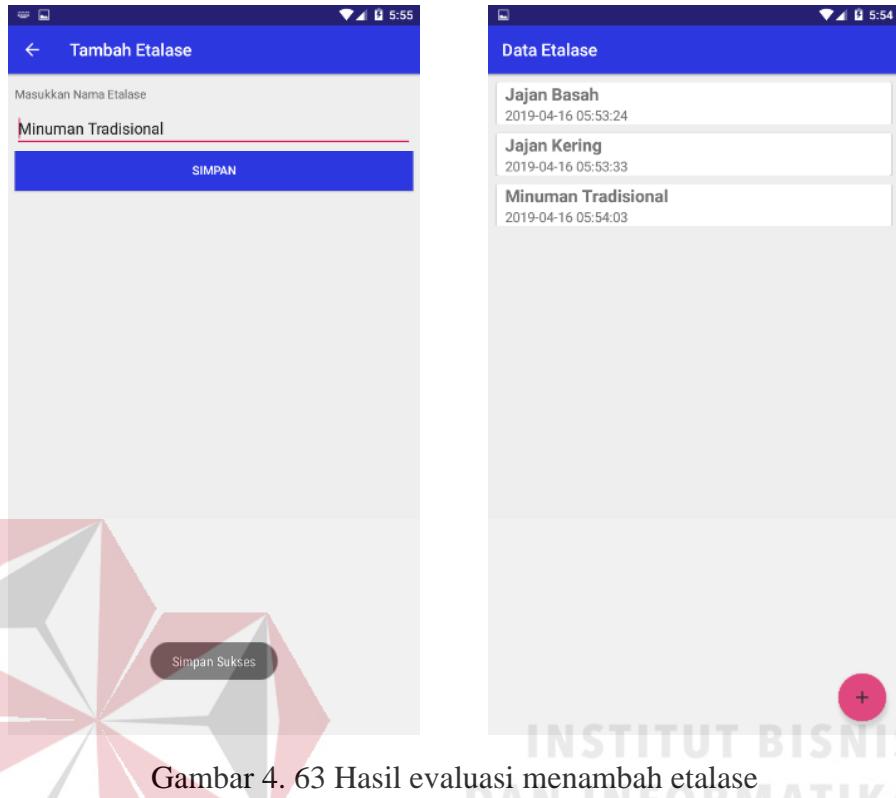
1) Pengguna mengisi form buat toko



Gambar 4. 62 Hasil evaluasi mengisi form buat toko

### 4.3.3 Mengelola Etalase

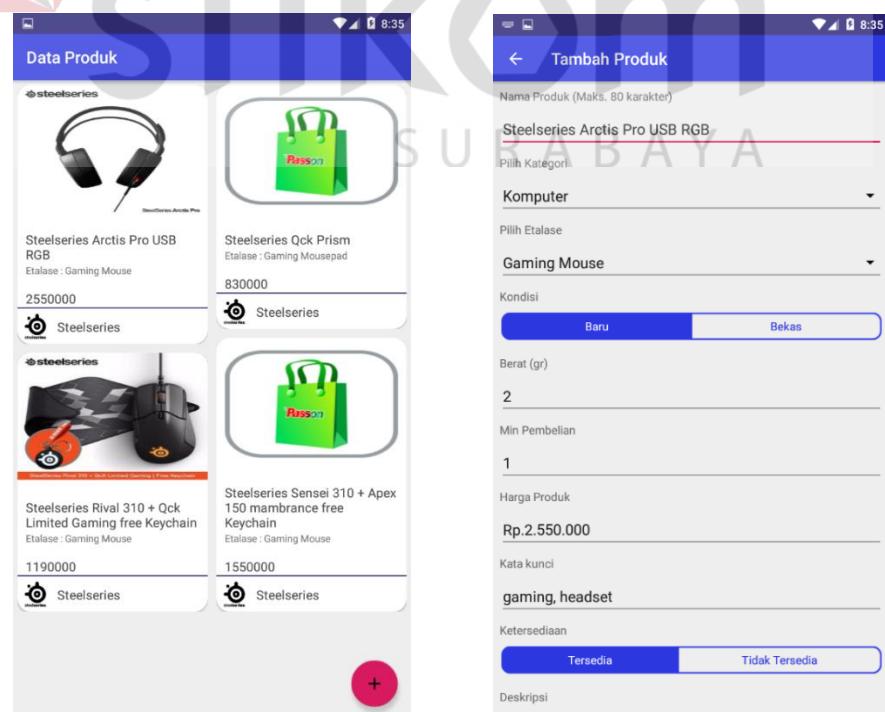
- Pengguna sebagai penjual menambah data etalase



Gambar 4. 63 Hasil evaluasi menambah etalase

### 4.3.4 Mengelola Produk

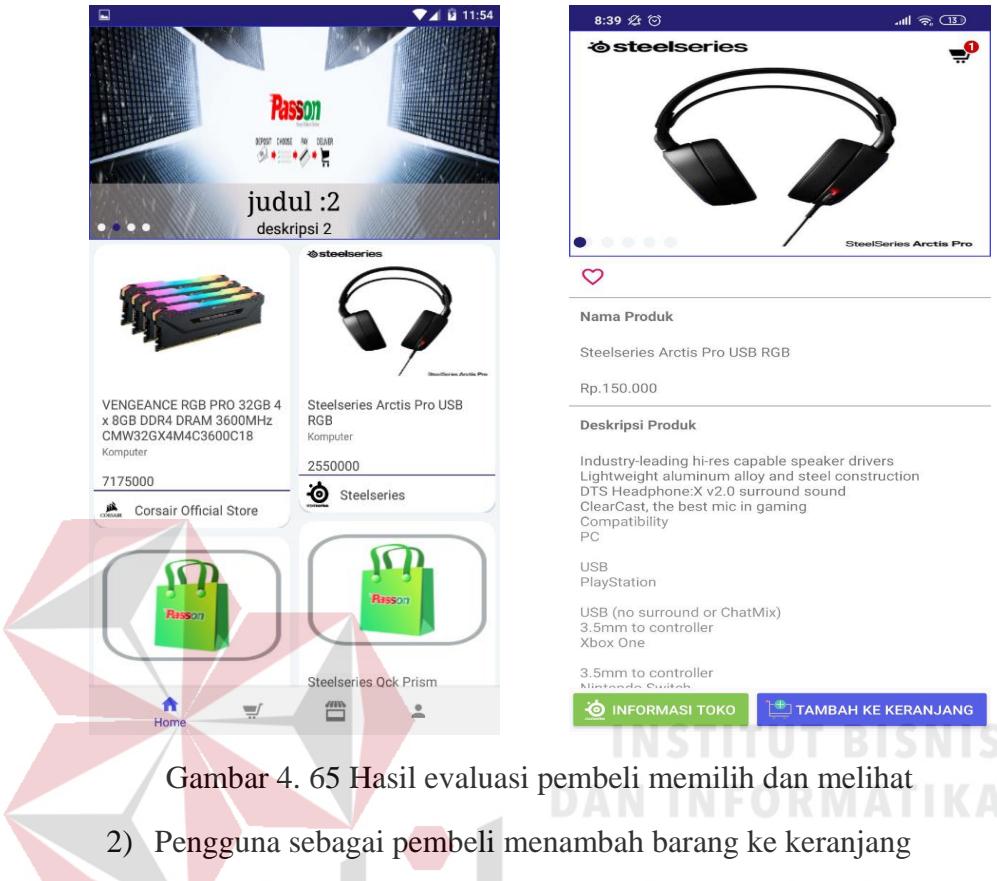
- Pengguna sebagai penjual menambah data produk



Gambar 4. 64 Hasil evaluasi menambah data produk

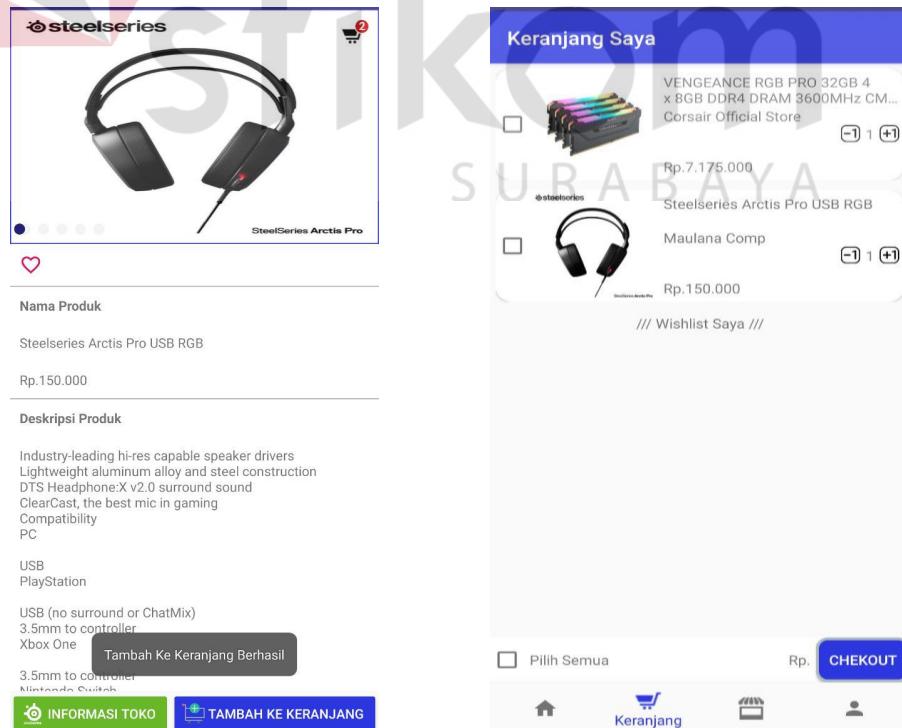
#### 4.3.5 Melakukan Pemesanan

1) Pengguna sebagai pembeli memilih dan melihat produk



2) Pengguna sebagai pembeli menambah barang ke keranjang

2) Pengguna sebagai pembeli menambah barang ke keranjang



Gambar 4. 66 Hasil evaluasi menambah barang ke keranjang

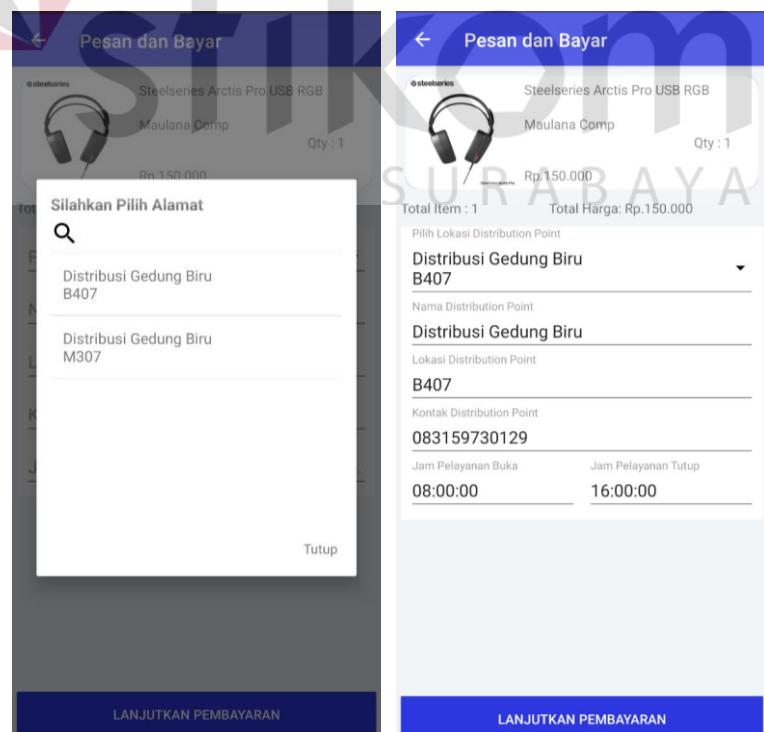
#### 4.3.6 Melakukan Pembelian

- Pengguna sebagai pembeli melakukan checkout daftar keranjang



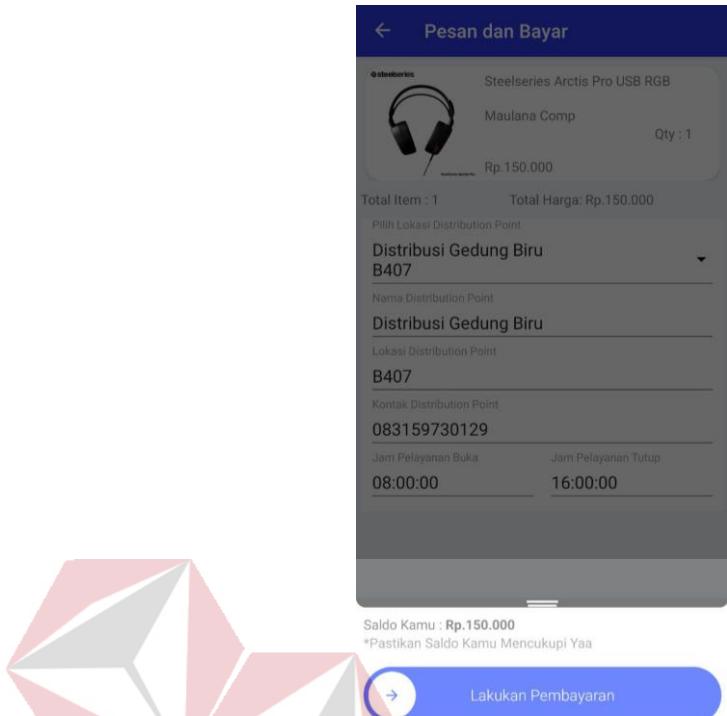
Gambar 4. 67 Hasil evaluasi melakukan checkout daftar keranjang

- Pengguna sebagai pembeli mengisi data pembelian



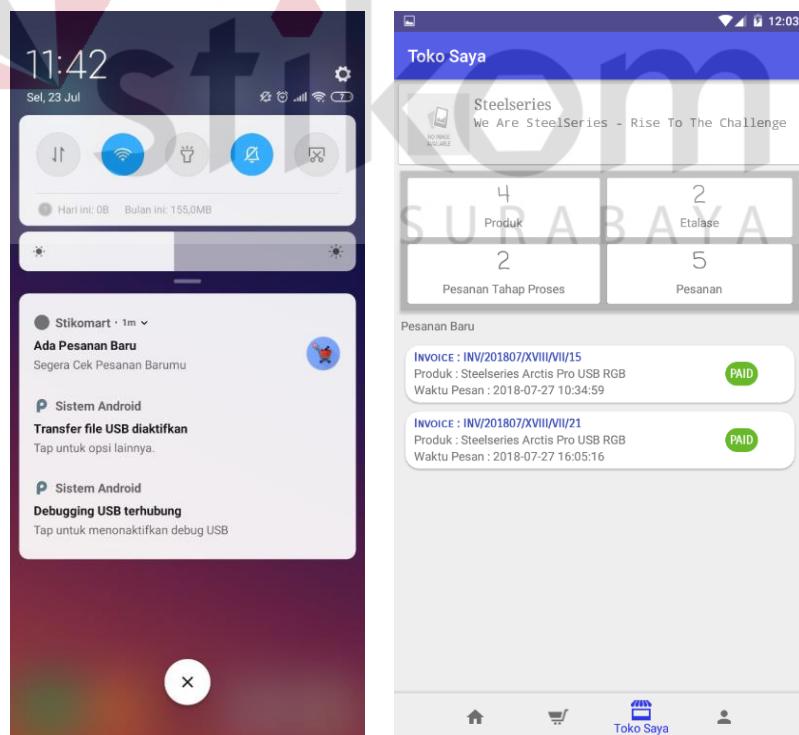
Gambar 4. 68 Hasil evaluasi mengisi data pembelian

3) Pengguna sebagai pembeli melakukan pembayaran



Gambar 4. 69 Hasil evaluasi melakukan pembayaran

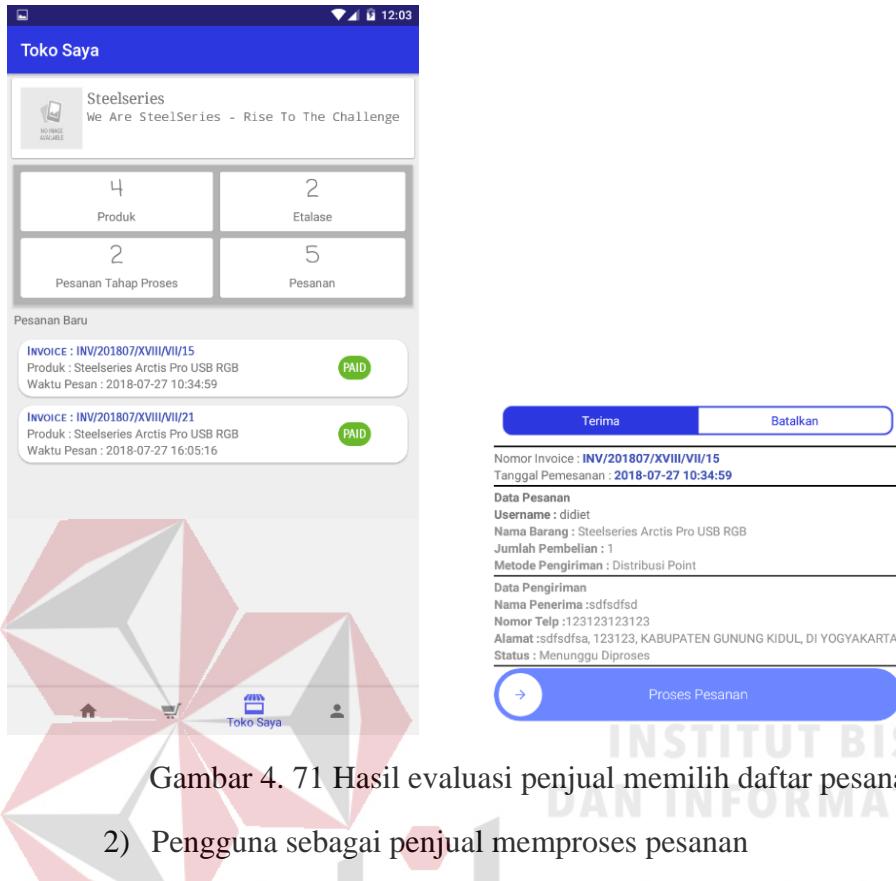
4) Pengguna sebagai penjual menerima pesanan



Gambar 4. 70 Hasil evaluasi penjual menerima pesanan

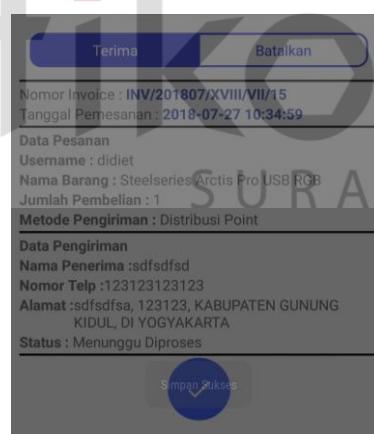
#### 4.3.7 Melakukan Penjualan

- Pengguna sebagai penjual memilih daftar pesanan



Gambar 4. 71 Hasil evaluasi penjual memilih daftar pesanan

- Pengguna sebagai penjual memproses pesanan



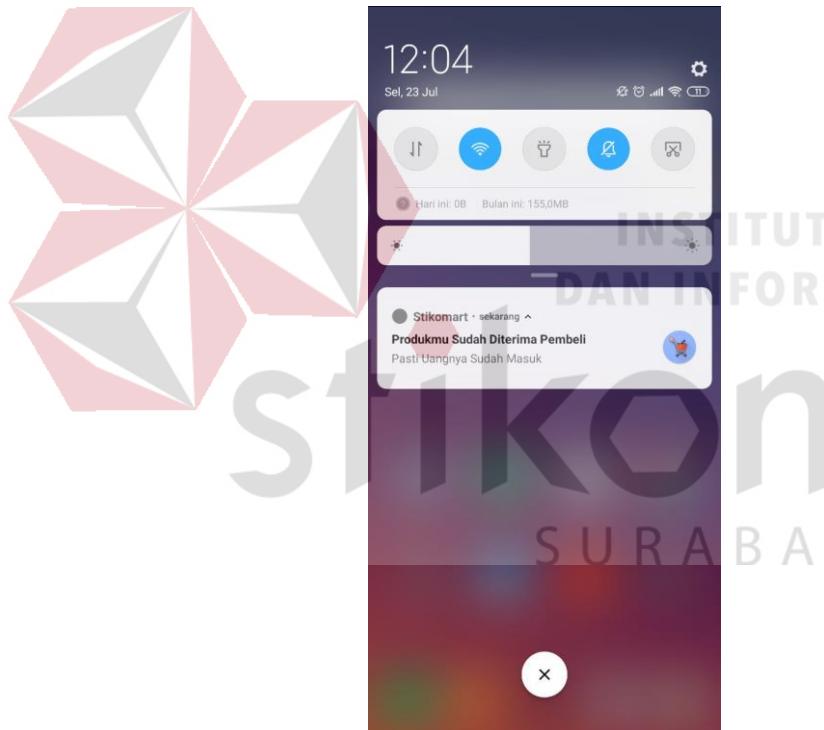
Gambar 4. 72 Hasil evaluasi penjual memproses pesanan

3) Pengguna sebagai penjual menyerahkan produk



Gambar 4. 73 Hasil evaluasi penjual menyerahkan produk

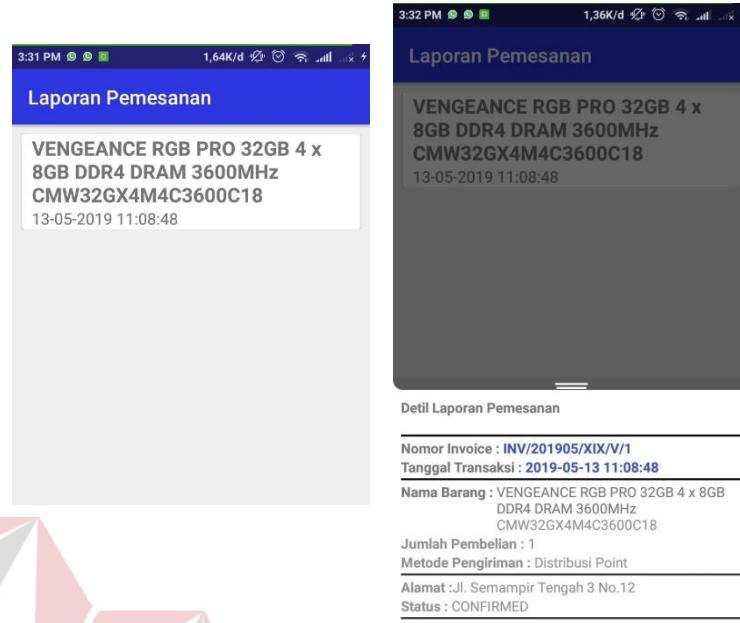
4) Pengguna sebagai penjual menerima dana ketika pembeli konfirmasi barang diterima pada distribution point



Gambar 4. 74 Hasil evaluasi penjual menerima dana ketika pembeli konfirmasi barang diterima pada distribution point

#### 4.3.8 Melihat Riwayat Pesanan

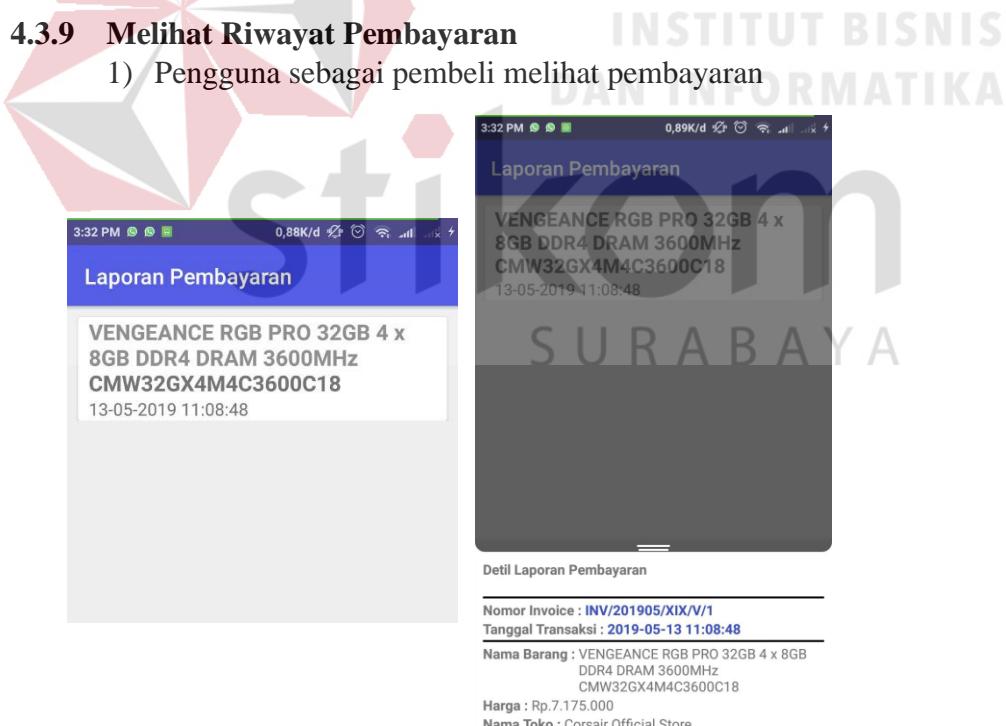
- Pengguna sebagai pembeli melihat pesanan



Gambar 4. 75 Hasil evaluasi pembeli melihat pesanan

#### 4.3.9 Melihat Riwayat Pembayaran

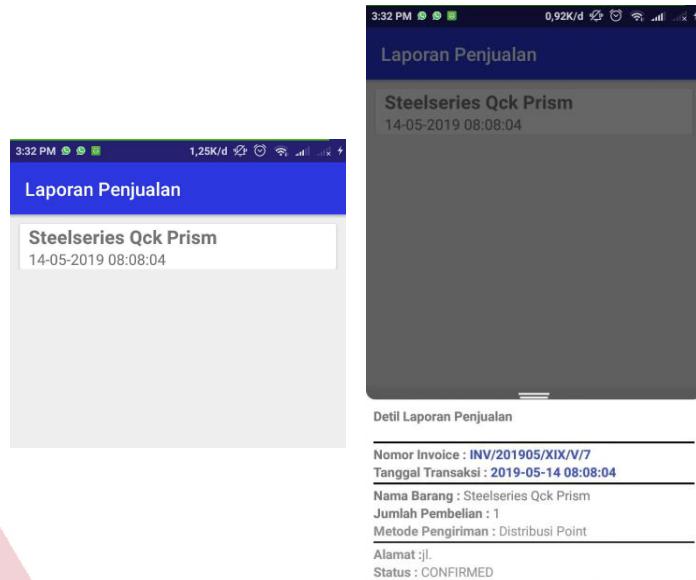
- Pengguna sebagai pembeli melihat pembayaran



Gambar 4. 76 Hasil evaluasi pembeli melihat pembayaran

#### 4.3.10 Melihat Riwayat Penjualan

1) Pengguna sebagai pembeli melihat daftar penjualan



Gambar 4. 77 Hasil evaluasi pembeli melihat daftar penjualan

## **BAB V**

### **PENUTUP**

#### **5.1 Kesimpulan**

Kesimpulan yang dapat diambil berdasarkan hasil implementasi dan evaluasi dari pembuatan aplikasi *Marketplace* Stikom Surabaya yang telah dilakukan bahwa :

1. Aplikasi dapat digunakan sebagai wadah bagi penjual untuk memasarkan produk.
2. Aplikasi dapat digunakan untuk melakukan transaksi penjualan maupun pembelian.
3. Aplikasi dapat membantu penjual dan pembeli untuk memonitor dan memproses transaksi yang berlangsung.

#### **5.2 Saran**

Adapun beberapa saran yang dapat digunakan untuk peningkatan aplikasi , diantaranya sebagai berikut :

1. Metode pembayaran dapat ditambahkan seperti pembayaran melalui alfamart, atm , dan lainnya.
2. Dengan penerapan metode *gamifikasi* diharapkan dapat meningkatkan minat pengguna dengan memberikan bonus kenaikan *level* seperti *voucher* potongan harga.
3. Penerapan algoritma khusus untuk pencarian produk agar dapat memaksimalkan hasil pencarian pengguna.
4. Penggunaan *scheduler task* untuk melakukan perubahan status secara otomatis.

## DAFTAR PUSTAKA

Dewanto, I. J. (2004). System Development Life Cycle Dengan Beberapa Pendekatan. *Fasilkom*, 2(1), 39-47.

Gata, W., & Gata, G. (2013). *Sukses Membangun Aplikasi Penjualan dengan Java*. Jakarta: Elex Media Komputindo.

Haryanti, S., & Irianto, T. (2011). Rancang Bangun Sistem Informasi E-Commerce Untuk Usaha Fashion Studi Kasus Omah Mode Kudus. *Journal Speed (Sentra Penelitian Engineering dan Edukasi)*, 3(1), 8-14.

Juansyah, A. (2015). Pembangunan Aplikasi Child Tracker Berbasis Assisted - Global Positioning System (A-GPS) Dengan Platform Android. *Jurnal Ilmiah Komputer dan Informatika (KOMPUTA)*, 1(1), 1-8.

Kasman, A. D. (2013). *Kolaborasi Dahsyat Android Dengan PHP dan MYSQL*. Yogyakarta: Lokomedia.

Kotler, P. (1997). *Manajemen Pemasaran Analisis Perencanaan, Implementasi dan Pengendalian* (terjemahan Jaka Wasana). Jakarta: Salemba Empat.

Luthfi, F. (2017). Penggunaan Framework Laravel Dalam Rancang Bangun Modul Back-End Artikel Website Bisnisbisnis.ID. *JISKa*, 2(1), 34-41.

Mansur. (2015). Business To Business (B2B) E-Marketplace Sebagai Media Promosi Produk Usaha Kecil Dan Menengah (UKM). *BBM (Buletin Bisnis & Manajemen)*, 01(01), 1-13.

Muzakir, A. (2016). Framework Phonegap Sebagai Teknologi Cross-Platform Mobile Development: Studi Kasus Kamus Tumbuhan. *KNTIA*, 4(1), E7-E11.

Pressman, R. (2015). *Rekayasa Perangkat Lunak: Pendekatan Praktisi Buku 1*. Yogyakarta: Andi.

Raharjo, B. (2011). *Belajar Otodida Membuat Database Menggunakan MySQL*. Bandung: Informatika Bandung.

Reddy, M. (2011). *API Design For C++*. Burlington: Elsevier.

Ridhoni, W. (2018). Rancang Bangun Website Responsif untuk Marketplace Online Berbasis Koperasi. *PHASTI Jurnal Teknik Informatika Politeknik Hasnur*, 4(1), 25-35.

Romeo, S. (2003). *Testing Dan Implementasi Sistem, Edisi Pertama*. Surabaya: STIKOM.

Sadgotra, W. Y., & Saputra, E. H. (2013). Perancangan Online Marketplace Untuk Usaha Kecil Dan Menengah (Ukm) Di Kabupaten Purworejo. *Jurnal Ilmiah DASI*, 14(04), 54-58. Retrieved from teknonisme.

Safaat, N. (2012). *Pemograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android*. Bandung: Informatika Bandung.

Sutami, W. D. (2005). Strategi Rasional Pedagang Pasar Tradisional. *BioKultur*, 1(2), 127-148.

Tanaem, P. F., Manongga, D., & Iriani, A. (2016). RESTFul Web Service Untuk Sistem Pencatatan Transaksi Studi Kasus PT. XYZ. *Jurnal Teknik Informatika dan Sistem Informasi*, 2(1), 1-10.

