

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terkait

Sebagai bukti penelitian yang akan dibuat, maka penelitian akan dibandingkan terhadap penelitian sejenis yang pernah dilakukan. Adapun Penelitian sebelumnya yang penulis angkat yaitu :

Berdasarkan penelitian Purnama Sari (2018), dengan judul “Penerapan Anggaran Penjualan Dengan Metode Least Square Dalam Memperkirakan Pendapatan Pada Home Industri Batik Madura” Home industri batik madura UD. Rahman batik Klampar Proppo Kabupaten Pamekasan belum menerapkan penggunaan anggaran penjualan untuk memperkirakan pendapatan di masa yang akan datang. Penggunaan anggaran penjualan dalam memperkirakan pendapatan sangat dibutuhkan oleh perusahaan maupun home industri. Penerapan anggaran penjualan dapat membantu manajemen perusahaan atau home industri dalam meningkatkan pendapatan. Selain itu dengan adanya anggaran penjualan dapat mengatasi atau meminimalisir kerugian.

Berdasarkan penelitian Muhamad Nadhif Hasan (2017), dengan judul “Penerapan Metode Least Square Dalam Menentukan stock pulsa Konter Roses Cell” Tujuan untuk meningkatkan keuntungan perusahaan. Salah satu kecerdasan bisnis yang harus dimiliki manager perusahaan adalah dapat mengubah data penjualan yang sangat banyak menjadi suatu informasi yang memiliki nilai bisnis melalui laporan analistik yang akurat. Dalam hal ini peneliti akan menggunakan

data penjualan pulsa di Konter Roses Cell sebagai bahan analisa, yang mendasari penelitian ini adalah pemilik dari konter tersebut masih menggunakan pengalokasi dan stok secara manual. Penentuan jumlah alokasipun juga masih dengan cara perkiraan, sehingga membuat pembagian dana alokasi kurang tepat. Tidak jarang terjadi kelebihan stok disatu produk, namun pada produk lain mengalami kekurangan.

Berdasarkan penelitian Siti Imas Maskanah (2018), dengan judul “Penerapan Model Peramalan enjualan Velg Mobil Menggunakan Metode Least Square Dan Regresi Linier Sederhana” Dengan melakukan prediksi yang tepat maka akan berdampak pada ketepatan dalam pengambilan kebijakan untuk dimasa mendatang sangat berpengaruh pada prediksi jumlah penjualan yang akan datang memerlukan penerapan metode-metode, hal ini bertujuan agar bisa meminimalkan kesalahan peramalan yang merupakan suatu seni dari ilmu memprediksi s esuatu yang belum terjadi dengan tujuan untuk memperkirakan peristiwa yang akan datang dimasa depan dengan selau memerlukan data-data masa lalu. Sehingga dengan peramalan, maka kemungkinan terjadinya peristiwa yang tidak sesuai dengan tujuan yang diharapkan diikuti dengan kesiapan untuk mengatisipasinya. Dalam penerapan model peramalan penjualan velg mobil menggunakan metode least square dan regrsi linier sederhana.

Berdasarkan penelitian Fajar Rohman Hariri (2018), dengan judul “Metode Least Square Untuk Prediksi Penjualan Sari Kedelai Rosi” Penjualan di tiga tempat tersebut setiap harinya mengalami naik turun, akibatnya tidak jarang hasil produksi banyak yang tidak terjual sehingga mengakibatkan kerugian. Selain

itu juga mengakibatkan pengadaan bahan baku produksi yang tidak sesuai dengan hasil penjualannya. Selama ini Sari Kedelai Rosi dalam pengadaan bahan baku tidak dinamis setiap harinya atau tidak disesuaikan dengan perkiraan penjualan hari berikutnya karena belum menggunakan sistem prediksi. Oleh karena itu perlu adanya suatu sistem yang dapat memprediksi penjualan setiap harinya. Sistem prediksi yang dibuat menggunakan metode *least square*.

Berdasarkan penelitian Indra Laksana Noerwan (2017), dengan judul “Peranan Peramalan Penjualan Terhadap Bahan Baku: Studi Kasus Least Square PD Sinar Rejeki Ban Di Kabupaten Pandeglang” Tujuan dari penelitian ini adalah mengetahui metode peramalan penjualan yang bersifat kuantitatif terhadap bahan baku. Populasi pada penelitian ini adalah semua data penjualan PD Sinar Rejeki Ban di Kabupaten Pandeglang Banten dari bulan Juli 2011 sampai September 2011. Penelitian ini menggunakan purposive sampling dan pengumpulan data ini dilakukan dengan cara observasi. Sedangkan teknik analisis data menggunakan metode kuadrat terkecil (*least square*). Hasil penelitian menemukan bahwa terdapat pengaruh peramalan penjualan terhadap bahan baku.

II.2. Uraia Teoritis

II.2.1. Perancangan

Perancangan Sistem adalah salah satu tahap dari metodologi pengembangan sistem, dan merupakan salah satu bagian yang sangat penting di dalam itu. Tahap ini dilakukan oleh sistem designer yang melakukan interaksi dengan pengguna sistem, hasil desain dievaluasi oleh user dari sudut pandang

kepentingan pemakai untuk kemudian diimplementasikan kembali oleh sistem designer. Berikut ini adalah beberapa prinsip yang perlu dipahami dalam melakukan desain sebuah sistem informasi :

1. Proses desain merupakan langkah lanjutan dari analisis data. Jadi desain harus dapat ditelusuri sampai ketinggian analisis.
2. Desain sebuah sistem harus meminimalkan kesenjangan intelektual.
3. Desain harus mengungkap keseragaman dan integrasi antar sub sistem yang kuat.
4. Desain harus berorientasi ke kondisi sekarang dan masa depan.
5. Desain harus mempertimbangkan konsep penanganan kesalahan disaat pengoperasian sistem.
6. Desain harus dinilai kualitasnya pada saat desain dibuat.
7. Desain harus dikaji lebih lanjut sehingga dapat meminimalkan kesalahan-kesalahan konseptual. (Saipul Anwar : 2018)

II.2.2. Aplikasi

Pengertian aplikasi adalah suatu bagian dari perangkat lunak yang dapat digunakan untuk menyelesaikan masalah-masalah yang khusus yang dihadapi *user* dengan menggunakan kemampuan komputer. (Fergiawan Listianto, dkk : 2017)

Aplikasi adalah program siap pakai yang dapat digunakan untuk menjalankan perintah-perintah dari pengguna aplikasi tersebut dengan tujuan mendapatkan hasil yang lebih akurat sesuai dengan tujuan pembuatan aplikasi

tersebut, aplikasi mempunyai arti yaitu pemecahan masalah yang menggunakan salah satu teknik pemrosesan data aplikasi yang biasanya berpacu pada sebuah komputasi yang diinginkan atau diharapkan maupun pemrosesan data yang diharapkan. (Ari Setiaji, dkk : 2020).

II.2.3. Peramalan

Peramalan adalah proses menaksirkan/ memperkirakan sesuatu di masa yang akan datang yang berdasarkan pada data yang ada di masa lalu yang kemudian dianalisis secara ilmiah dengan memakai metode statistika dengan tujuan supaya memperbaiki peristiwa yang akan terjadi di waktu yang akan datang. Peramalan (*forecasting*) merupakan alat bantu yang penting dalam perencanaan yang efektif dan efisien khususnya dalam bidang ekonomi. Dalam organisasi modern mengetahui keadaan yang akan datang tidak saja penting untuk melihat yang baik atau buruk tetapi juga bertujuan untuk melakukan persiapan peramalan. Tujuan diadakannya peramalan atau *forecasting* adalah untuk meminimalisasi resiko serta faktor ketidakpastian. Dengan adanya hasil peramalan, diharapkan tindakan atau keputusan dari suatu perusahaan atau organisasi dapat memberi dampak lebih baik pada jangka yang akan datang (Yanuar Adi Kurniawan:2018).

II.2.4. Kuadrat Terkecil (*Least Square*)

Metode kuadrat terkecil, yang lebih dikenal dengan nama *Least –Squares Method*, adalah salah satu metode ‘pendekatan’ yang paling penting dalam dunia

keteknikan untuk: (a) regresi ataupun pembentukan persamaan dari titik – titik data diskretnya (dalam pemodelan), dan (b). analisis sesatan pengukuran (dalam validasi model). (Kurniawan:2018)

Menganggap bahwa hubungan antara biaya dengan perubahan volume kegiatan berbentuk hubungan garis lurus dengan persamaan garis regresi $\underline{Y = a + b X}$. Rumus perhitungan a dan b tersebut adalah sebagai berikut:

$$b = \frac{n \cdot \sum XY - \sum X \cdot \sum Y}{n \cdot \sum X^2 - (\sum X)^2} \quad \text{Dimana : } a = \text{Biaya tetap} \quad \dots\dots\dots(1)$$

$$n \cdot \sum X^2 - (\sum X)^2 \quad b = \text{Biaya variabel}$$

y = Biaya sesungguhnya yang diamati

$$a = \frac{\sum Y - b \cdot \sum X}{n} \quad x = \text{Tingkat kegiatan} \quad \dots\dots\dots(2)$$

$$n \quad n = \text{Jumlah data}$$

Menurut Fajar Rohman Hariri (2018) Metode *Least Square* merupakan salah satu metode berupa data deret berkala atau time series, yangmana dibutuhkan data-data penjualan dimasa lampau untuk melakukan peramalan penjualan dimasa mendatang sehingga dapat ditentukan hasilnya. *Least Square* adalah metode peramalan yang digunakan untuk melihat *trend* dari data deret waktu . Persamaan 1 merupakan persamaan metode *Least Square*.

$$Y = a + bx(1) \quad \dots\dots\dots(3)$$

Keterangan:

Y : Jumlah Penjualan

a dan b : Koefisien

x / t : waktu tertentu dalam bentuk kode

Dalam menentukan nilai x / t seringkali digunakan teknik alternatif dengan memberikan skor atau kode. Dalam hal ini dilakukan pembagian data menjadi dua kelompok, yaitu:

a. Data genap, maka skor nilai t nya: ..., -5, -3, -1, 1, 3, 5, ...

b. Data ganjil, maka skor nilai t nya: ..., -3, -2, -1, 0, 1, 2, 3, ...

Kemudian untuk mengetahui koefisien a dan b dicari dengan persamaan 2 dan 3.

$$a = \frac{\sum Y}{n} \dots\dots\dots(4)$$

$$b = \frac{\sum tY}{\sum t^2} \dots\dots\dots(5)$$

Studi Kasus Data persediaan obat :

Berikut data persediaan obat berupa Novagesic selama Januari 2020 sampai dengan Desember 2020 adalah sebagai berikut :

Tabel II.1. Data Persediaan Obat

Bulan	Tahun	Persediaan (Unit)
Januari	2020	100
Februari	2020	123
Maret	2020	145
April	2020	160
Mei	2020	201
Juni	2020	239
Juli	2020	232
Agustus	2020	273
September	2020	137
Oktober	2020	140
November	2020	101
Desember	2020	142

Berikut ini data actual persediaan obat berupa Novagesic pada periode Januari 2021 :

Tabel II.2. Peramalan Persediaan Obat Berupa Novagesic

Bulan	Tahun	Persediaan Obat	X	XY	X ²
Januari	2020	100	-11	-1100	121
Februari	2020	123	-9	-1107	81
Maret	2020	145	-7	-1015	49
April	2020	160	-5	-800	25
Mei	2020	201	-3	-603	9
Juni	2020	239	-1	-239	1
Juli	2020	232	1	232	1
Agustus	2020	273	3	819	9
September	2020	137	5	685	25
Oktober	2020	140	7	980	49
November	2020	101	9	909	81
Desember	2020	142	11	1562	121
Jumlah		1993	0	323	572

Untuk mencari nilai a dan b adalah sebagai berikut :

$$a = \sum Y / n$$

$$a = 1993 / 12$$

$$= 166.08$$

$$b = \sum XY / \sum X^2$$

$$= 323 / 572$$

$$= 0.56$$

Persamaan garis liniernya adalah : $Y = a + bx$

$$= 166.08 + 0.56 X$$

Dengan menggunakan persamaan tersebut, dapat diramalkan persediaan obat berupa Novagesic pada Januari tahun 2021 adalah :

$$Y = 166.08 + 0.56 X \text{ (untuk tahun 2020 nilai X adalah 13),}$$

$$\text{sehingga : } Y = 166.08 + 0.56 (13)$$

$$= 166.08 + 7.28$$

$$= 173.36$$

$$= 173$$

Dengan demikian maka persediaan obat berupa Novagesic pada Januari 2021 adalah 173 unit dan lebih tinggi dibandingkan dengan persediaan obat berupa Novagesic pada Desember 2020.

II.2.5 Penjualan

Penjualan merupakan salah satu kegiatan yang dilakukan perusahaan untuk mempertahankan bisnisnya untuk berkembang dan untuk mendapatkan laba atau keuntungan yang diinginkan. Penjualan juga berarti proses kegiatan menjual, yaitu dari kegiatan penetapan harga jual sampai produk didistribusikan ke tangan konsumen(pembeli). Kegiatan penjualan merupakan kegiatan pelengkap atau suplemen dari pembelian, untuk memungkinkan terjadinya transaksi. Jadi kegiatan pembelian dan penjualan merupakan satu kesatuan untuk dapat terlaksananya transfer hak atau transaksi. Oleh karena itu, kegiatan penjualan seperti halnya kegiatan penjualan seperti halnya kegiatan pembelian, terdiri dari serangkaian kegiatan yang meliputi penciptaan permintaan, menemukan si pembeli, negosiasi harga, dan syarat-syarat pembayaran.dalam hal ini, penjualan ini, seperti penjual harus menentukan kebijaksanaan dan prosedur yang akan diikuti memungkinkan dilaksankannya rencana penjualan yang ditetapkan. (Harni : 2019)

II.2.7 Normalisasi

Normalisasi merupakan sebuah teknik dalam logical desain sebuah basis data, teknik pengelompokkan atribut dari suatu relasi sehingga membentuk struktur relasi yang baik tanpa redundansi. Tujuan normalisasi adalah mengorganisasikan data kedalam tabel-tabel untuk memenuhi kebutuhan pemakai, menghilangkan kerangkapan data, mengurangi kompleksitas, mempermudah modifikasi data. (Mukhlisulfatih Latief : 2018)

1. Proses Normalisasi

- a. Data diuraikan dalam bentuk tabel, selanjutnya dianalisis berdasarkan persyaratan tertentu kebeberapa tingkat.
- b. Apabila tabel yang diuji belum memenuhi persyaratan tertentu maka tabel tersebut perlu dipecah menjadi beberapa tabel yang lebih sederhana sampai memenuhi bentuk yang optimal.

2. Tahapan Normalisasi :

- 1) Bentuk tidak normal : Menghilangkan perulangan grup.

Tabel II.1. Contoh bentuk tidak normal (Unnormal)

No-Mhs	Nama Mhs	Jurusan	Kode-MK	Nama-MK	Kode Dosen	Nama Dosen	Nilai
2683	Welli	MI	M1350	Manajemen DB	B104	Ati	A
			M1465	Analisis Perc. Sistem	B317	Dita	B
5432	Bakti	AK	M1350	Manajemen DV	B104	Ati	C
			Akn201	Akuntansi	D310	Lia	B
			MKT300	Dasar Pemasaran	B212	Lola	A

Sumber : Mukhlisulfatih Latief : 2018

2) Bentuk Normal pertama (1NF) : Menghilangkan ketergantungan sebagian.

Yaitu : suatu relasi dikatakan sudah memenuhi bentuk normal kesatu bila setiap data bersifat atomik yaitu setiap irisan baris dan kolom hanya mempunyai satu nilai data.

Tabel II.2. Contoh Bentuk Normal Pertama (1NF)

No-Mhs	Nama Mhs	Jurusan	Kode-MK	Nama-MK	Kode Dosen	Nama Dosen	Nilai
2683	Welli	MI	M1350	Manajemen DB	B104	Ati	A
2683	Welli	MI	M1465	Analisis Perc. Sistem	B317	Dita	B
5432	Bakti	AK	M1350	Manajemen DV	B104	Ati	C
5432	Bakti	AK	Akn201	Akuntansi	D310	Lia	B
5432	Bakti	AK	MKT300	Dasar Pemasaran	B212	Lola	A

Sumber : Mukhlisulfatih Latief : 2018

3) Bentuk Normal kedua (2NF) : Menghilangkan ketergantungan transitif.

Yaitu : suatu relasi dikatakan sudah memenuhi bentuk normal kedua bila relasi tersebut sudah memenuhi bentuk normal kesatu dan atribut yang bukan key sudah tergantung penuh terhadap key-nya.

Tabel II.3. Contoh Bentuk Normal Kedua (2NF)

Kode-MK	Nama-MK	Kode Dosen	Nama Dosen
M1350	Manajemen DB	B104	Ati
M1465	Analisis Perc. Sistem	B317	Dita
M1350	Manajemen DV	B104	Ati
Akn201	Akuntansi	D310	Lia
MKT300	Dasar Pemasaran	B212	Lola

Sumber : Mukhlisulfatih Latief : 2018

4) Bentuk Normal ketiga (3NF) : Menghilangkan anomali-anomali hasil dari ketergantungan fungsional. Yaitu : suatu relasi dikatakan sudah memenuhi bentuk normal ketiga bila relasi tersebut sudah memenuhi bentuk normal kedua dan atribut yang bukan key tidak tergantung transitif terhadap *key*-nya. (Mukhlisulfatih Latief : 2018)

Tabel II.4. Contoh Tabel Mahasiswa Dan Tabel Kuliah (3NF)

No_Mhs	Nama Mhs	Jurusan
2683	Welli	MI
5432	Bakti	AK

Sumber : Mukhlisulfatih Latief : 2018

II.2.8. Basis Data (*Database*)

Basis data merupakan kumpulan dari data yang saling berhubungan antara satu dengan yang lain. Basis data atau *database* merupakan salah satu komponen yang penting dalam sistem informasi, karena berfungsi berfungsi sebagai basis penyedia informasi bagi pemakainya, Sistem basis data adalah suatu sistem informasi yang mengintegrasikan kumpulan dari data yang saling berhubungan dengan yang lainnya dan untuk membuatnya tersedia beberapa aplikasi yang bermacam-macam dalam suatu sistem organisasi. Sistem basis data adalah suatu sistem menyusun dan mengelola *record-record* menggunakan komputer untuk menyimpan atau merekam serta memelihara data operasional lengkap sebuah organisasi atau perusahaan sehingga mampu menyediakan informasi yang optimal yang diperlukan. (Priyo Sutopo,dkk 2018).

II.2.9. *Android*

Android adalah sistem operasi yang berbasis Linux untuk telepon seluler seperti telepon pintar dan komputer tablet. Android menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam peranti bergerak. Awalnya, Google Inc. membeli Android Inc., pendatang baru yang membuat peranti lunak untuk ponsel. Kemudian untuk mengembangkan Android, dibentuklah Open Handset Alliance, konsorsium dari 34 perusahaan peranti keras, peranti lunak, dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, TMobile, dan Nvidia. Pada saat perilisan perdana Android, 5 November 2007, Android bersama Open Handset Alliance menyatakan mendukung pengembangan standar terbuka pada perangkat seluler. Di lain pihak, Google merilis kode-kode Android di bawah lisensi Apache, sebuah lisensi perangkat lunak dan standar terbuka perangkat seluler (Harni Kusniyati Dan Dicky : 2018)

Android merupakan sistem operasi *mobile*. *Android* tidak membedakan antara aplikasi inti dengan aplikasi pihak ketiga. *Application Programming Interface* (API) yang disediakan menawarkan akses ke *hardware*, maupun data data ponsel sekalipun, atau data sistem sendiri. *Android* merupakan sebuah sistem operasi perangkat *mobile* berbasis *linux* yang mencakup sistem operasi, *middleware*, dan aplikasi. Beberapa pengertian lain dari *Android*, yaitu :

1. Merupakan *platform* terbuka (*Open Source*) bagi para pengembang (*Programmer*) untuk membuat aplikasi.
2. Merupakan sistem operasi yang dibeli *Google Inc.* dari *Android Inc.*

3. Bukan bahasa pemrograman, tetapi hanya menyediakan lingkungan hidup atau *run time enviroment* yang disebut DVM (*Dalvik Virtual Machine*) yang telah dioptimasi untuk alat/*device* dengan sistem memori yang kecil. (Ni Kadek Ceryna, dkk: 2018 : 101)

Android merupakan sistem operasi berbasis Linux yang dirancang untuk perangkat bergerak layar sentuh (*mobile*) seperti *smartphone* dan *tablet*. *Android* adalah sistem operasi yang bersifat *open source* dan *google* merilis kodenya di bawah lisensi *apache*. *Android* awalnya dikembangkan oleh *Android, Inc.*, dengan dukungan finansial dari *google* dan kemudian *google* membelinya pada tahun 2005. *Android* dibuat menggunakan bahasa pemrograman C(core), C++, Java(UI) dan dukungan berbagai bahasa pemrograman lain. Berdasarkan *dashboard developer android*, *Android 8.0/1 Oreo* adalah versi *Android* yang paling banyak digunakan, yakni sekitar 28.3% dari keseluruhan perangkat *Android* di seluruh dunia. Hingga saat ini, *Android* sistem operasi berbasis *mobile* yang paling banyak digunakan di seluruh dunia dan android sudah di rilis versi 10 pada 3 september 2019. (Roberto Kaban, dkk: 2019)

II.2.10. Andorid Studio

Android Studio adalah *Integrated Development Enviroment* (IDE) untuk sistem operasi *Android* yang dibangun berdasarkan perangkat lunak IntelliJ IDEA. Selain editor kode dan alat pengembang yang didesain khusus untuk pengembangan *android*, *Android studio* merupakan pengganti dari *Eclipse Android Development Tools* (ADT) yang sebelumnya merupakan IDE utama

untuk pengembangan aplikasi *android*. *Android studio* menyediakan banyak fitur untuk meningkatkan produktivitas dalam pengembangan aplikasi berbasis android seperti tersedianya sistem pembangunan berbasis *gradle* yang *fleksibel*, emulator yang memiliki banyak fitur, integrasi dengan GitHub, dukungan bawaan untuk *Google Cloud Platform* sehingga mempermudah pengembang aplikasi untuk mengintegrasikan *Google Cloud Messaging* dan *App Engine*. (Roberto Kaban, dkk: 2019)

II.2.11 Tools Pembangunan Android

Untuk membangun sebuah sistem operasi *Android* dapat menggunakan Mac, *Windows* PC, ataupun *Linux*. *Tools* yang dibutuhkan gratis dan dapat di *download* dari *web*. Berikut adalah beberapa *tools* yang digunakan untuk membangun aplikasi *android*.

1. JDK (*Java Development Kit*)
2. Android SDK
3. ADT (*Android Development Tools*) (Ni Kadek Ceryna, dkk : 2018 : 102)

II.2.12. Perkembangan Android

Adapun perkembangan Android adalah sebagai berikut :

a. Android versi 1.1

Android memang diluncurkan pertama kali pada tahun 2007, namun sistem operasi ini mulai dirilis dan diterapkan ke berbagai gadget pada tanggal 9 Maret 2009 silam. Android versi 1.1 merupakan Android awal yang dimana

versi ini baru memberikan sentuhan di beberapa aplikasinya seperti sistem antarmuka bagi pengguna (user interface) yang lebih baik, serta beberapa aplikasi yang lain.

b. Android versi 1.5 (Cupcake)

Pada bulan Mei 2009 Android kembali mengalami perubahan versi. Android versi 1.1 kemudian disempurnakan dengan Android versi 1.5 atau yang dikenal sebagai Android Cupcake.

c. Android versi 1.6 (Donut)

Donut (versi 1.6) diluncurkan dalam tempo kurang dari 4 bulan semenjak peluncuran perdana Android Cupcake, yaitu pada bulan September 2009.

d. Android versi 2.0/2.1 (Eclair)

Masih ditahun yang sama, Android kembali merilis operating sistem versi terbarunya, yaitu Android versi 2.0/2.1 Eclair. Android Eclair diluncurkan oleh Google 3 bulan setelah peluncuran.

e. Android versi 2.2 (Froyo: Frozen Yoghurt)

Butuh 5 bulan bagi Google untuk melakukan regenerasi dari Android Eclair versi sebelumnya ke versi Froyo Frozen Yoghurt. Pada tanggal 20 Mei 2010, Android versi 2.2 alias Android Froyo ini dirilis.

f. Android versi 2.3 (Gingerbread)

7 bulan kemudian Android kembali melakukan gebrakan dengan merilis kembali Android versi 2.3 atau yang dikenal sebagai Android Gingerbread.

g. Android versi 3.0/3.1 (Honeycomb)

Pada bulan Mei 2011 Android versi 3.0/3.1 atau Android Honeycom dirilis. Android Honeycomb merupakan sebuah sistem operasi Android yang tujuannya memang dikhususkan bagi penggunaan tablet berbasis Android.

h. Android versi 4.0 (ICS: Ice Cream Sandwich)

Android ICS atau Ice Cream Sandwich juga dirilis pada tahun yang sama dengan Honeycomb, yaitu pada bulan Oktober 2011.

i. Android versi 4.1 (Jelly Bean)

Android Jelly Bean merupakan versi Android yang terbaru pada saat ini. Salah satu gadget yang menggunakan sistem operasi Jelly Bean adalah Google Nexus 7 yang diprakarsai oleh ASUS, vendor asal Taiwan yang juga menjadi teman satu kampung halaman dengan Acer.

j. Android versi 4.4 (Kit Kat)

Kehadiran android kitkat merupakan peluncuran produk OS anyar yang diluncurkan pada 4 september 2013, sebelumnya banyak kabar beredar jikalau android akan meluncurkan OS baru yang bernama Android Key Lime Pie namun setelah di analisa tidak sesuai dengan ejaan orang umum, sehingga namanya diganti dengan OS Android KitKat yang sebagian besar orang sudah familiar dengan itu.

k. Android versi 5.0.2 (Lollipop)

Android Lollipop merupakan keberadaan OS Android yang memang saat ini sudah menjadi trend baru di industri smartphone, hal ini tak lepas dari keunikan dan kelebihan yang banyak di miliki dari OS tersebut. Kehadiran

android versi ini amat di nanti oleh sekian banyak orang karna diharapkan sistem operasi Lollipop ini bias lebih baik dibandingkan versi-versi sebelumnya.

l. Android versi 6.0 (Marshmallow)

Android 6.0 Marshmallow adalah versi dari sistem operasi mobile Android. Pertama kali diperkenalkan Mei 2015 di Google I / O di bawah kode nama Android M, secara resmi dirilis pada Oktober 2015. Android Marshmallow memperkenalkan model izin aplikasi didesain ulang sekarang ada hanya delapan kategori izin, dan aplikasi yang tidak lagi secara otomatis diberikan semua hak akses mereka ditentukan pada waktu instalasi. (Harni Kusniyati Dan Dicky : 2018)

II.2.13. MySql

MySQL (*My Structure Query Language*) merupakan sebuah program pembuat database yang bersifat *open source*, artinya semua orang dapat menggunakannya dan dapat dijalankan pada semua *platform* baik windows maupun linux. MySQL juga merupakan sebuah perangkat lunak sistem manajemen basis data SQL yang bersifat jaringan sehingga dapat digunakan untuk aplikasi multi user. MySQL juga sering dikenal dengan nama sistem manajemen database relasional. Suatu database relasional menyimpan data dalam table yang terpisah. Tabel – table tersebut terhubung oleh suatu relasi terdefinisi yang memungkinkan user memperoleh kombinasi data dari beberapa table dalam suatu permintaan. Untuk administrasi database, seperti

pembuatan database, pembuatan tabel, dan sebagainya dapat digunakan aplikasi berbasis web seperti phpMyAdmin dengan aplikasi XAMPP. (Saipul Anwar : 2018)

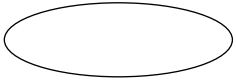
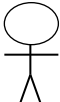
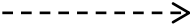
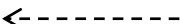
II.2.14 UML (*Unified Modelling Language*)

Hasil pemodelan pada OOAD terdokumentasikan dalam bentuk *Unified Modeling Language* (UML). UML adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan sistem. UML saat ini sangat banyak dipergunakan dalam dunia industri yang merupakan standar bahasa pemodelan umum dalam industry perangkat lunak dan pengembangan sistem. (Ade Handini : 2018)

1. *Diagram Use Case (Use Case Diagram)*

Use case diagram merupakan pemodelan untuk melakukan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Dapat dikatakan 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 digunakan dalam use case diagram, yaitu :

Tabel II.5. Simbol Use Case Diagram

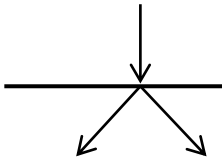
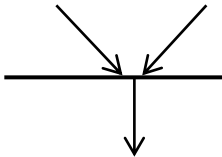
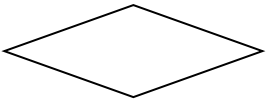
Gambar	Keterangan
	Use case menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal nama use case.
	Aktor adalah <i>abstraction</i> dari orang atau sistem yang lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasi aktor, harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem. Orang atau sistem bisa muncul dalam beberapa peran. Perlu dicatat bahwa aktor berinteraksi dengan use case, tetapi tidak memiliki control terhadap use case.
	Asosiasi antara aktor dan use case, digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengindikasikan aliran data.
	Asosiasi antara aktor dan use case yang menggunakan panah terbuka untuk mengindikasikan bila aktor berinteraksi secara pasif dengan sistem.
	<i>Include</i> , merupakan di dalam use case lain (<i>required</i>) atau pemanggilan use case oleh use case lain, contohnya adalah pemanggilan sebuah fungsi program.
	<i>Extend</i> , merupakan perluasan dari use case lain jika kondisi atau syarat terpenuhi.

(Sumber : Ade Handini : 2018)

2. Diagram Aktivitas (*Activity Diagram*)

Activity Diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Simbol-simbol yang digunakan dalam *activity diagram*, yaitu :

Tabel II.6. Simbol Diagram Aktivitas

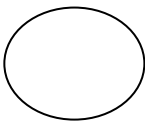
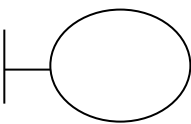
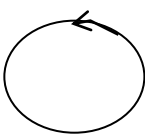
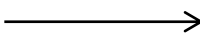
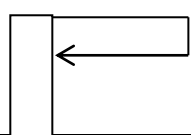
Gambar	Keterangan
	<i>Start point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktifitas.
	<i>End point</i> , akhir aktifitas.
	<i>Activites</i> , menggambarkan suatu proses/kegiatan bisnis.
	<i>Fork</i> (Percabangan), digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel atau untuk menggabungkan dua kegiatan paralel menjadi satu.
	<i>Join</i> (penggabungan) atau rake, digunakan untuk menunjukkan adanya dekomposisi.
	<i>Decision Points</i> , menggambarkan pilihan untuk pengambilan keputusan, <i>true</i> , <i>false</i> .
	<i>Swimlane</i> , pembagian <i>activity</i> diagram untuk menunjukkan siapa melakukan apa.

(Sumber : Ade Handini : 2018)

3. Sequence Diagram

Sequence diagram menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek. Simbol-simbol yang digunakan dalam *sequence diagram*, yaitu :

Tabel II.7. Simbol Sequence Diagram

Gambar	Keterangan
	<i>Entity Class</i> , merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data.
	<i>Boundary Class</i> , berisi kumpulan kelas yang menjadi <i>interface</i> atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan formentry dan <i>form</i> cetak.
	<i>Control class</i> , suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek
	<i>Message</i> , simbol mengirim pesan antar <i>class</i> .
	<i>Recursive</i> , menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.
	<i>Activation</i> , <i>activation</i> mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivitas sebuah operasi.
	<i>Lifeline</i> , garis titik-titik yang terhubung dengan objek, sepanjang <i>lifeline</i> terdapat <i>activation</i> .

(Sumber : Ade Handini : 2018)

4. Class Diagram (Diagram Kelas)

Class Diagram merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem, juga memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem. *Class diagram* juga menunjukkan atribut-atribut dan operasi-operasi dari sebuah kelas

dan *constraint* yang berhubungan dengan objek yang dikoneksikan. *Class diagram* secara khas meliputi: Kelas (*Class*), Relasi, *Associations*, *Generalization* dan *Aggregation*, Atribut (*Attributes*), Operasi (*Operations/ Method*), *Visibility*, tingkat akses objek eksternal kepada suatu operasi atau atribut. Hubungan antar kelas mempunyai keterangan yang disebut dengan *multiplicity* atau kardinaliti.

Tabel II.8. Simbol Class Diagram

Multiplicity	Penjelasan
1	Satu dan hanya satu
0..*	Boleh tidak ada atau 1 atau lebih
1..*	1 atau lebih
0..1	Boleh tidak ada, maksimal 1
n..n	Batasan antara. Contoh 2..4 mempunyai arti minimal 2 maksimum 4

(Sumber : Ade Handini : 2018)