

**RANCANG BANGUN APLIKASI MANAJEMEN PENJUALAN BARANG  
DENGAN SISTEM KONSINYASI BERBASIS ANDROID**

**(Skripsi)**

**Oleh**

**AHMAD TAQIYUDIN**



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM**

**UNIVERSITAS LAMPUNG**

**BANDAR LAMPUNG**

**2021**

## **ABSTRAK**

### **RANCANG BANGUN APLIKASI MANAJEMEN PENJUALAN BARANG DENGAN SISTEM KONSINYASI BERBASIS ANDROID**

**Oleh**

**AHMAD TAQIYUDIN**

UMKM memiliki peranan penting dalam mendorong perekonomian negara. UMKM menyumbang angka hingga 99.9% dalam hal penyerapan tenaga kerja di Indonesia. Tujuan dari penulisan ini adalah dapat membantu UMKM mengontrol manajerial internal agar menjadi lebih baik lagi dengan menggunakan smartphone Android. Metode Penelitian yang digunakan adalah metode studi pustaka, analisis dan perancangan, kemudian dilakukan tahapan implementasi pada sistem yang telah dirancang. Hasil yang ingin dicapai adalah terciptanya sebuah aplikasi yang dapat membantu manajerial UMKM. Kesimpulan yang didapat adalah peralatan ini telah diuji dan dapat digunakan menggunakan smartphone Android.

**Kata Kunci:** Konsinyasi, Manajerial, Aplikasi Android

## **ABSTRACT**

### **THE IMPLEMENTATION OF ANDROID-BASED APPLICATION FOR GOODS SALES MANAGEMENT WITH CONSIGNMENT SYSTEM**

**By**

**AHMAD TAQIYUDIN**

*UMKM have an important role in driving the country's economy. UMKM contribute up to 99.9% in employment absorption in Indonesia. The purpose of this paper is to help UMKM control their internal managerial to be even better by using an Android smartphone. The research method that used in this research is literature review, analysis and design. And then the implementation stage is carried out on the system that has been designed. The result to be achieved is the creation of an application that can help the UMKM managerial. The conclusion obtained is this equipment has been tested and can be used using an Android smartphone.*

**Key words:** Consignment, Managerial, Android Application

**RANCANG BANGUN APLIKASI MANAJEMEN PENJUALAN BARANG  
DENGAN SISTEM KONSINYASI BERBASIS ANDROID**

**Oleh**

**AHMAD TAQIYUDIN**

**Skripsi**

**Sebagai Salah Satu Syarat untuk Memperoleh Gelar**

**SARJANA KOMPUTER**

**Pada**

**Jurusan Ilmu Komputer**

**Fakultas Matematika dan Ilmu Pengetahuan Alam**



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM**

**UNIVERSITAS LAMPUNG**

**BANDAR LAMPUNG**

**2021**

Judul Skripsi : **RANCANG BANGUN APLIKASI MANAJEMEN  
PENJUALAN BARANG DENGAN SISTEM  
KONSINYASI BERBASIS ANDROID**

Nama Mahasiswa : **Ahmad Taqiyudin**

No. Pokok Mahasiswa : 1517051115

Jurusan : Ilmu Komputer

Fakultas : Matematika dan Ilmu Pengetahuan Alam



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

**Dewi Asiah Shofiana, S.Kom, M.Kom**  
NIP. 19950929 202012 2 030

2. Mengetahui  
Ketua Jurusan Ilmu Komputer  
FMIPA Universitas Lampung

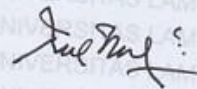
**Didik Kurniawan. S.Si., MT**  
NIP. 19800419 200501 1 004



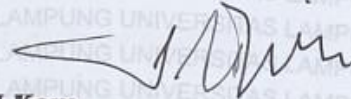
**MENGESAHKAN**

**1. Tim Penguji**

**Ketua : Dr. Ir. Kurnia Muludi, M.S.Sc.** .....



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



**Penguji  
Bukan Pembimbing : Dwi Sakethi, M.Kom.** .....



**2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam**



**Dr. Eng. Supto Dwi Yuwono, S.Si., M.T.**  
NIP. 19740705 200003 1 001

**Tanggal Lulus Ujian Skripsi: 30 Juli 2021**

## PERNYATAAN

Saya yang bertanda tangan di bawah ini, menyatakan bahwa skripsi saya yang berjudul **“Rancang Bangun Aplikasi Manajemen Penjualan Barang Dengan Sistem Konsinyasi Berbasis Android”** ini merupakan hasil karya sendiri dan bukan hasil karya orang lain. Semua tulisan yang tertuang dalam skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung. Apabila di kemudian hari terbukti bahwa skripsi ini merupakan hasil salinan atau dibuat oleh orang lain, maka saya bersedia menerima sanksi berupa pencabutan gelar akademik yang telah saya terima.

Bandar Lampung, 11 Agustus 2021



Ahmad Taqiyudin

1517051115

## RIWAYAT HIDUP



Penulis dilahirkan pada tanggal 21 Februari 1997 di Podosari, Kecamatan Pringsewu, Kabupaten Pringsewu. Penulis merupakan anak kedua dari tiga bersaudara, dengan Ibu bernama Lestari Prihatmini dan Ayah bernama M. Nurhidayat. Penulis menempuh Pendidikan formal pertama kali di TK Aisyiah Pringsewu tahun 2001 sampai 2003. Melanjutkan Pendidikan Dasar di SD N 1 Pringsewu Selatan pada tahun 2003 hingga 2009. Kemudian masuk Pendidikan Sekolah Menengah Pertama di SMP Muhammadiyah 1 Pringsewu pada tahun 2009 hingga 2012. Selanjutnya penulis melanjutkan Sekolah Menengah Atas di SMA Negeri 1 Pringsewu pada tahun 2012 hingga 2015. Pada tahun 2015 penulis terdaftar sebagai mahasiswa S1 Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam (FMIPA) Universitas Lampung melalui jalur Seleksi Bersama Masuk Perguruan Tinggi Negeri (SBMPTN). Selama menjalani peran mahasiswa, penulis aktif dalam beberapa kegiatan diantaranya:

1. Himpunan Mahasiswa Jurusan Ilmu Komputer (Himakom) Fakultas MIPA Universitas Lampung. Penulis menjabat sebagai Anggota Muda pada tahun 2015, dan menjadi Anggota Badan Khusus pada tahun 2016.
2. Rohani Islam (ROIS) Fakultas MIPA Universitas Lampung. Penulis menjabat sebagai Anggota Muda Rois pada tahun 2015, Anggota Bidang Kaderisasi pada tahun 2016, Kepala Bidang Media dan Informasi pada tahun 2017, dan menjadi Anggota Istimewa pada tahun 2018.
3. Unit Kegiatan Mahasiswa Bina Rohani Mahasiswa (BIROHMAH) Universitas Lampung. Penulis menjabat sebagai Anggota Keluarga Muda



Birohmah 2015, Pengurus Media Center Birohmah pada tahun 2016 dan 2017, dan menjadi Kepala Departemen Kemediaan pada tahun 2018.

4. Mengikuti Karya Wisata Ilmiah Fakultas MIPA sebagai peserta di Desa Batutegi Tanggamus pada tahun 2016.
5. Asisten Dosen dan Praktikum beberapa mata kuliah di Jurusan Ilmu Komputer pada tahun 2016-2017.
6. Penulis melakukan Kuliah Kerja Nyata Universitas Lampung di Desa Surya Mataram Kecamatan Marga Tiga Kabupaten Lampung Timur pada tahun 2018.
7. Penulis melakukan kerja praktik di PT. Andaglos Global Teknologi selama 40 Hari pada tahun 2019.
8. Guru Produktif di Sekolah Qur'an Daarul Fattah Bandar Lampung pada tahun 2021.

## **MOTTO**

*“Berbahagialah orang yang dapat menjadi tuan bagi dirinya, menjadi pemandu untuk nafsunya, dan menjadi kapten untuk bahtera hidupnya.”*

**(Ali bin Abi Thalib)**

*“Allah tidak membebani seseorang melainkan sesuai dengan kesanggupannya.”*

**(QS. Al Baqarah: 286)**

*“Maka apabila kamu telah selesai (dari suatu urusan), kerjakanlah dengan sungguh-sungguh (urusan) yang lain.”*

**(QS Al Insyirah: 7)**

*“Improvise, Adapt, Overcome”*

## **PERSEMBAHAN**

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

Kupersembahkan karya ini kepada:

Kedua orang tua tercinta, ibu dan bapak yang telah membesarkan, mendidik, memotivasi, mendukung, memberikan semangat dan selalu mendoakan untuk kesuksesanku. Terimakasih untuk semua jeri payah dan pengorbanan yang telah diberikan kepadaku. Mba Isna, Yasmin dan Semua keluarga besar tercinta.

Semua teman teman yang telah kebersamaiku selama ini, terimakasih banyak atas bantuan, dukungan, semangat, perjuangan, dan kenangan yang telah kita lalui bersama.

Dosen dan Guru yang telah mendidik, mengajari dan membimbingku selama ini.

Keluarga Dibawah Langit yang Sama

Keluarga Ilmu Komputer 2015

Almamater Tercinta, Universitas Lampung

## SANWACANA

*Alhamdulillah* rabbi'l'alamiin, puji syukur kehadiran Allah *Subhanahu wa ta'ala* yang telah melimpahkan nikmat dan hidayah-Nya sehingga penulis dapat menyelesaikan skripsi di Jurusan Ilmu Komputer Fakultas MIPA Universitas Lampung dengan judul “RANCANG BANGUN APLIKASI MANAJEMEN PENJUALAN BARANG DENGAN SISTEM KONSINYASI BERBASIS ANDROID”.

Shalawat serta salam semoga senantiasa tercurahkan kepada baginda Rasulullah, Nabi Muhammad *Shallallahu 'alaihi wa sallam* serta keluarga dan para sahabatnya yang selalu kita nantikan syafa'atnya di *yaumul qiyamah* kelak, *Aamiin*.

Penulis mengucapkan terimakasih kepada semua pihak yang telah membantu dan memiliki peran besar dalam penyusunan skripsi ini, antara lain:

1. Kedua Orang Tuaku, Ayahanda M. Nurhidayat dan Ibunda Lestari Prihatmini, Mbak Isna, Adik Yasmin atas curahan kasih sayang motivasi pengorbanan serta iringan doa yang tiada henti.
2. Bapak Dr. Ir. Kurnia Muludi, M.S.Sc., selaku Pembimbing Utama yang senantiasa mencurahkan waktu, tenaga, ilmu pengetahuan, motivasi, nasihat, dan kritikan sehingga terselesaikannya skripsi ini.

3. Ibu Dewi Asiah Shofiana, S.Kom, M.Kom., selaku Pembimbing Kedua yang senantiasa mencurahkan waktu, tenaga, ilmu pengetahuan, motivasi, nasihat, dan kritikan sehingga terselesaikannya skripsi ini.
4. Bapak Drs. Rd. Irwan Adi Pribadi, M.Kom., sebagai pembahas yang telah memberikan saran kepada penulis dalam perbaikan penulisan skripsi.
5. Bapak Dr. Eng. Suropto Dwi Yuwono, M.T., sebagai dekan FMIPA Universitas Lampung.
6. Bapak Didik Kurniawan, S.Si, M.T., selaku Ketua Jurusan Ilmu Komputer Universitas Lampung.
7. Ibu Astria Hijriani, S.Kom, M.Kom., sebagai Sekretaris Jurusan Ilmu Komputer Universitas Lampung.
8. Bapak Dr. Ir. Kurnia Muludi, M.S.Sc., sebagai Pembimbing Akademik selama penulis menjadi mahasiswa Ilmu Komputer Universitas Lampung.
9. Bapak dan Ibu Dosen Jurusan Ilmu Komputer yang telah memberikan ilmu selama penulis menjadi mahasiswa.
10. Ibu Ade Nora Maela, Mas Zainuddin, Mas Ardi Novalia dan Mas Sam yang telah membantu segala urusan administrasi di Jurusan Ilmu Komputer.
11. Keluarga Dibawah Langit yang Sama sebagai teman berbagi getirnya kehidupan dari bangun sampai bangun lagi.
12. Orang orang baik Muhammad Razif Rizqullah, Muhammad Afif Hidayatullah, Muhammad Ikhwan Al Rasyid, Wahyu Kurniawan, Kevin Joan, Zulaikah yang selalu kebersamai dan menjadi pusat semangat dalam penyelesaian skripsi ini.

13. Sobat ngopi Rama Aji Wijaya dan Yudhi Agung yang selalu menemani keluhan keluhan hidup ini dengan secangkir kopi.
14. Teman yang membantu saya ketika mengalami kesulitan Uki, Bisri, Naufal, Dendi, dan Panji.
15. Keluarga Himakom dan Rois FMIPA yang telah memberikan pengalaman organisasi yang sangat berharga.
16. Keluarga Birohmah Unila 2018 yang telah menemani perjalanan hidup penuh liku dan memberikan pengalaman serta pelajaran yang berharga.
17. Keluarga Ilmu Komputer 2015 yang tidak dapat disebutkan satu persatu yang telah memberikan kenangan selama perkuliahan.
18. Almamater Tercinta Universitas Lampung.
19. Dan semua pihak yang telah membantu terlaksana dan terselesaikannya skripsi ini.

Penulis menyadari bahwa skripsi ini memiliki kekurangan, sehingga memerlukan saran yang membangun agar menjadi lebih baik. Penulis berharap skripsi ini dapat bermanfaat sebagai sumber informasi dan literatur bagi penulisan dan penelitian karya ilmiah selanjutnya.

Bandar Lampung, 12 Agustus 2021

Penulis

Ahmad Taqiyudin



## DAFTAR ISI

Halaman

<b>DAFTAR TABEL .....</b>	<b>xv</b>
<b>DAFTAR GAMBAR.....</b>	<b>xvi</b>
<b>DAFTAR KODE PROGRAM .....</b>	<b>xix</b>
<b>I. PENDAHULUAN.....</b>	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	3
1.3 Batasan Masalah.....	4
1.4 Tujuan.....	4
1.5 Manfaat.....	4
<b>II. TINJAUAN PUSTAKA .....</b>	<b>5</b>
2.1 Penjualan .....	5
2.2 Konsinyasi .....	5
2.3 Aplikasi Android .....	7
2.4 Android Studio .....	7

2.5	<i>Unified Modeling Language (UML)</i> .....	8
2.5.1	Penggunaan <i>Unified Modeling Language (UML)</i> .....	8
2.5.2	Pemodelan <i>Unified Modeling Language (UML)</i> .....	10
2.6	<i>Extreme Programming (XP)</i> .....	14
2.7	<i>Database</i> .....	15
2.8	<i>Shared Preference</i> .....	16
2.9	REST API.....	16
2.10	JSON ( <i>Javascript Object Notation</i> ).....	16
2.11	<i>Location Based Service</i> .....	17
2.12	<i>Black Box Testing</i> .....	17
<b>III.</b>	<b>METODOLOGI PENELITIAN</b> .....	<b>19</b>
3.1	Waktu dan Tempat Penelitian .....	19
3.2	Alat Pendukung .....	19
3.3	Tahapan Penelitian .....	20
3.3.1	<i>Planning</i> .....	21
3.3.2	<i>Design</i> .....	21
3.3.3	<i>Coding</i> .....	46
3.3.4	<i>Testing</i> .....	47
3.3.5	<i>Release</i> .....	52

<b>IV. HASIL DAN PEMBAHASAN .....</b>	<b>53</b>
4.1 Analisis Kebutuhan .....	53
4.2 Implementasi Sistem .....	53
4.3 Pengujian Sistem .....	76
4.3.1 Pengujian Versi Android.....	76
4.3.2 Pengujian Ukuran Layar .....	77
4.3.3 Pengujian <i>User Interface</i> Aplikasi versi Penjual .....	77
4.3.4 Pengujian <i>User Interface</i> Aplikasi versi Distributor.....	78
<b>V. KESIMPULAN DAN SARAN .....</b>	<b>80</b>
5.1 Kesimpulan.....	80
5.2 Saran .....	80

## DAFTAR TABEL

	Halaman
Tabel 1. Simbol <i>Use Case Diagram</i> .....	10
Tabel 2. Simbol <i>Activity Diagram</i> .....	12
Tabel 3. Simbol <i>Sequence Diagram</i> .....	12
Tabel 4. Simbol <i>Class Diagram</i> .....	13
Tabel 5. Rancangan Daftar Pengujian Fungsional Aplikasi .....	48
Tabel 6. Pengujian Versi Android.....	76
Tabel 7. Pengujian Ukuran Layar .....	77
Tabel 8. Pengujian <i>User Interface</i> Aplikasi versi Penjual .....	78
Tabel 9. Pengujian <i>User Interface</i> Aplikasi versi Distributor.....	79

## DAFTAR GAMBAR

	Halaman
Gambar 1. Proses <i>Extreme Programming</i> .....	14
Gambar 2. Diagram Alir Penelitian .....	20
Gambar 3. <i>Use Case Diagram</i> Aplikasi.....	22
Gambar 4. <i>Activity Diagram</i> Penjual Menawar Harga Barang.....	24
Gambar 5. <i>Activity Diagram</i> Distributor Menawarkan Barang .....	25
Gambar 6. <i>Activity Diagram</i> Penjual Meminta Barang .....	26
Gambar 7. <i>Activity Diagram</i> Penjual Melaporkan Transaksi .....	27
Gambar 8. <i>Activity Diagram</i> Meminta Laporan Transaksi .....	28
Gambar 9. <i>Activity Diagram</i> Melihat Transaksi .....	29
Gambar 10. <i>Class Diagram</i> Aplikasi Manajemen Barang Konsinyasi.....	30
Gambar 11. <i>Layout Splash Screen</i> .....	31
Gambar 12. <i>Layout Sign Up</i> .....	32
Gambar 13. <i>Layout Log In</i> .....	33
Gambar 14. <i>Layout Tab</i> Transaksi .....	34
Gambar 15. <i>Layout Detail</i> Transaksi .....	35
Gambar 16. <i>Layout Form</i> Laporan Transaksi.....	36
Gambar 17. <i>Layout Profil</i> .....	37
Gambar 18. <i>Layout Edit</i> Profil.....	38

Gambar 19. <i>Layout Edit</i> Profil Usaha .....	39
Gambar 20. <i>Layout</i> Riwayat Transaksi .....	40
Gambar 21. <i>Layout Tab</i> Stok .....	41
Gambar 22. <i>Layout</i> Tambah Barang .....	42
Gambar 23. <i>Layout Edit</i> Barang .....	43
Gambar 24. <i>Layout Tab</i> Barang .....	44
Gambar 25. <i>Layout Tab</i> Penjual .....	45
Gambar 26. Arsitektur aplikasi .....	46
Gambar 27. Halaman <i>Splash Screen</i> .....	54
Gambar 28. Halaman <i>Sign Up</i> .....	56
Gambar 29. Halaman <i>Log In</i> .....	57
Gambar 30. JSON <i>Login Response</i> .....	58
Gambar 31. Halaman <i>Tab</i> Transaksi .....	59
Gambar 32. Halaman Detail Transaksi .....	60
Gambar 33. Halaman Verifikasi Pelunasan .....	61
Gambar 34. Halaman <i>Form</i> Laport Transaksi .....	62
Gambar 35. Halaman Profil .....	63
Gambar 36. Pengambilan data menggunakan token. ....	64
Gambar 37. Pengambilan data tanpa token. ....	64
Gambar 38. Halaman <i>Edit</i> Profil .....	65
Gambar 39. Halaman <i>Edit</i> Profil Usaha .....	66
Gambar 40. Halaman Tentukan Lokasi .....	67
Gambar 41. Halaman Riwayat Transaksi .....	68
Gambar 42. Halaman <i>Tab</i> Stok .....	69



Gambar 43. Halaman Tambah Barang.....	70
Gambar 44. Halaman <i>Edit</i> Barang .....	71
Gambar 45. Halaman <i>Tab</i> Barang.....	72
Gambar 46. Halaman <i>Tab</i> Penjual .....	73
Gambar 47. Halaman Buat Transaksi .....	74
Gambar 48. Halaman Buat Tawaran Harga Barang.....	75

## DAFTAR KODE PROGRAM

	Halaman
Kode Program 1. Fungsi meminta perizinan permintaan lokasi .....	55
Kode Program 2. Fungsi pengambilan dan penyimpanan titik koordinat.....	55
Kode Program 3. Fungsi Penyimpanan token ke dalam <i>Shared Preference</i> . .....	58
Kode Program 4. Penggunaan token untuk autentikasi .....	64

## **I. PENDAHULUAN**

### **1.1 Latar Belakang**

Usaha Mikro, Kecil, dan Menengah (UMKM) seringkali disebut sebagai salah satu pilar perekonomian di Indonesia. Fleksibilitas dan kemampuan UMKM dalam menyesuaikan diri terhadap kondisi pasar yang berubah dengan sangat cepat menyebabkan UMKM lebih diperhitungkan dibandingkan dengan perusahaan skala besar (Partomo dan Soejoedono, 2002).

Saat ini, menurut data dari Kementerian Koperasi dan UKM Republik Indonesia tahun 2017, diketahui bahwa UMKM memiliki pangsa pasar 99,99% atau 62,9 juta unit dari total keseluruhan pelaku usaha di Indonesia, sementara usaha besar hanya 0,01% atau 5400 unit. Dalam hal penyerapan tenaga kerja, UMKM menyerap 97% tenaga kerja nasional. Presentase tersebut menggambarkan besarnya kontribusi UMKM terhadap perekonomian negara.

Akan tetapi dalam perkembangannya, peningkatan jumlah UMKM di Indonesia berbanding terbalik dengan peningkatan levelnya. Tercatat dalam kurun waktu tahun 2007-2017, jumlah UMKM mengalami pertumbuhan sampai 25,48%, tetapi peningkatan level dari Usaha Mikro ke Usaha Kecil hanya 5,49%. Begitupula dari Usaha Kecil ke Usaha Menengah yang hanya menyentuh angka 3,42%. Hal ini

tentunya menjadi tugas besar dari generasi untuk meningkatkan kinerja UMKM agar lebih optimal.

Permasalahan yang juga dihadapi UMKM di Indonesia ialah kurangnya pemahaman tentang keuangan dan akuntansi. Sistem pembukuan keuangan yang baik adalah hal utama dalam perencanaan suatu perusahaan. Minimnya pengetahuan tentang metode pemasaran produk yang baik terbukti menjadi masalah dalam perkembangan UMKM di Indonesia. Faktor ini menyebabkan penjualan produk dengan modal besar menjadi kurang maksimal (Niode, 2009).

Usaha yang umumnya dilakukan UMKM untuk meningkatkan penjualan apabila memiliki modal terbatas adalah dengan sistem konsinyasi. Konsinyasi adalah sistem penjualan yang mana pemilik barang (*consigner*) menitipkan barang dagangannya kepada pihak yang menjual barang (*consignee*) untuk kemudian dijual dengan memberikan komisi tertentu (Widayat, 1994).

Pada beberapa penelitian digunakan aplikasi *Microsoft Excel* sebagai alat bantu penjualan yang dimiliki para pelaku UMKM untuk manajemen penjualan mereka. Namun penggunaan *Microsoft Excel* memiliki kekurangan seperti modal yang harus dikeluarkan untuk membeli sebuah komputer serta menambah gaji petugas pencatat yang siaga di komputer. Hal ini berakibat sulitnya pelaku UMKM dalam meningkatkan level usaha. Berdasarkan data yang dihimpun oleh Waiwai Marketing pada tahun 2015, terdapat 41 juta pengguna aktif Android di Indonesia. Data ini menempatkan Indonesia sebagai negara dengan pengguna Android terbanyak di Asia Tenggara.

Pembuatan aplikasi Android telah banyak menciptakan solusi-solusi baru atas permasalahan sosial yang ada. Aplikasi Android merupakan aplikasi yang dirancang khusus untuk *platform mobile* seperti iOS, Android, atau *Windows Mobile* (Pressman and Maxim, 2014). Keunggulan utama dari aplikasi Android adalah memberikan kemudahan pengguna untuk mendapatkan informasi secara *portable* (Turban et al., 2018).

Penggunaan *platform mobile* atau *smartphone* yang semakin meningkat membuat aplikasi Android menjadi aplikasi yang umum digunakan sehari-hari. Selain mudah dipakai, aplikasi Android juga mudah dibawa serta dapat digunakan di berbagai tempat (Islam and Mazumder, 2010). Penggunaan aplikasi Android untuk manajemen penjualan barang konsinyasi akan semakin maksimal apabila didukung oleh sifatnya yang fleksibel.

Berdasarkan latar belakang tersebut, penelitian ini akan membahas rancang bangun suatu aplikasi manajemen penjualan dengan sistem konsinyasi. Aplikasi ini memungkinkan UMKM dan warung/minimarket untuk memanajemen penjualan konsinyasi mereka secara lebih rapi dan cepat. Terdapat juga fitur yang memungkinkan pengguna dalam menemukan tempat terbaik untuk menjual barang dagangan mereka.

## **1.2 Rumusan Masalah**

Berdasarkan identifikasi masalah yang telah dipaparkan, maka permasalahan yang dikaji dalam penelitian ini adalah bagaimana merancang dan membangun suatu

aplikasi Android yang dapat digunakan untuk mempermudah manajemen penjualan barang konsinyasi.

### **1.3 Batasan Masalah**

Batasan masalah dalam penelitian ini yaitu sebagai berikut:

1. Aplikasi hanya dapat dijalankan melalui sistem operasi Android dengan minimal versi 5.1 Lollipop.
2. Membutuhkan koneksi internet untuk dapat beroperasi.
3. Aplikasi ini memiliki dua tingkatan pengguna, yaitu Distributor sebagai pemilik barang yang menitipkan barang konsinyasi dan Penjual yang menjual barang konsinyasi.
4. Tahap pengujian menggunakan *Black-Box Testing*.

### **1.4 Tujuan**

Tujuan dari penelitian ini adalah dibangunnya sebuah aplikasi manajemen sistem penjualan barang dengan sistem konsinyasi berbasis Android.

### **1.5 Manfaat**

Penelitian ini diharapkan dapat membantu manajerial Distributor maupun Penjual agar dapat memaksimalkan profit yang diharapkan, sehingga dapat memacu pertumbuhan ekonomi di Indonesia dengan menggunakan perangkat bergerak berbasis Android.



## II. TINJAUAN PUSTAKA

### 2.1 Penjualan

Penjualan adalah kegiatan menukar barang atau jasa dengan alat tukarnya. Pada perusahaan, penjualan meliputi penjualan tunai dan penjualan kredit. Penjualan secara kredit menimbulkan piutang dagang. Pada umumnya terdapat syarat-syarat tertentu yang harus ditunaikan saat melakukan penjualan, misalnya ongkos angkut dan asuransi (Sumarsan, 2017).

### 2.2 Konsinyasi

Konsinyasi diartikan sebagai kegiatan penitipan barang dagangan kepada agen jual baik individu maupun lembaga untuk dijual dengan pembayaran kemudian (Kamus Besar Bahasa Indonesia, 2017). Penjualan konsinyasi melibatkan dua pihak yaitu pihak yang memiliki barang disebut Pengamanat (*consignor*) dan pihak yang mengusahakan penjualan barang disebut Komisioner (*consignee*) (Simamora, 2008). Penjualan konsinyasi dalam pengertian sehari-hari dikenal dengan sebutan ‘titip jual’. Sebelum melakukan kegiatan konsinyasi, mereka menyusun suatu kontrak perjanjian dagang atau akad dagang. Ketentuan-ketentuan dalam penjualan konsinyasi pada umumnya dinyatakan secara tertulis

dan menekankan sifat hubungan kerjasama antara kedua belah pihak (Yendrawati, 2005).

Transaksi dengan cara penjualan konsinyasi mempunyai keuntungan tertentu dibandingkan dengan penjualan secara langsung. Adapun keuntungan bagi Distributor adalah:

1. Konsinyasi merupakan cara untuk memperluas pasaran yang dapat dijamin oleh seorang produsen, pabrikan, atau distributor, terutama apabila:
  - a. Barang-barang yang bersangkutan baru diperkenalkan.
  - b. Permintaan produk tidak menentu.
  - c. *Branding* belum terkenal.
2. Harga barang yang bersangkutan tetap dapat dikontrol oleh distributor. Hal ini disebabkan kepemilikan atas barang tersebut masih dikendalikan oleh distributor sehingga harga masih bisa dijangkau.
3. Jumlah barang yang dijual dan stok di gudang akan lebih mudah dikontrol sehingga risiko kekurangan atau kelebihan stok dapat ditekan sehingga memudahkan pengamanat dalam memperkirakan rencana produksi.

Keuntungan untuk penjual diantaranya:

1. Penjual tidak dibebani risiko menanggung kerugian bila gagal dalam penjualan barang konsinyasi.
2. Kebutuhan akan modal kerja dapat dikurangi, sebab penjual hanya berfungsi sebagai penerima dan penjual barang konsinyasi untuk distributor.

3. Penjual berhak mendapatkan komisi dari hasil penjualan barang sesuai dengan perjanjian (Sabeni, 1999).

### **2.3 Aplikasi Android**

Aplikasi merupakan suatu perangkat lunak yang menjadi *front-end* dalam suatu sistem, dan digunakan untuk mengolah data menjadi suatu informasi serta berguna bagi pihak yang bersangkutan. Android adalah sistem operasi berbasis Linux yang dirancang untuk perangkat bergerak layar sentuh seperti telepon pintar dan komputer tablet. Android awalnya dikembangkan oleh Android, Inc. dengan dukungan finansial dari Google, yang kemudian membelinya pada tahun 2005. Sistem Operasi ini secara resmi dirilis pada tahun 2007, bersamaan dengan didirikannya *Open Handset Alliance* (OHA), suatu konsorsium dari perusahaan-perusahaan perangkat keras, perangkat lunak, dan telekomunikasi yang bertujuan untuk memajukan standar terbuka perangkat seluler. Ponsel Android pertama mulai dijual pada bulan Oktober 2008 (Yudhanto dan Wijayanto, 2018). Pressman dan Bruce (2014) berpendapat bahwa aplikasi Android adalah aplikasi yang telah dirancang khusus untuk *platform mobile* seperti iOS, Android, dan Windows Mobile.

### **2.4 Android Studio**

Android Studio adalah *Integrated Development Environment* (IDE) resmi untuk mengembangkan aplikasi Android yang dibuat langsung oleh Google selaku pimpinan Konsorsium OHA yang mengembangkan Android. Android Studio dirilis 16 Mei 2013 pada acara Google I/O Conference tahun 2013. Sejak saat itu,

Android Studio resmi menggantikan Eclipse sebagai IDE resmi untuk mengembangkan aplikasi Android (Suryana, 2018).

## **2.5 *Unified Modeling Language (UML)***

*Unified Modeling Language (UML)* adalah tujuan umum, perkembangan, dan bahasa permodelan di bidang rekayasa perangkat lunak yang dimaksudkan untuk menyediakan standar lisensi dalam memvisualisasikan desain sistem. UML adalah metodologi kolaborasi antara metode Booch, OMT (*Object Modeling Technique*), dan OOSE (*Object Oriented Software Engineering*). UML merupakan metodologi yang paling sesuai digunakan untuk analisis dan perancangan sistem dengan metodologi berorientasi objek yang mengadaptasi maraknya penggunaan bahasa pemrograman berorientasi objek (OOP) (Nugroho, 2009).

### **2.5.1 *Penggunaan Unified Modeling Language (UML)***

Menurut Henderi (2009), langkah-langkah penggunaan *Unified Modeling Language (UML)* adalah sebagai berikut.

- a. Membuat daftar proses bisnis dari level tertinggi untuk mendefinisikan aktivitas dan proses yang mungkin muncul.
- b. Memetakan *use case* untuk setiap proses bisnis untuk mendefinisikan dengan tepat fungsional yang harus disediakan oleh sistem. Setelah itu, sederhanakan *use case diagram* dan lengkapi dengan *requirement*, *constraints*, dan catatan-catatan lain.

- c. Membuat *deployment diagram* secara kasar untuk mendefinisikan arsitektur fisik sistem.
- d. Mendefinisikan *requirement* non-fungsional lainnya dan *security* yang juga harus disediakan oleh sistem.
- e. Membuat *activity diagram* berdasarkan *use case diagram*.
- f. Mendefinisikan objek-objek level atas *package* atau *domain* dan membuat *sequence* atau *collaboration* untuk tiap alur pekerjaan. Jika suatu *use case* memiliki kemungkinan alur normal dan *error*, buat lagi satu diagram untuk masing-masing alur.
- g. Membuat rancangan model *user interface* yang menyediakan antar muka bagi pengguna untuk menjalankan skenario *use case*.
- h. Membuat *class diagram* berdasarkan model-model yang sudah ada. Setiap *package* atau domain dipecah menjadi hirarki *class* lengkap dengan atribut dan metodenya. Langkah ini akan lebih baik jika untuk setiap *class* dibuat unit test untuk menguji fungsionalitas *class* dan interaksi dengan *class* lain.
- i. Setelah *class diagram* dibuat, komponen diagram juga dibuat agar dapat melihat kemungkinan pengelompokkan *class* menjadi komponen-komponen. Selain itu, definisikan juga tes integrasi untuk setiap komponen agar memastikan komponen bereaksi dengan baik.
- j. Memperhalus *deployment diagram* yang sudah dibuat. Memperinci kemampuan dan *requirement* piranti lunak, sistem operasi, jaringan dan sebagainya. Petakan komponen ke dalam node.
- k. Pembangunan sistem dimulai. Ada dua pendekatan yang tepat digunakan, yaitu pendekatan *use case* dengan meng-*assign* setiap *use case* kepada tim

pengembang tertentu untuk mengembangkan unit kode yang lengkap dengan tes, atau menggunakan pendekatan komponen. Pendekatan komponen yaitu meng-*assign* setiap komponen kepada tim pengembang tertentu (Henderi, 2009).

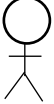


### 2.5.2 Pemodelan *Unified Modeling Language* (UML)



*Unified Modeling Language* (UML) dapat dimodelkan dengan *use case diagram*, *activity diagram*, *sequence diagram*, dan *class diagram*.

#### a. *Use Case Diagram*




*Use Case Diagram* adalah gambaran dari beberapa atau seluruh aktor dan *use case* dengan tujuan mengenali interaksi mereka dalam suatu sistem (Triandini dan Suardika, 2012). *Use case diagram* menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Hal yang ditentukan adalah “apa” yang perlu dilakukan sistem, bukan “bagaimana”. Suatu *use case* merepresentasikan interaksi antara aktor dengan sistem. Simbol *use case diagram* disajikan pada Tabel 1.

Tabel 1. Simbol *Use Case Diagram*

No.	Simbol	Nama	Keterangan
1		<i>Actor</i>	Menjelaskan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Generalization</i>	Hubungan yang merepresentasikan objek anak berbagi perilaku dengan struktur data dari objek yang ada di atas objek induk ( <i>ancestor</i> ).
3		<i>Include</i>	<i>Use case</i> sumber secara eksplisit

4		<i>Extend</i>	Menjelaskan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> .
5		<i>Association</i>	Menghubungkan objek satu dengan objek yang lainnya.






Tabel 1. (lanjutan)

6		Sistem	Menjelaskan paket yang menampilkan sistem secara terbatas
7		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan oleh sistem
8		<i>Dependency</i>	Hubungan yang merepresentasikan perubahan yang terjadi pada suatu elemen mandiri (independen) akan mempengaruhi elemen yang bergantung padanya, yaitu elemen yang tidak mandiri (dependen)

b. *Activity Diagram*

*Activity Diagram* merupakan suatu tipe dari diagram *workflow* yang menggambarkan tentang aktivitas dari pengguna ketika melakukan setiap kegiatan dan aliran sekuensial (Triandini dan Suardika, 2012). *Activity Diagram* akan sangat berguna pada awal pemodelan proses karena akan membantu memahami keseluruhan proses. *Activity Diagram* juga bermanfaat untuk menggambarkan interaksi antara beberapa *use case*. Simbol *activity diagram* disajikan pada Tabel 2.

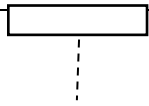
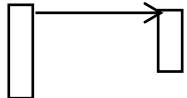
Tabel 2. Simbol *Activity Diagram*

No.	Simbol	Nama	Keterangan
1		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi.
2		<i>Action</i>	Pernyataan dari sistem yang mencerminkan eksekusi dari suatu aksi.
3		<i>Initial Node</i>	Objek dibentuk atau diawali.
4		<i>Activity Final Node</i>	Objek yang dibentuk dan diselesaikan atau diakhiri.
5		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran.

c. *Sequence Diagram*

*Sequence diagram* adalah grafik dua dimensi yang menyajikan objek dalam dimensi horizontal, sedangkan *life line* ditunjukkan dalam dimensi vertikal. *Sequence Diagram* menggambarkan interaksi antar objek di dalam dan di sekitar termasuk pengguna dan *display* berupa pesan yang digambarkan terhadap waktu. Simbol *Sequence Diagram* disajikan pada Tabel 3.

Tabel 3. Simbol *Sequence Diagram*


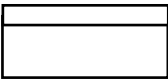
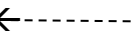
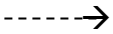

No	Simbol	Nama	Keterangan
1		<i>Life Line</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
2		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktivitas yang terjadi.



d. *Class Diagram*

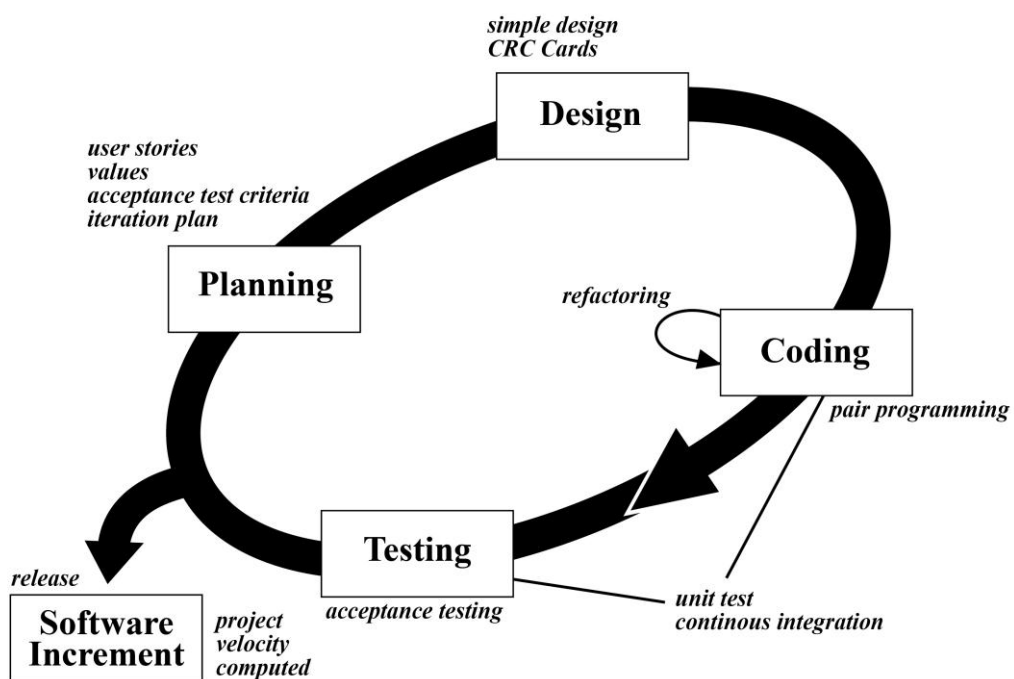
*Class diagram* merupakan himpunan dari objek-objek yang sejenis. Suatu objek memiliki keadaan sesaat (*state*) dan perilaku (*behavior*). *State* suatu objek adalah kondisi objek tersebut yang dinyatakan dalam *attribute*. Perilaku suatu objek mendefinisikan bagaimana objek bereaksi dan memberikan reaksi. *Class diagram* merupakan salah satu diagram utama dari UML untuk menggambarkan *class* pada suatu sistem. Analisis pembuatan *class diagram* merupakan aktivitas inti yang sangat mempengaruhi arsitektur perangkat lunak yang dirancang hingga ke tahap pengkodean. *Class diagram* juga memiliki tiga area pokok yaitu nama atribut dan metode. Simbol *class diagram* disajikan pada Tabel 4.

Tabel 4. Simbol *Class Diagram*

No	Simbol	Nama	Keterangan
1		<i>Generalization</i>	Hubungan yang merepresentasikan objek anak ( <i>descendent</i> ) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk.
2		<i>Message</i>	Himpunan dari objek yang berbagi atribut serta operasi yang sama.
3		<i>Realization</i>	Operasi yang benar-benar dilakukan suatu objek.
4		<i>Dependency</i>	Hubungan yang merepresentasikan perubahan yang terjadi pada suatu elemen mandiri mempengaruhi elemen yang bergantung padanya.
5		<i>Association</i>	Menghubungkan antara objek satu dengan objek yang lain.

## 2.6 *Extreme Programming (XP)*

*Extreme Programming* merupakan salah satu metodologi perangkat lunak yang banyak digunakan untuk mengembangkan aplikasi. XP pertama kali diperkenalkan menjadi metodologi pengembangan perangkat lunak oleh Kent Beck. XP sangat cocok untuk pengembangan proyek yang memerlukan adaptasi cepat dalam perubahan-perubahan yang terjadi selama pengembangan aplikasi. Selain itu, XP juga cocok untuk anggota tim yang tidak terlalu banyak dan berada pada lokasi yang sama dalam pengembangan sistem (Suryantara, 2017). Terdapat empat tahapan dalam pengembangan perangkat lunak menggunakan XP, yaitu tahap *planning*, *design*, *coding*, dan *testing*. Tahapan-tahapan XP diilustrasikan pada Gambar 1.



Gambar 1. Proses *Extreme Programming* (Aristoteles dkk., 2018)

*Planning* merupakan aktivitas yang dimulai dengan membentuk *user stories*. Desain merupakan fase ketika *user stories* yang diperoleh dirancang ke dalam desain. *Coding* merupakan fase pengembangan sistem ke dalam bentuk kode-kode program. Pada fase *coding*, sistem dapat dirilis dalam bentuk kecil dan ditunjukkan kepada *customer*, sehingga apabila terjadi perubahan akan dilakukan pengkodean ulang. *Testing* merupakan fase pengujian terhadap sistem yang dibuat. Apabila sistem telah disetujui oleh *customer*, maka sistem akan dirilis.

## 2.7 Database

*Database* merupakan sekumpulan data yang saling terhubung secara logis, yang mendeskripsikan data yang ada. *Database* didesain untuk menemukan keperluan informasi pada perusahaan (Connolly and Begg 2010). Menurut Wahana Komputer (2010), *database* mempunyai kelebihan dalam penyimpanan data yaitu:

### a. *Simple*

Model relasional mempunyai model penyimpanan yang berbentuk tabel dan merupakan metode penyimpanan data yang tidak *redundant*. Setiap tabel didesain relasional untuk objek tunggal yang mengandung data, yang konsekuen terhadap aspek tertentu dari *database*.

### b. Fleksibilitas

*Database* mempunyai kemampuan untuk meng-*update* nilainya di suatu tempat apabila data berubah. Semua *query*, *form*, dan data lain akan berubah secara konsekuen dengan nilai yang baru (Wahana Komputer, 2010).

## **2.8 *Shared Preference***

*Shared Preference* adalah sebuah class yang berfungsi sebagai tempat penyimpanan data primitive pada file internal aplikasi dalam bentuk *key-value*. *Shared Preference* menyediakan method sederhana untuk membaca dan menulis data. Class ini digunakan untuk menyimpan nilai-kunci dari data transfer yang kita terima dengan ukuran data yang relative lebih kecil (Google Developers, 2020).

## **2.9 REST API**

*Representational State Transfer* (REST) merupakan gaya arsitektur tentang bagaimana data dapat dikirim dengan elegan berdasarkan permintaan dan respon dari klien. Sementara API adalah akronim dari *Application Programming Interface* atau dalam bahasa Indonesia diterjemahkan sebagai Antarmuka Pemrograman Aplikasi. API merupakan kendaraan yang membawa data dalam komunikasi antar aplikasi. API memungkinkan aplikasi berkomunikasi dengan aplikasi lain tanpa harus mengetahui bagaimana data dibentuk demi keamanan data (Pranata dkk. 2018).

## **2.10 JSON (*Javascript Object Notation*)**

JSON atau *Javascript Object Notation* berasal dari literal bahasa pemrograman JavaScript. JSON bukanlah termasuk suatu bahasa pemrograman, namun JSON adalah sebuah format skrip yang digunakan untuk pertukaran data (Pranata dkk. 2018).

JSON merupakan format teks yang tidak bergantung pada bahasa pemrograman apapun karena menggunakan gaya bahasa yang umum digunakan oleh *Programmer* keluarga C termasuk C, C++, C#, Java, JavaScript, Perl, Python dll. Oleh karena sifat-sifat tersebut, menjadikan JSON ideal sebagai bahasa pertukaran data.

### **2.11 Location Based Service**

*Location Based Service* (LBS) merupakan layanan informasi berbasis lokasi yang dapat diakses menggunakan perangkat bergerak melalui jaringan Internet dan seluler serta memanfaatkan kemampuan penunjuk lokasi pada piranti *mobile*. Dengan kata lain LBS merupakan kemampuan sebuah perangkat bergerak dengan bantuan GPS untuk menunjukkan suatu letak lokasi (Nurmayanti dkk., 2017).

### **2.12 Black Box Testing**

*Black Box Testing* merupakan teknik pengujian perangkat lunak yang berfokus pada domain informasi dari perangkat lunak tersebut. *Black Box Testing* juga dapat diartikan sebagai metode pengujian yang menguji fungsionalitas sistem (Indriasari, 2012). Metode ini dilakukan untuk memastikan apakah fungsi perangkat lunak berjalan dengan benar jika diberikan masukan yang bervariasi. Menurut Paramarta (2013), salah satu pengujian *Black Box* adalah *equivalence partitioning*.

*Equivalence partitioning* dilakukan dengan cara mengembangkan sistem dan mengidentifikasi kelas data yang mungkin dimasukan pengguna sistem ke

dalam antarmuka yang disediakan, baik kelas data yang benar maupun kelas data yang salah. Kelas data yang sudah diidentifikasi kemudian diuji ke dalam antarmuka yang ada agar terlihat apakah fungsional sistem yang sudah disediakan berjalan dengan baik atau tidak. Berdasarkan hasil pengujian ini, didapat hasil bahwa seluruh fungsional yang telah dibangun sudah bisa melayani dengan baik untuk kelas data yang benar ataupun kelas data yang salah (Paramarta, 2013).

### **III. METODOLOGI PENELITIAN**

#### **3.1 Waktu dan Tempat Penelitian**

Penelitian ini dilakukan di Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Lampung yang berada di Jalan Soemantri Brojonegoro No.1 Gedung Meneng, Rajabasa, Bandar Lampung. Penelitian ini dilaksanakan mulai 1 November 2019 sampai dengan November 2020.

#### **3.2 Alat Pendukung**

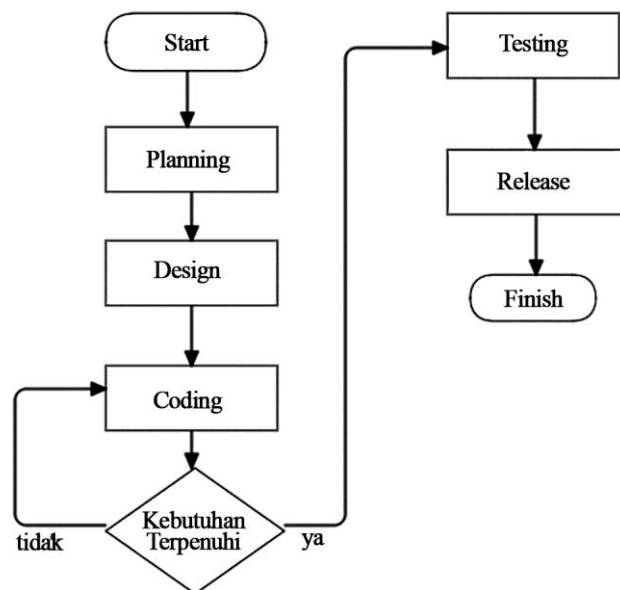
Alat pendukung yang digunakan dalam penelitian ini adalah sebagai berikut.

1. Perangkat Keras
  - i. Laptop Intel® Core™ i3-2365 CPU @1.40GHz (4 CPUs), 10GB RAM.
  - ii. *Smartphone* Android (versi 9.0 Pie).
2. Perangkat Lunak
  - i. Sistem Operasi Linux Ubuntu 20.04.
  - ii. Android Studio, digunakan untuk membuat aplikasi Android.
  - iii. Visual Studio Code, digunakan untuk membuat aplikasi REST API.
  - iv. Postman, digunakan untuk menguji REST API.
  - v. Mendeley Desktop, digunakan untuk mengecek referensi pustaka.

- vi. WPS Office, digunakan untuk membuat penulisan penelitian.
- vii. Figma, digunakan untuk membuat desain *wireframe* dan *user interface*.
- viii. Creatly Web Apps, digunakan untuk membuat *use case diagram*, *activity diagram*, dan *class diagram*.

### 3.3 Tahapan Penelitian

Tahapan penelitian merupakan langkah-langkah yang dilakukan dalam melakukan penelitian. Adapun tahapan penelitian yang dilakukan dalam membangun aplikasi manajemen penjualan barang dengan sistem konsinyasi berbasis Android diilustrasikan pada Gambar 2.



Gambar 2. Diagram Alir Penelitian

Diagram alir metodologi penelitian pada Gambar 2 menggambarkan beberapa tahapan yaitu *Planning*, *Design*, *Coding*, *Testing*, dan *Release*.



### 3.3.1 *Planning*

Tahap *planning* dalam penelitian ini berupa identifikasi permasalahan yang ada serta metode penyelesaian masalah tersebut dengan membentuk *user stories* yang dijadikan acuan dalam pengembangan aplikasi. *User stories* dibentuk berdasarkan permasalahan yang ada di lapangan ketika melakukan manajemen penjualan barang dengan sistem konsinyasi. *User stories* didapat dari wawancara yang dilakukan secara langsung dengan pihak yang sering melakukan penjualan dengan sistem konsinyasi di Universitas Lampung.

#### a. *User Stories*

*User Stories* digunakan untuk menggambarkan *output* aplikasi, fitur, dan fungsi-fungsi yang akan dibuat. Adapun *user stories* dalam penelitian ini yaitu:

- i. Aplikasi dapat membantu distributor mengelola stok barang dagangannya.
- ii. Aplikasi dapat membantu distributor menemukan penjual barang yang tepat.
- iii. Aplikasi dapat membantu penjual untuk menemukan barang yang akan dijual.
- iv. Aplikasi dapat menyimpan seluruh detail transaksi yang pernah terjadi antar penjual dan distributor.

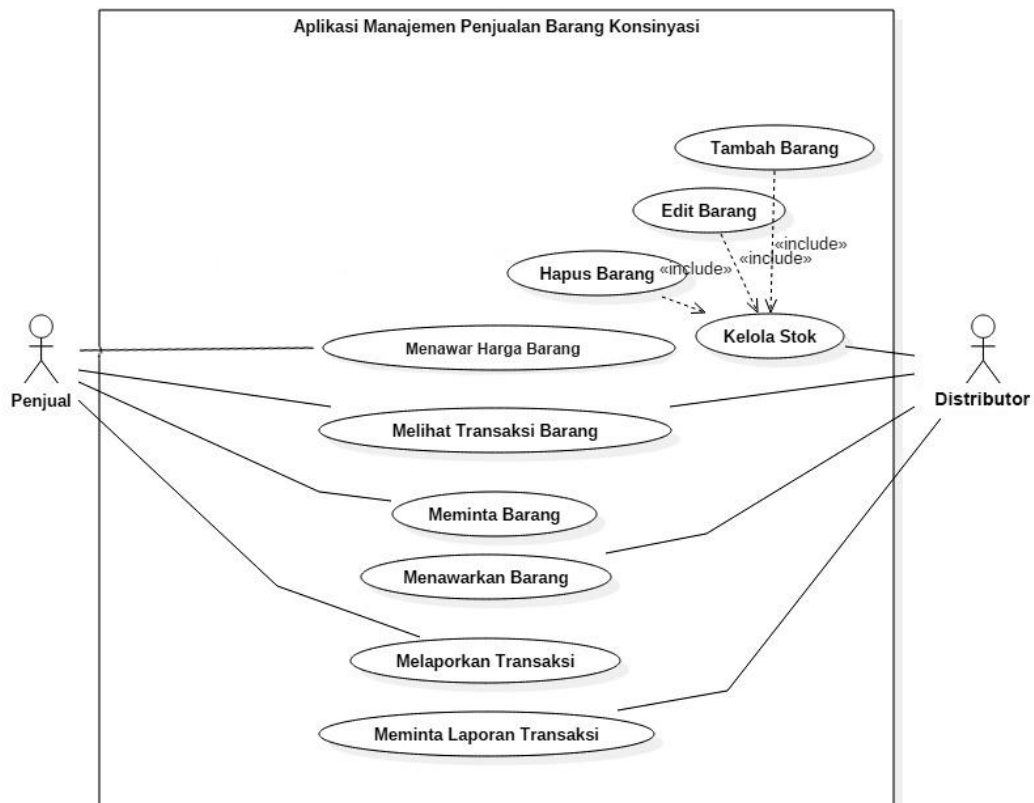
### 3.3.2 *Design*

Tahap *design* pada penelitian ini berupa rancangan aplikasi yang akan dibuat berdasarkan *user stories* yang sudah didapat pada tahap *planning*. Dalam

penelitian ini langkah-langkah yang digunakan untuk merancang aplikasi yaitu menggunakan *Unified Modeling Language* (UML). Pada tahap ini, dirancang desain *use case diagram*, *activity diagram*, *sequence diagram*, *class diagram* dan *prototype* aplikasi.

#### a. *Use Case Diagram*

*Use Case Diagram* adalah diagram yang dibuat untuk mendeskripsikan interaksi antara pengguna dengan aplikasi. *Use Case Diagram* aplikasi manajemen penjualan barang konsinyasi disajikan pada Gambar 3.



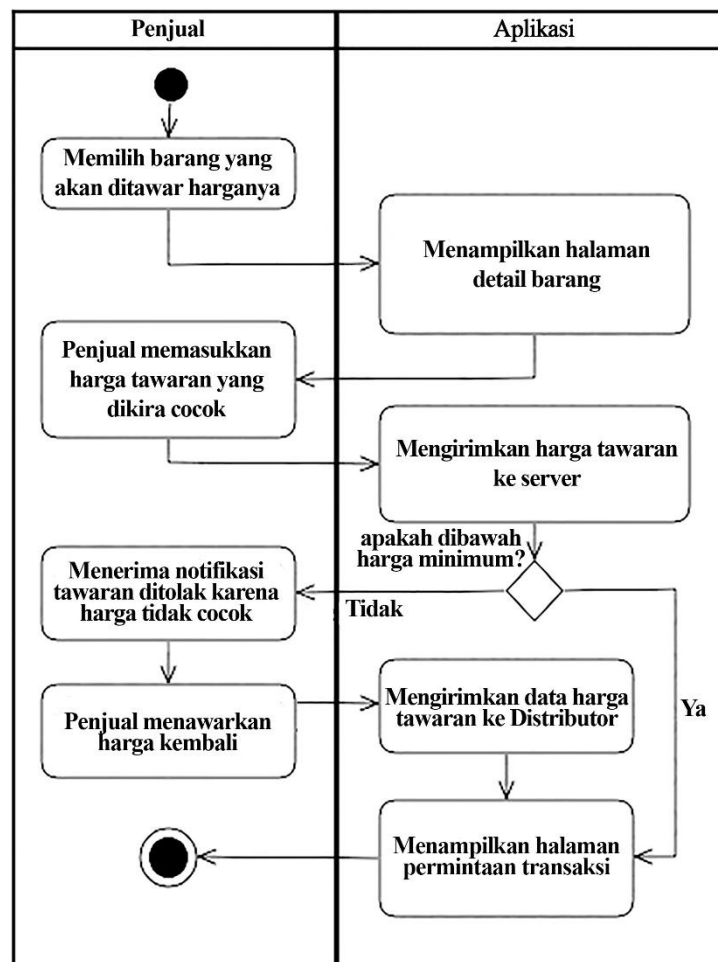
Gambar 3. *Use Case Diagram* Aplikasi

**b. *Activity Diagram***

*Activity diagram* atau diagram aktivitas merupakan gambaran aliran fungsionalitas aplikasi. Adapun penjelasan dari *activity diagram* aplikasi tersebut adalah sebagai berikut.

### 1. Activity Diagram Penjual Menawar Harga Barang

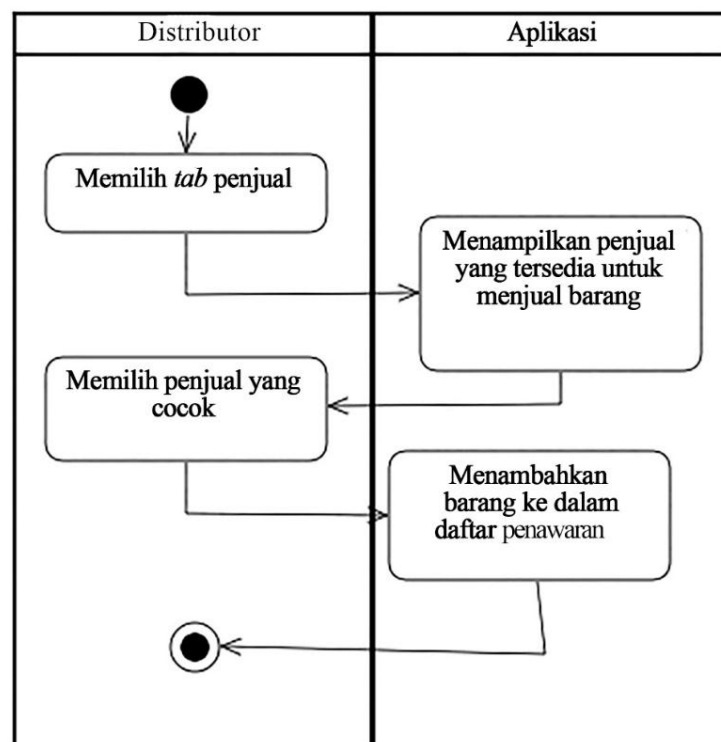
Activity Diagram Penjual Menawar Harga Barang dimulai dari Penjual menawarkan barang yang bisa ditawar. Kemudian harga tawaran dikirimkan ke sistem untuk dicek apakah merugikan distributor atau tidak sebelum akhirnya menjadi transaksi permintaan barang. Di aplikasi versi Distributor, terdapat tanda ceklis untuk menyatakan apakah barang tersebut dapat ditawar atau tidak pada sesi tambah barang atau edit barang. Activity Diagram Penjual Menawar Harga Barang disajikan pada gambar 4.



Gambar 4. Activity Diagram Penjual Menawar Harga Barang

## 2. *Activity Diagram* Distributor Menawarkan Barang

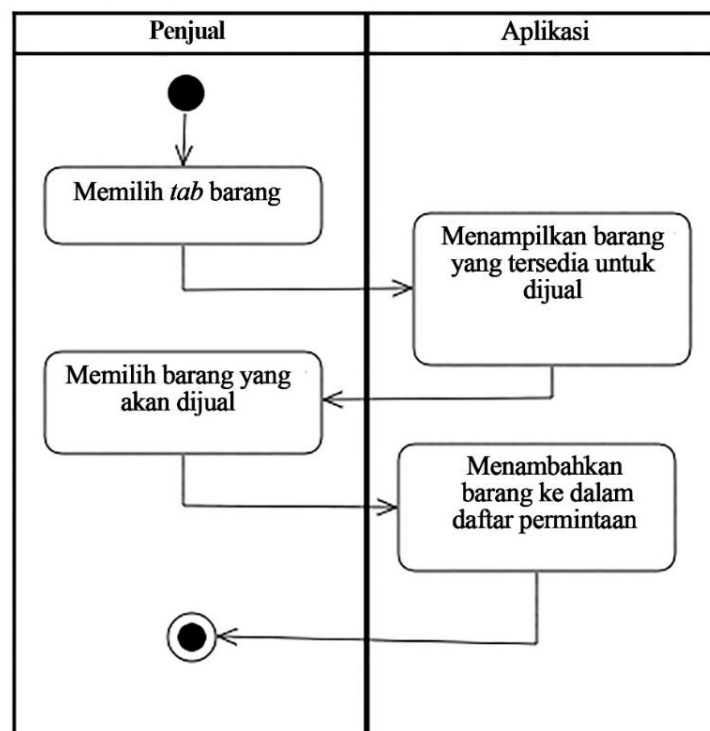
*Activity Diagram* Distributor Menawarkan Barang dimulai dari Distributor membuka aplikasi dan memilih *tab* penjual, kemudian aplikasi akan menampilkan penjual yang tersedia dalam daftar. Distributor diharuskan untuk memilih barang mana yang hendak ditawarkan kepada Penjual yang cocok. Setelah distributor memilih barang yang akan ditawarkan, aplikasi akan menambahkan barang ke dalam daftar penawaran. *Activity Diagram* Distributor Menawarkan Barang dapat dilihat pada Gambar 5.



Gambar 5. *Activity Diagram* Distributor Menawarkan Barang

### 3. *Activity Diagram* Penjual Meminta Barang

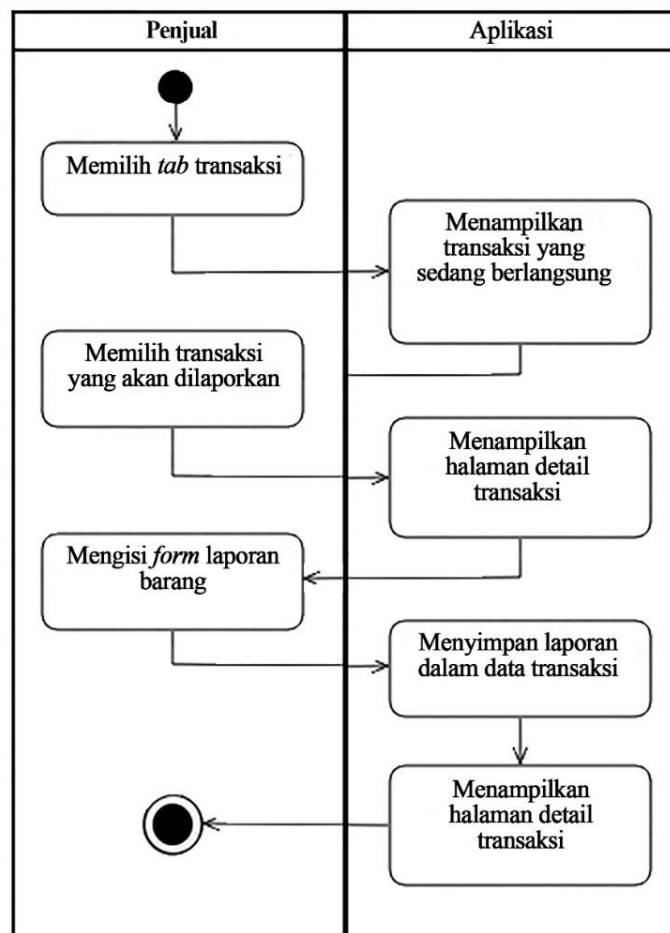
*Activity Diagram* Penjual Meminta Barang dimulai dari penjual membuka aplikasi dan memilih *tab* barang, kemudian aplikasi akan menampilkan barang-barang yang tersedia untuk dijual dalam daftar. Pengguna diharuskan untuk memilih barang mana yang hendak dijual. Setelah pengguna memilih barang yang akan dijual dan menambahkan harga tawaran jika merupakan barang yang bisa ditawar, aplikasi akan menambahkan barang ke dalam daftar permintaan. *Activity Diagram* Penjual Meminta Barang dapat dilihat pada Gambar 6.



Gambar 6. *Activity Diagram* Penjual Meminta Barang

#### 4. Activity Diagram Penjual Melaporkan Transaksi

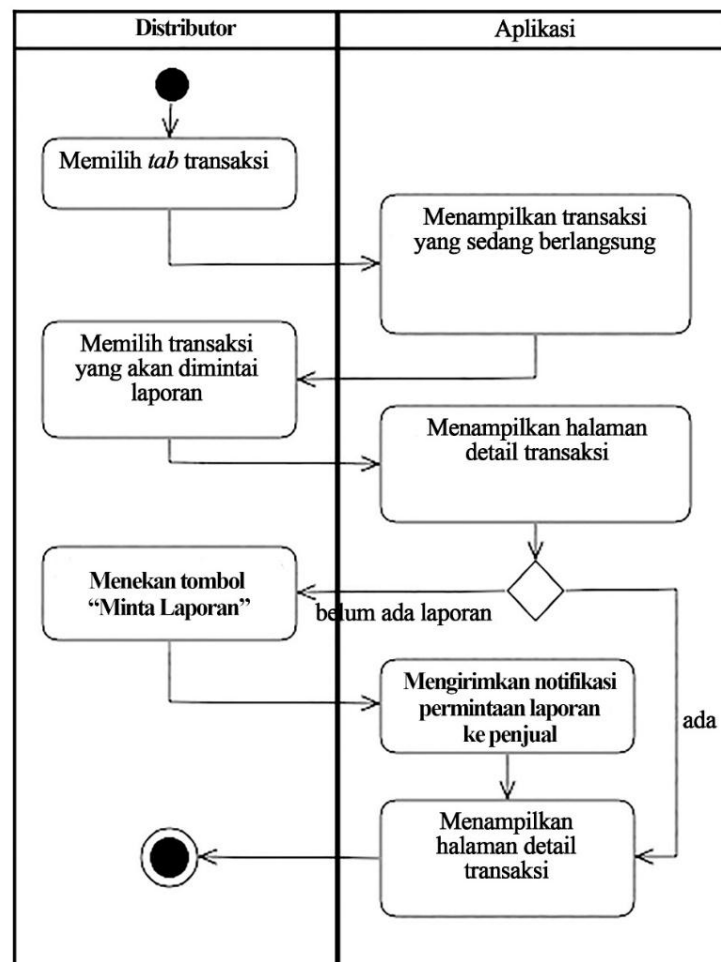
*Activity Diagram* Penjual Melaporkan Transaksi dimulai *tab* transaksi yang akan menampilkan transaksi yang sedang berlangsung saat itu. Penjual diharuskan memilih transaksi yang akan dilaporkan. Setelah aplikasi menampilkan detail dari transaksi yang dipilih, penjual dapat menekan tombol lapor di bagian bawah dan aplikasi akan menampilkan laman formulir laporan untuk diisi jenis laporannya. Kemudian laporan akan disimpan dan akan ditampilkan di detail transaksi. *Activity Diagram* Penjual Melaporkan Transaksi disajikan dalam Gambar 7.



Gambar 7. Activity Diagram Penjual Melaporkan Transaksi

### 5. Activity Diagram Distributor Meminta Laporan Transaksi

Activity Diagram Distributor Meminta Laporan Transaksi dimulai dari distributor memilih *tab* transaksi, kemudian aplikasi akan menampilkan transaksi yang sedang berlangsung pada hari itu. Distributor diharuskan memilih transaksi yang akan diminta untuk melaporkan. Setelah itu, aplikasi akan mengecek apakah laporan tersebut sudah tersedia. Jika belum, tombol minta laporan akan tersedia di bagian bawah. Activity Diagram Meminta Laporan Transaksi disajikan dalam Gambar 8.

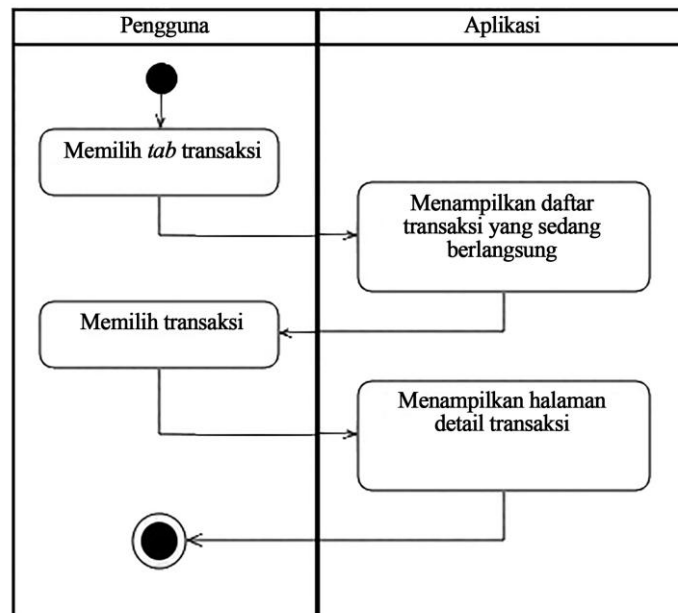


Gambar 8. Activity Diagram Meminta Laporan Transaksi



## 6. *Activity Diagram* Melihat Transaksi

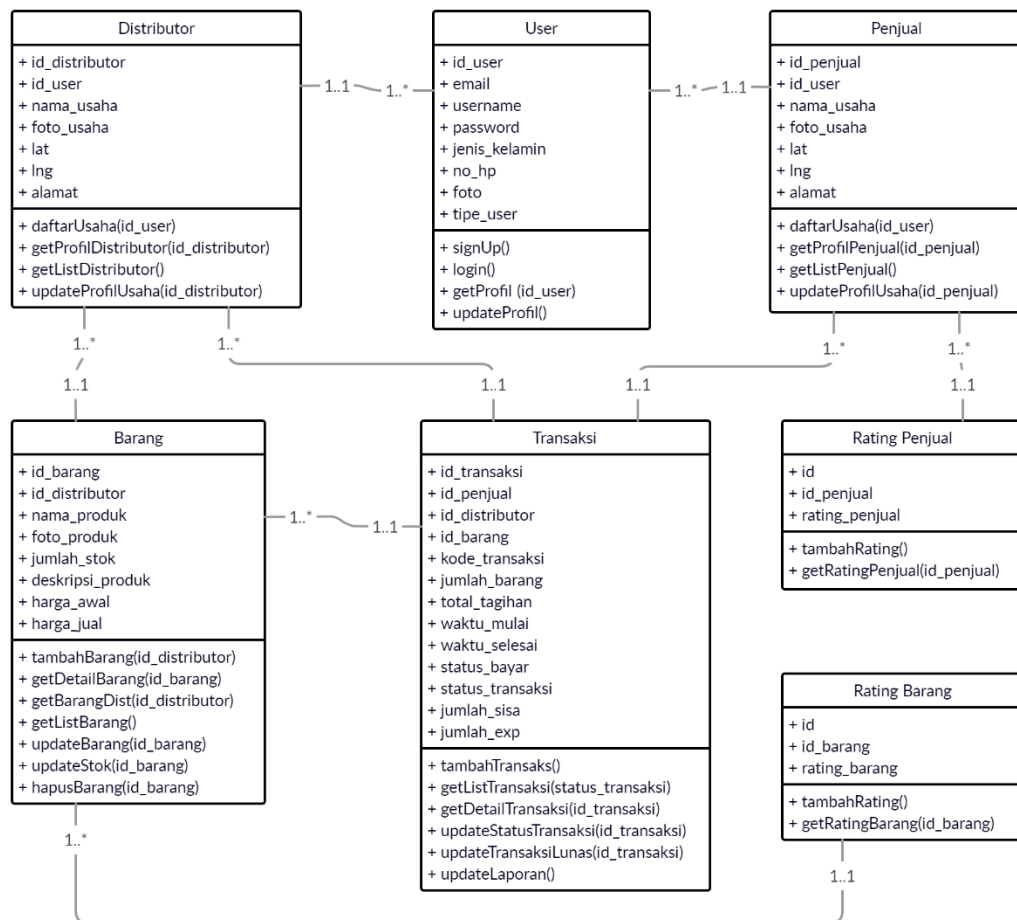
*Activity Diagram* Melihat Transaksi dapat dilakukan oleh distributor maupun penjual. Dimulai dengan pengguna memilih *tab* transaksi, kemudian aplikasi akan menampilkan transaksi yang sedang berlangsung hari ini. Apabila pengguna memilih transaksi yang ingin dilihat, maka aplikasi akan menampilkan transaksi yang dipilih. *Activity Diagram* Melihat Transaksi disajikan pada Gambar 9.



Gambar 9. *Activity Diagram* Melihat Transaksi

### c. *Class Diagram*

*Class diagram* adalah diagram yang menggambarkan struktur dan deskripsi *class*, *package*, dan objek beserta hubungannya satu sama lain. *Class diagram* pada aplikasi manajemen penjualan barang konsinyasi disajikan pada Gambar 10.



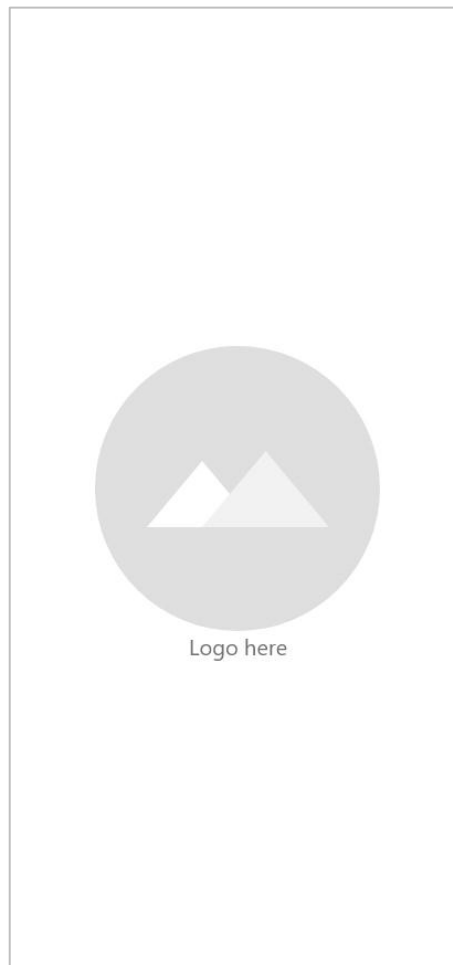
Gambar 10. *Class Diagram* Aplikasi Manajemen Barang Konsinyasi

#### d. *Spike Solutions Prototype*

*Spike Solutions Prototype* merupakan proses yang dibuat untuk menggambarkan sebuah tampilan (*interface*) dari aplikasi yang akan dibentuk. Aplikasi manajemen penjualan barang konsinyasi berbasis Android dirancang dengan tampilan yang *user-friendly* sehingga diharapkan dapat mempermudah pengguna dalam menggunakan aplikasi ini. Aplikasi manajemen penjualan barang konsinyasi dibuat dengan beberapa *layout* sebagai berikut.

### 1. *Layout Splash Screen*

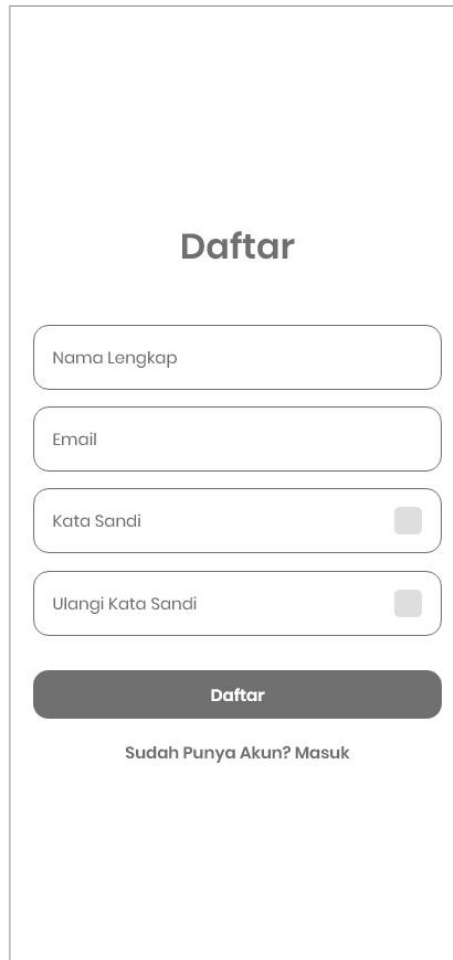
*Layout Splash Screen* adalah tampilan awal yang akan muncul ketika pengguna menjalankan aplikasi. *Splash Screen* digunakan sebagai identitas aplikasi. Aplikasi versi Distributor dan versi Penjual memiliki bentuk *Design Interface Layout Splash Screen* yang sama. *Design Interface* dari *Layout Splash Screen* disajikan pada Gambar 11.



Gambar 11. *Layout Splash Screen*

## 2. *Layout Sign Up*

*Layout Sign Up* merupakan tampilan dari halaman tempat pengguna bisa mendaftarkan diri sebagai pengguna aplikasi. Aplikasi versi Distributor dan versi Penjual memiliki bentuk *Design Interface Layout Sign Up* yang sama. *Layout Sign Up* disajikan pada Gambar 12.



The image shows a registration form with the following elements:

- Daftar**: The title of the form, centered at the top.
- Nama Lengkap**: A text input field for the user's full name.
- Email**: A text input field for the user's email address.
- Kata Sandi**: A text input field for the password, with a small square icon on the right side.
- Ulangi Kata Sandi**: A text input field for the password confirmation, with a small square icon on the right side.
- Daftar**: A dark gray button with white text, centered below the input fields.
- Sudah Punya Akun? Masuk**: A link for users who already have an account, centered below the button.

Gambar 12. *Layout Sign Up*

### 3. *Layout Login*

Setelah mendaftarkan diri di bagian *Sign Up*, pengguna diharuskan untuk menginputkan *username* dan *password* yang sudah dibuat untuk kemudian diverifikasi bahwa pengguna yang bersangkutan benar sudah memiliki akun untuk menggunakan aplikasi. Aplikasi versi Distributor dan versi Penjual memiliki bentuk *Design Interface* yang sama. *Layout Login* disajikan pada Gambar 13.



Ilustrasi

Kontrol Barang dagangmu sekarang bisa lebih mudah!

## Masuk

Alamat Surel

Kata Sandi

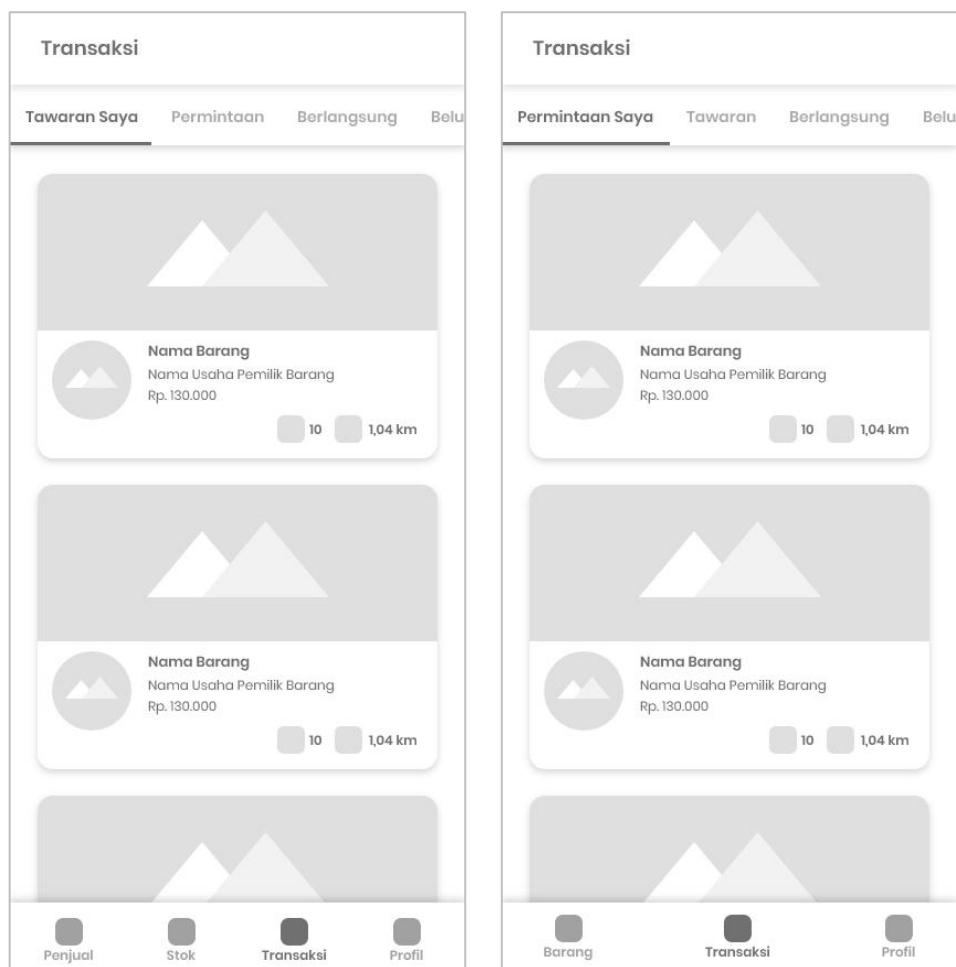
Masuk

Belum Punya Akun? Buat Akun

Gambar 13. *Layout Log In*

#### 4. *Layout Tab Transaksi*

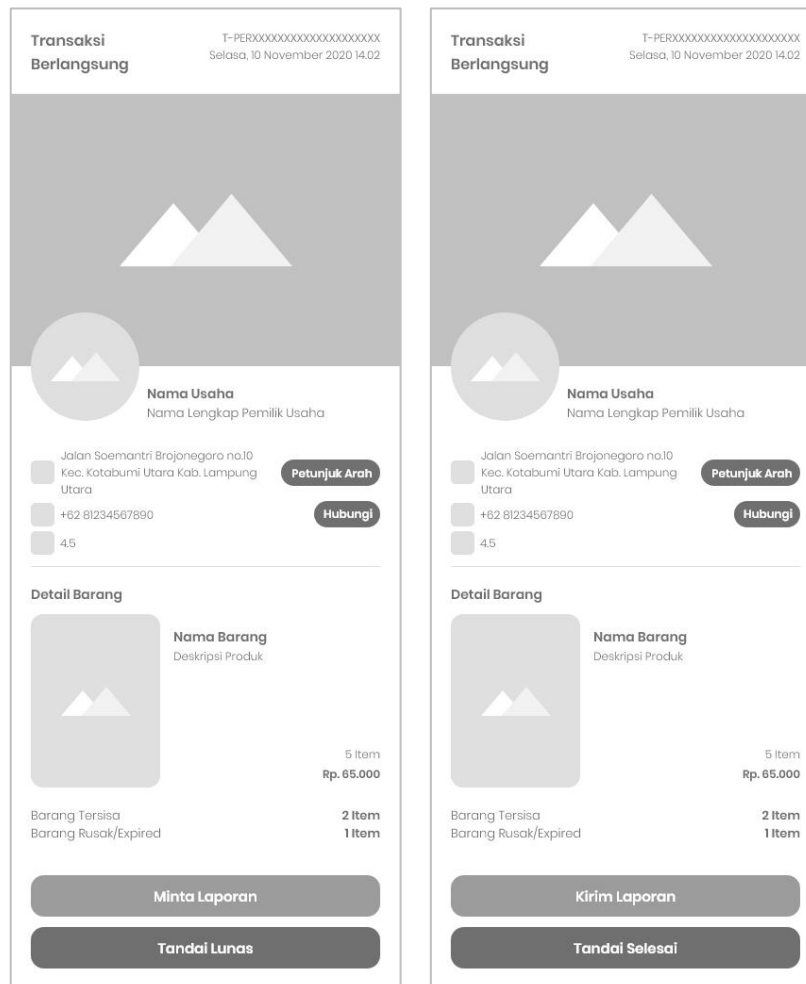
*Tab* Transaksi terbagi menjadi 4 bagian. Bagian ‘Tawaran’ untuk melihat kembali tawaran penjualan barang yang diberikan kepada penjual tertentu, ‘Permintaan’ untuk melihat permintaan barang dari penjual disekitar, ‘Berlangsung’ untuk melihat transaksi yang sedang berlangsung, dan ‘Belum dibayar’ untuk melihat transaksi yang belum dibayar. *Layout Tab* Transaksi disajikan pada Gambar 14.



Gambar 14. *Layout Tab* Transaksi

## 5. *Layout* Detail Transaksi

*Layout* Detail Transaksi adalah tampilan yang akan muncul ketika pengguna menekan salah satu dari daftar transaksi. Dari 4 jenis transaksi yang ada, perbedaan terdapat pada tombol dibawah detail. Jika dalam Permintaan/Tawaran, tombol menampilkan fungsi Terima/Tolak. Dalam Transaksi Berlangsung, menampilkan Minta Laporan dan Kirim Laporan. Proses penyelesaian transaksi berlangsung dua Arah dengan Distributor mengirimkan kode pelunasan dan Penjual memverifikasi kode. *Layout* Detail Transaksi disajikan pada Gambar 15.



Gambar 15. *Layout* Detail Transaksi

## 6. *Layout Form* Laporan Transaksi

*Layout Form* Laporan Transaksi merupakan bagian dari Aplikasi versi Penjual. *Form* Laporan Transaksi dikemas dalam bentuk *Alert Dialog* yang muncul setelah Penjual menekan tombol ‘Kirim Laporan’ yang ada pada laman Detail Transaksi Berlangsung. Kemudian Aplikasi akan menampilkan *Alert Dialog* yang berisi form Buat Laporan beserta jumlah barang yang ingin dilaporkan. *Layout Form* Laporan Transaksi disajikan pada Gambar 16.

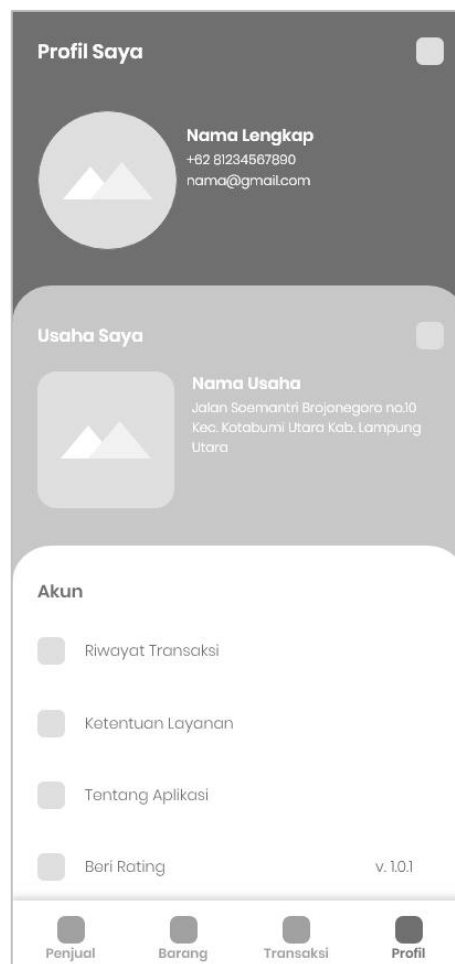
The screenshot shows a mobile application interface. At the top, there is a header for 'Transaksi Berlangsung' with a transaction ID 'T-PERXXXXXXXXXXXXXXXXXXXX' and a timestamp 'Selasa, 10 November 2020 14.02'. Below this, a modal dialog titled 'Buat Laporan' is displayed. The dialog contains a dropdown menu for 'Jenis Laporan', a text input field for 'Jumlah Barang', and a 'Kirim Laporan' button. Below the dialog, the background shows 'Detail barang' with a product image, 'Nama Barang', 'Deskripsi Produk', '5 Item', 'Rp. 65.000', and 'Barang Tersisa 2 Item'.

Gambar 16. *Layout Form* Laporan Transaksi



## 7. *Layout* Profil

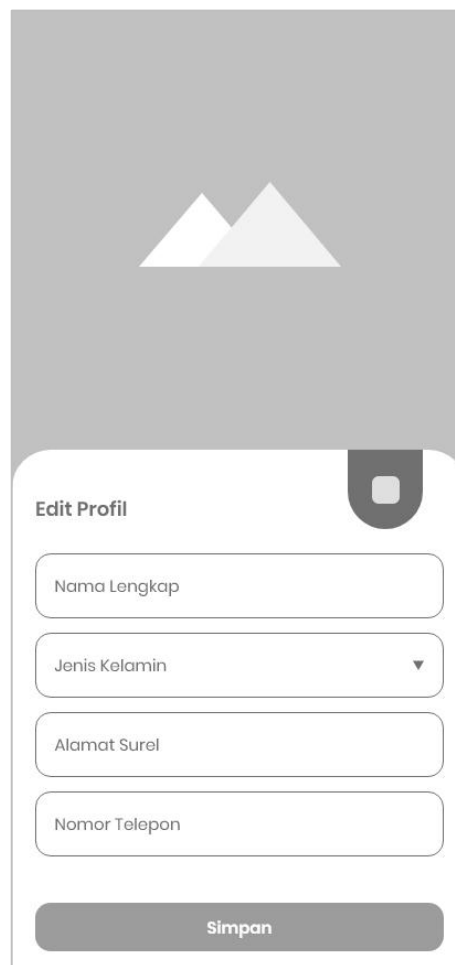
*Layout* Profil adalah tampilan yang berisi informasi pribadi Pengguna. Terdapat tampilan Profil Pribadi Pengguna dan juga Pilihan untuk mengubahnya, Profil Usaha Pengguna dan juga pilihan untuk mengubahnya, dan beberapa informasi terkait Aplikasi termasuk juga Riwayat Transaksi. Aplikasi versi Distributor dan versi Penjual memiliki bentuk *Design Interface Layout* Profil yang sama. *Layout* Profil disajikan pada Gambar 17.



Gambar 17. *Layout* Profil

## 8. *Layout* Edit Profil

*Layout* Edit Profil adalah laman yang menyajikan Profil Pribadi pengguna baik itu Penjual ataupun Distributor. Berisi Form Identitas Pribadi seperti Foto, Nama, Email, Jenis Kelamin, dan Nomor Telepon beserta tombol Simpan jika telah selesai mengubah Form Edit Profil. *Layout* edit Profil disajikan pada Gambar 18.

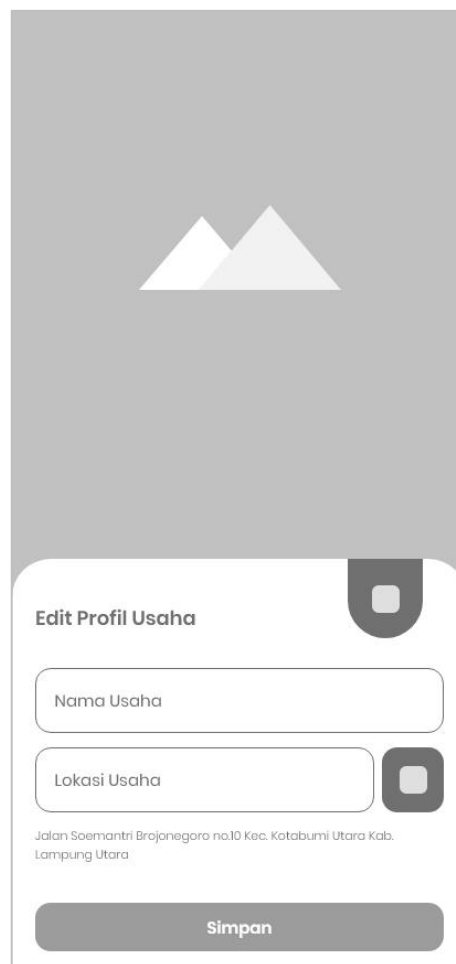


The image shows a mobile application interface for editing a user profile. At the top, there is a grey header area with a white mountain-like icon. Below this is a white rounded rectangle containing the form. The form is titled "Edit Profil" in the top left corner. It consists of four input fields stacked vertically: "Nama Lengkap", "Jenis Kelamin" (with a dropdown arrow), "Alamat Surel", and "Nomor Telepon". At the bottom of the form is a dark grey button labeled "Simpan".

Gambar 18. *Layout* Edit Profil

## 9. *Layout* Edit Profil Usaha

*Layout* Edit Profil Usaha adalah laman untuk mendaftarkan tempat usaha dari Distributor atau Penjual. Berisi foto warung/usaha, Nama warung/usaha, beserta lokasi warung/usaha. Untuk menandai lokasi, pengguna dapat mengetuk tombol bergambar lokasi di samping kolom lokasi. Setelah pengguna menyimpan lokasi, aplikasi akan menampilkan titik *latitude* dan *longitude* beserta alamat lokasi. *Layout* Edit Profil Usaha disajikan pada Gambar 19.

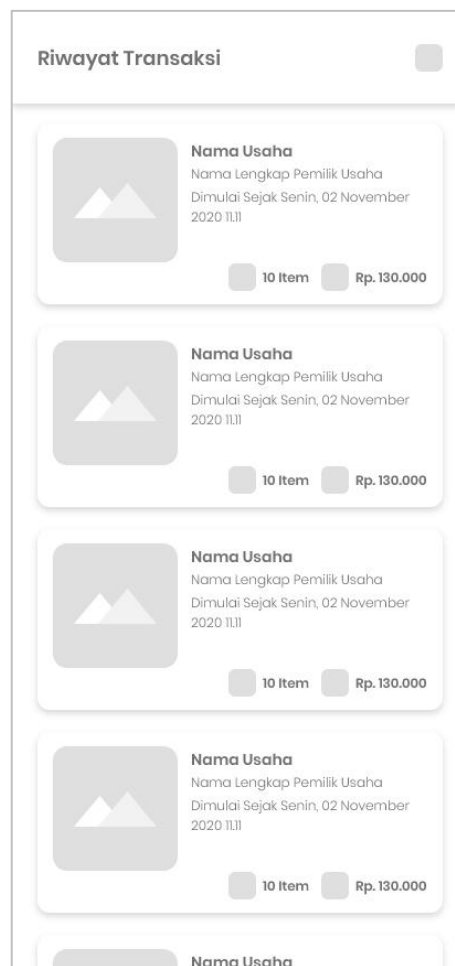


The image shows a mobile application interface for editing a business profile. At the top, there is a placeholder for a business photo, represented by a simple mountain icon. Below this, the title "Edit Profil Usaha" is displayed. The form contains two main input fields: "Nama Usaha" and "Lokasi Usaha". The "Lokasi Usaha" field includes a location pin icon to the right. Below the location field, the current location address is shown: "Jalan Soemantri Brajanegoro no.10 Kec. Kotabumi Utara Kab. Lampung Utara". At the bottom of the form is a prominent "Simpan" (Save) button.

Gambar 19. *Layout* Edit Profil Usaha

## 10. *Layout* Riwayat Transaksi

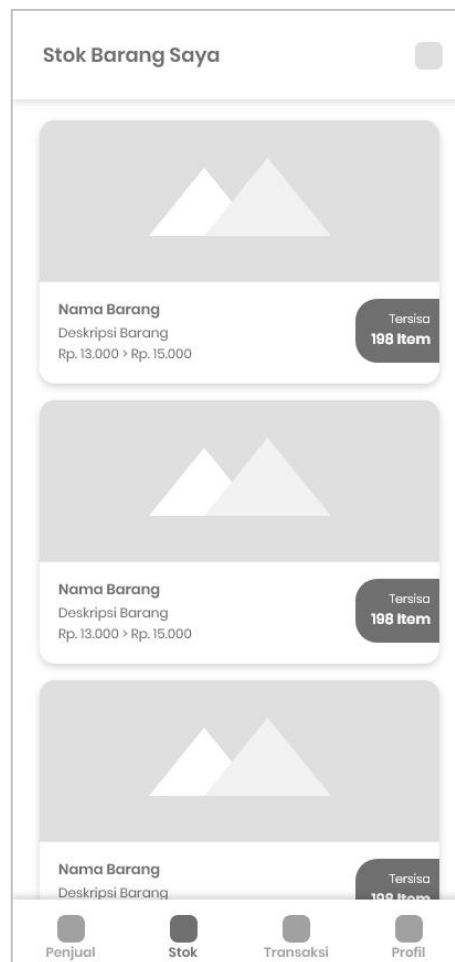
*Layout* Riwayat Transaksi adalah daftar rinci transaksi yang pernah terjadi melalui aplikasi. Aplikasi akan menampilkan data berdasarkan urutan waktu dimulainya transaksi. Pengguna juga dapat merurut tampilan urutan daftar riwayat transaksi dengan menekan tombol Urutkan di pojok kanan atas layar. Aplikasi versi Distributor dan versi Penjual memiliki bentuk *Design Interface Layout* Riwayat Transaksi yang sama. *Layout* Riwayat Transaksi disajikan pada Gambar 20.



Gambar 20. *Layout* Riwayat Transaksi

## 11. *Layout Tab Stok*

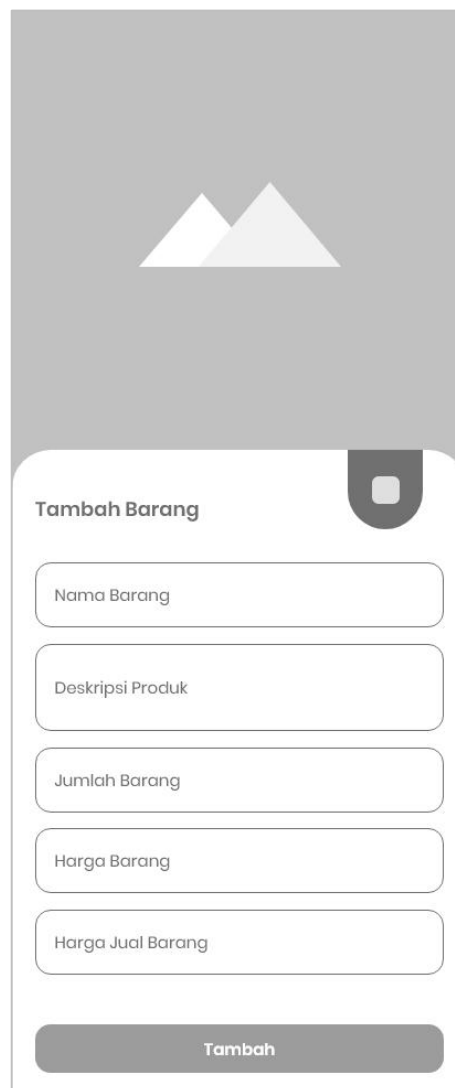
*Tab Stok* merupakan salah satu bagian dari Aplikasi versi Distributor. *Tab* ini menampilkan daftar barang yang dimiliki oleh Distributor. Di dalam *tab* ini, Distributor dapat menambah, merubah, atau menghapus barang. Untuk menambah barang, Distributor dapat menekan tombol tambah di pojok kanan atas layar. Sementara untuk merubah dan menghapus barang, Distributor dapat menekan daftar barang yang diubah atau dihapus. *Layout Tab Stok* disajikan pada Gambar 21.



Gambar 21. *Layout Tab Stok*

## 12. *Layout* Tambah Barang

*Layout* Tambah Barang adalah laman yang akan muncul jika distributor menekan tombol tambah di ujung kanan laman *Tab* Stok. *Layout* Tambah Barang berisi form identitas barang seperti Nama Barang, Deskripsi, Jumlah Stok yang tersedia, dan juga Harga Barang. *Layout* Tambah Barang disajikan pada Gambar 22.

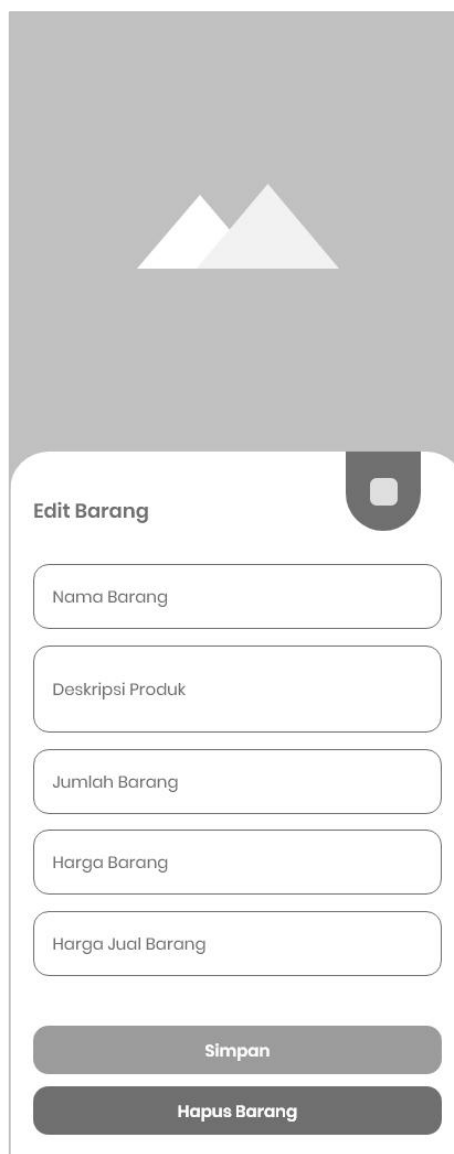


The image shows a mobile application interface for adding a new item. The form is titled "Tambah Barang" and is set against a light gray background. It features five input fields stacked vertically, each with a placeholder text: "Nama Barang", "Deskripsi Produk", "Jumlah Barang", "Harga Barang", and "Harga Jual Barang". At the bottom of the form is a dark gray button labeled "Tambah". The form is framed by a white border and is positioned below a header area with a gray background and a white mountain-like icon.

Gambar 22. *Layout* Tambah Barang

### 13. *Layout* Edit Barang

*Layout* Edit Barang adalah laman yang akan tampil jika pengguna memilih salah satu barang di *Tab* Stok. Laman ini digunakan untuk mengubah detail barang. Distributor dapat merubah stok yang tersedia, menambahkan diskon di Deskripsi Produk, Mengubah Harga, dan Menghapus barang tertentu apabila sudah tidak dijual lagi. *Layout* Edit Barang disajikan pada Gambar 23.

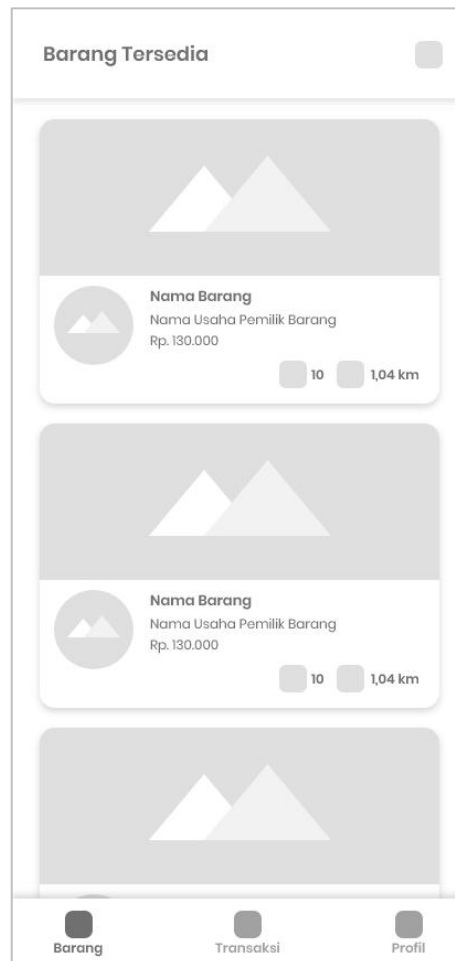


The image shows a mobile application interface for editing a product. At the top, there is a grey header with a white mountain icon. Below the header is a white rounded rectangle containing the title "Edit Barang" and a back arrow icon. The form consists of five input fields: "Nama Barang", "Deskripsi Produk", "Jumlah Barang", "Harga Barang", and "Harga Jual Barang". At the bottom, there are two buttons: a grey "Simpan" button and a dark grey "Hapus Barang" button.

Gambar 23. *Layout* Edit Barang

#### 14. *Layout Tab Barang*

*Tab* Barang terdapat pada Aplikasi versi Penjual. *Tab* ini menampilkan daftar barang terdekat yang bisa dijual. Penjual dapat memilih barang yang berpotensi untuk dijual. *Layout* *Tab* Barang disajikan pada Gambar 24.

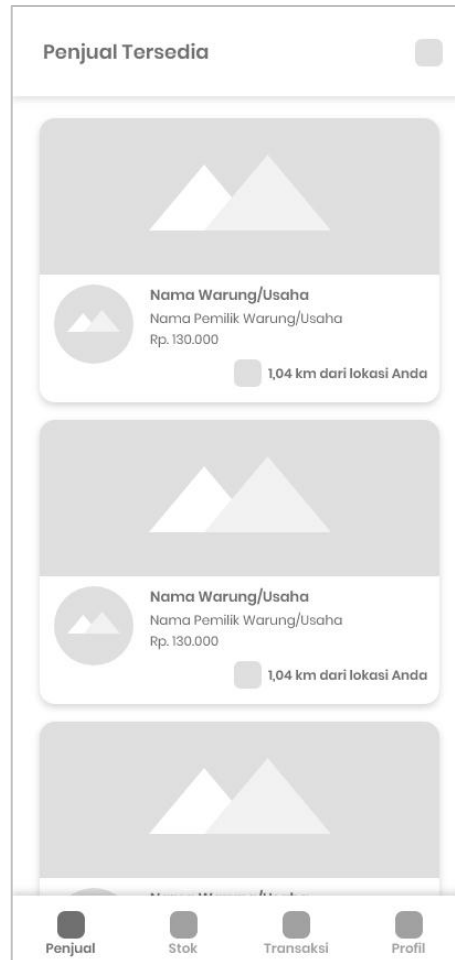


Gambar 24. *Layout Tab* Barang



### 15. *Layout Tab Penjual*

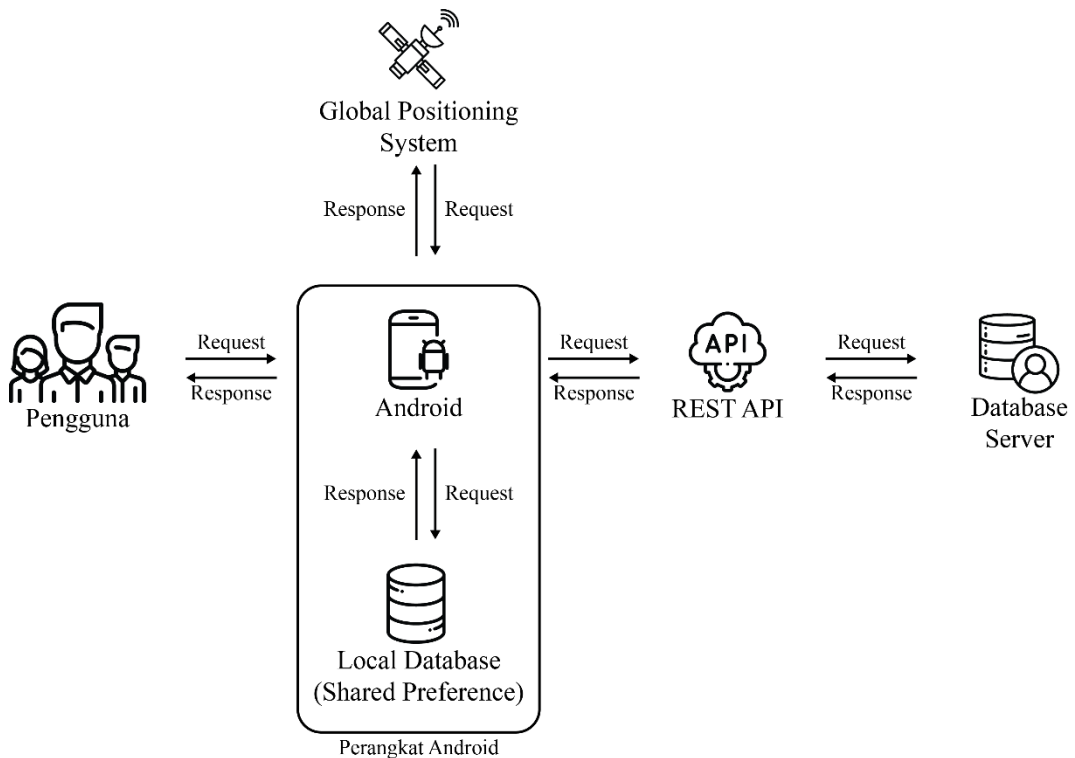
*Tab Penjual* menyajikan daftar penjual yang bersedia menjual barang yang dimiliki untuk dijual. *Tab* ini berada pada aplikasi versi Distributor. Distributor dapat memilih penjual yang akan menjualkan barangnya. *Layout Tab Penjual* disajikan pada Gambar 25.



Gambar 25. *Layout Tab Penjual*

### 3.3.3 Coding

*Coding* adalah salah satu tindakan dari langkah-langkah pemrograman dengan menuliskan kode atau skrip dalam bahasa pemrograman. Sebelum melakukan *coding*, diperlukan pemahaman bentuk arsitektur aplikasi yang dibutuhkan agar kode yang dibuat dapat sesuai dengan kebutuhan rancang bangun aplikasi. Adapun arsitektur yang telah dirancang dalam penelitian ini disajikan pada gambar 26.



Gambar 26. Arsitektur aplikasi

Berdasarkan arsitektur aplikasi yang telah dirancang, pengembangan aplikasi manajemen penjualan barang konsinyasi berbasis Android membutuhkan dua aplikasi yang membutuhkan kode program yaitu bagian aplikasi Android yang terhubung langsung ke pengguna dan REST API yang menghubungkan aplikasi

Android dengan tempat penyimpanan data daring. Penyimpanan data daring digunakan untuk proses transaksi data antara aplikasi versi Distributor dan aplikasi versi Penjual yang merupakan dua aplikasi yang berbeda.

Pada aplikasi Android digunakan Bahasa pemrograman *Kotlin* yang merupakan bahasa yang didukung penuh oleh pengembang Android. Sementara pada aplikasi REST API digunakan bahasa pemrograman PHP. Pengembangan aplikasi manajemen penjualan barang dengan sistem konsinyasi dikembangkan berdasarkan *user stories* dan desain yang telah dibuat menggunakan permodelan UML.

### **3.3.4 Testing**

*Testing* yang dilakukan dalam pengembangan aplikasi manajemen penjualan barang dengan sistem konsinyasi adalah dengan menggunakan metode *Black-Box Testing*. Metode ini merupakan teknik pengujian perangkat lunak yang berfokus pada domain informasi dari perangkat lunak tersebut. Dengan kata lain *Black-Box Testing* adalah metode pengujian yang menguji fungsionalitas sistem.

Pengujian secara *Black-Box Testing* dilakukan untuk memastikan semua fungsionalitas dari aplikasi berjalan dengan baik. Apabila masih terdapat kesalahan pada aplikasi, maka aplikasi akan diperbaiki untuk masuk kembali ke dalam proses pengkodean. Rancangan pengujian aplikasi manajemen penjualan barang dengan sistem konsinyasi terdiri dari pengujian fungsional. Pengujian

fungsional dibagi menjadi beberapa kelas uji seperti pengujian versi Android, ukuran layar, dan pengujian fitur utama yang disajikan pada tabel berikut.

Tabel 5. Rancangan Daftar Pengujian Fungsional Aplikasi

No	Kelas Uji	Daftar Pengujian	Skenario Uji	Hasil yang Diharapkan
1.	Versi Android	Pengujian kompatibilitas versi sistem operasi Android	Pengujian pada Android versi 5.1 ( <i>Lollipop</i> )	Kompatibel dengan Android versi 5.1 ( <i>Lollipop</i> )
			Pengujian pada Android versi 6.0 ( <i>Marshmallow</i> )	Kompatibel dengan Android versi 6.0 ( <i>Marshmallow</i> )
			Pengujian pada Android versi 7.0 ( <i>Nougat</i> )	Kompatibel dengan Android versi 7.0 ( <i>Nougat</i> )
			Pengujian pada Android versi 8.0 ( <i>Oreo</i> )	Kompatibel dengan Android versi 8.0 ( <i>Oreo</i> )
			Pengujian pada Android versi 9.0 ( <i>Pie</i> )	Kompatibel dengan Android versi 9.0 ( <i>Pie</i> )
			Pengujian pada Android versi 10	Kompatibel dengan Android versi 10
		Pengujian ukuran layar Android	Pengujian pada Android dengan layar berukuran 5.5"	Tampilan terlihat baik dan responsif pada layar berukuran 5.5"
			Pengujian pada Android dengan layar berukuran	Tampilan terlihat baik dan responsif pada layar berukuran 6"

			6"	
--	--	--	----	--

Tabel 5. (lanjutan)

No	Kelas Uji	Daftar Pengujian	Skenario Uji	Hasil yang Diharapkan
3.	<i>User Interface</i> Aplikasi versi Penjual	Pengujian pada <i>Tab</i> Barang	Pengujian fitur Cari Barang	Fitur dapat berjalan dengan normal.
			Pengujian fitur Menampilkan Katalog Barang	Fitur dapat berjalan dengan normal.
			Pengujian fitur Buat Transaksi	Fitur dapat berjalan dengan normal.
		Pengujian pada <i>Tab</i> Transaksi	Pengujian fitur menampilkan daftar transaksi	Fitur dapat berjalan dengan normal.
			Pengujian fitur Menolak/Menerima Tawaran	Fitur dapat berjalan dengan normal.
			Pengujian fitur Selesaikan Transaksi	Fitur dapat berjalan dengan normal.
			Pengujian fitur Kirim Laporan	Fitur dapat berjalan dengan normal.
		Pengujian pada <i>Tab</i> Profil	Pengujian fitur Edit Profil	Fitur dapat berjalan dengan normal.
			Pengujian fitur Edit Profil Usaha	Fitur dapat berjalan dengan normal.
			Pengujian fitur menampilkan riwayat transaksi	Fitur dapat berjalan dengan normal.

Tabel 5. (lanjutan)

No	Kelas Uji	Daftar Pengujian	Skenario Uji	Hasil yang Diharapkan
4.	User Interface Aplikasi versi Distributor	Pengujian pada <i>Tab</i> Penjual	Pengujian fitur Cari Penjual	Fitur dapat berjalan dengan normal.
			Pengujian fitur Menampilkan Daftar Penjual	Fitur dapat berjalan dengan normal.
			Pengujian fitur Buat Transaksi	Fitur dapat berjalan dengan normal.
		Pengujian pada <i>Tab</i> Stok	Pengujian Tambah Stok	Fitur dapat berjalan dengan normal.
			Pengujian Edit Stok Barang	Fitur dapat berjalan dengan normal.
		Pengujian pada <i>Tab</i> Transaksi	Pengujian fitur menampilkan daftar transaksi	Fitur dapat berjalan dengan normal.
			Pengujian fitur Menolak/Menerima Tawaran	Fitur dapat berjalan dengan normal.
			Pengujian fitur Pelunasan Transaksi	Fitur dapat berjalan dengan normal.
			Pengujian fitur Minta Laporan	Fitur dapat berjalan dengan normal.
		Pengujian pada <i>Tab</i> Profil	Pengujian fitur Edit Profil	Fitur dapat berjalan dengan normal.
			Pengujian fitur Edit Usaha	Fitur dapat berjalan dengan normal.
			Pengujian fitur riwayat transaksi	Fitur dapat berjalan dengan normal.

### **3.3.5 Release**

*Release* merupakan penyerahan hasil aplikasi kepada pengguna, baik itu Penjual maupun Distributor. *Release* dilakukan apabila semua permintaan dari *user stories* dan pengujian telah berhasil dilakukan.



## V. KESIMPULAN DAN SARAN

### 5.1 Kesimpulan

Adapun kesimpulan yang diperoleh dari penelitian yang telah dilakukan adalah telah berhasil dibangun Aplikasi Manajemen Penjualan Barang dengan Sistem Konsinyasi untuk membantu manajerial Distributor maupun Penjual dalam Penjualan Barang dengan Sistem Konsinyasi berbasis Android.

### 5.2 Saran

Adapun saran yang diperoleh dari penelitian yang telah dilakukan adalah sebagai berikut:

1. Aplikasi dapat dikembangkan dengan menambahkan fitur *cart* atau keranjang belanja agar Penjual dan Distributor dapat menawarkan atau meminta lebih dari satu barang dalam satu transaksi.
2. Aplikasi dapat dikembangkan dengan menambahkan fitur navigasi dalam peta untuk mempermudah proses penemuan lokasi di dalam aplikasi.
3. Aplikasi dapat dikembangkan dengan menambahkan fitur *chat* atau obrolan yang langsung *build-in* dalam aplikasi agar tidak perlu lagi menggunakan bantuan aplikasi lain.

4. Aplikasi dapat dikembangkan dengan menambahkan fitur Lupa Password agar pengguna dapat langsung terhubung kembali apabila terdapat masalah saat *login*.
5. Aplikasi dapat dikembangkan dengan menambahkan fitur Verifikasi Data Penjual dan Distributor agar mengurangi risiko kerugian usaha akibat dari informasi palsu yang belum terverifikasi.

## DAFTAR PUSTAKA

- Aristoteles, Nur Efendi, Febi Eka Febriansyah, Wisnu Lukito, & Firmansyah. 2018. "Sistem Informasi Kuliah Kerja Nyata (KKN) Berbasis Android Universitas Lampung." *Jurnal Komputasi* 6(2):1–10.
- Connolly, Thomas M., & Carolyn E. Begg. 2010. *Database Systems : A Practical Approach to Design, Implementation, and Management*. London: Addison-Wesley.
- Google Developers. 2020. "Save Key-Value Data | Android Developers." Retrieved December 20, 2020 (<https://developer.android.com/training/data-storage/shared-preferences>).
- Henderi. 2009. *Object Oriented Modelling With Unified Modeling Language (Uml)*. Tangerang.
- Indriasari, Sofiyanti. 2012. "Sistem Informasi Berbasis Web Untuk Membantu Kegiatan Tracer Study Program Diploma Institut Pertanian Bogor." *Jurnal Sains Terapan Edisi II* 2:48–58.
- Islam, Rafiqul, & Tapas Mazumder. 2010. "Mobile Application and Its Global Impact." *International Journal of Engineering & Technology IJET-IJENS* 10(06):72–78.
- Niode, Indris Yanto. 2009. "Sektor UMKM Di Indonesia : Profil, Masalah Dan Strategi Pemberdayaan." *Jurnal Kajian Ekonomi Dan Bisnis OIKOS-NOMOS* 2:1–10.

- Nugroho, Adi. 2009. *Rekayasa Perangkat Lunak Menggunakan UML Dan Java*. Yogyakarta: Penerbit Andi.
- Nurmayanti, Aristoteles, & Astria Hijriani. 2017. “Aplikasi Informasi Dokter Spesialis Di Bandar Lampung Berbasis Android Dengan Menggunakan Teknologi Location Base Service.” *Jurnal Komputasi* 5(2):55–63.
- Paramarta, I. Komang Adi. 2013. “Sistem Informasi Geografis Rencana Tata Ruang Wilayah (RTRW) Kabupaten Buleleng Berbasis Web.” *Jurnal Nasional Pendidikan Teknik Informatika (JANAPATI)* 2(3):243. doi: 10.23887/janapati.v2i3.9812.
- Partomo, Tiktik Sartika., & Abd. Rachman. Soejoedono. 2002. *Ekonomi skala kecil/menengah dan koperasi*. Jakarta: Ghalia Indonesia.
- Pranata, Beni Adi, Astria Hijriani, & Akmal Junaidi. 2018. “Perancangan Application Programming Interface (Api) Berbasis Web Menggunakan Gaya Arsitektur Representational State Transfer (Rest) Untuk Pengembangan Sistem Informasi Administrasi Pasien Klinik Perawatan Kulit.” *Jurnal Komputasi* 6(1):33–42. doi: 10.23960/komputasi.v6i1.1554.
- Pressman, Roger S., & Bruce R. Maxim. 2014. *Software Engineering : A Practitioner’s Approach*. New York: McGraw-Hill Education.
- Sabeni, Arifin. 1999. *Pokok-Pokok Akuntansi Lanjutan*. Yogyakarta: Liberty.
- Simamora, Maria. 2008. *Analisa Penerapan Akuntansi Konsinyasi*. Medan: USU e-Repository.
- Sumarsan, Thomas. 2017. *Akuntansi Dasar Dan Aplikasi Dalam Bisnis*. Second. Jakarta Barat: PT. Indeks.

- Suryana, Dayat. 2018. *Android Studio: Belajar Android Studio*. Bandung: Dayat Suryana Independent.
- Suryantara, I. Gusti Ngurah. 2017. *Merancang Aplikasi Dengan Metodologi Extreme Programming*. Jakarta: Elex Media Komputindo.
- Triandini, Evi, & I. Gede Suardika. 2012. *Step by Step Desain Proyek Menggunakan UML*. Yogyakarta: Penerbit Andi.
- Turban, Efraim, Jon Outland, David King, Jae Kyu Lee, Ting-Peng Liang, & Deborah C. Turban. 2018. *Electronic Commerce 2018: A Managerial and Social Networks Perspective*.
- Wahana Komputer. 2010. *Panduan Belajar MySQL Database Server*. Jakarta: MediaKita.
- Widayat, Utoyo. 1994. *Akuntansi Keuangan Lanjutan Ikhtisar Teori Dan Soal*. Jakarta: Lembaga Penerbit Fakultas Ekonomi Universitas Indonesia.
- Yendrawati, Reni. 2005. *Akuntansi Keuangan Lanjutan 1*. 1st ed. Yogyakarta: Ekonisia.
- Yudhanto, Yudha, & Ardhi Wijayanto. 2018. *Mudah Membuat Dan Berbisnis Aplikasi Android Dengan Android Studio*. Jakarta: Elex Media Komputindo.