

## BAB II LANDASAN TEORI

### 2.1 Penelitian Terdahulu

Dalam penelitian ini akan mendukung lima perpustakaan yang akan digunakan untuk penelitian, berikut ini adalah penelitian-penelitian yang diambil yaitu:

**Tabel 2.1 Studi Literature**

1	Judul	IbM Penggunaan Sistem Informasi Akuntansi Dalam Pengolaan Keuangan Usaha Warung Makan Kelurahan Bahu
	Penulis	Hisky Ryan Kawulur
	Tanggal/Tahun	(2018)
	Tujuan Penelitian	Untuk meningkatkan kemampuan pengelolaan keuangan mitra dalam pencatatan transaksi arus masuk dan arus keluar.
	Permasalahan	Mereka tidak mencatat transaksi penjualan atau pembelian dalam bisnis mereka, yang berarti mereka tidak mengetahui laba bersih mereka
	Subjek Penelitian	Usaha warung makan kelurahan bahu
	Metode Penelitian	
	Hasil Penelitian	Saat ini mitra memiliki kemampuan untuk mengelola dan meningkatkan sektor keuangan mereka.
2	Judul	Sistem Informasi Akuntansi Pendapatan Pada Toko Melosa Bandar Lampung
	Penulis	Muhammad Ma'mur dan Nurmaswan
	Tanggal/Tahun	(2019)
	Tujuan Penelitian	Sistem akuntansi yang dapat mempermudah dalam menginput data pendapatan, mempermudah pencarian data pendapatan, dan memudahkan admin dalam mengarsipkan data pendapatan. Untuk meningkatkan kinerja admin atau pengguna.
	Permasalahan	Toko Melosa dalam mengelola data pendapatan masih dilakukan secara manual terutama dalam bidang pencatatan pendapatan, pembuatan laporan dan penyimpanan laporan.
	Subjek Penelitian	Toko Melosa Bandar Lampung
	Metode Penelitian	Extreme programming (XP)
	Hasil Penelitian	Mempermudah dalam menyajikan data laporan pendapatan yang tepat dan akurat.

Tabel 2.1 Studi Literature (lanjutan)

		Mempermudah dalam melakukan pencarian data pendapatan Toko Melosa. Mempermudah dalam menginput data-data pendapatan Toko Melosa.
3	Judul	Aplikasi Pencatatan Laporan Penjualan Kita-Kita.Net Berbasis Web
	Penulis	Victor Marudut Mulia Siregar, Heru Sugara, dan Gabriel Andrian Purba
	Tanggal/Tahun	(2019)
	Tujuan Penelitian	Untuk merancang dan mengimplementasikan aplikasi pencatatan laporan penjualan berbasis web pada Kita-Kita.Net menggunakan metode prototype.
	Permasalahan	Kesalahan dalam pencatatan laporan, pencatatan laporan penjualan karena perbedaan buku laporan, kesalahan dalam pekerjaan yang sama berkali-kali karena hilangnya dokumen laporan, serta kesalahan dalam menghitung pendapatan antara manajer dan karyawan.
	Subjek Penelitian	Kita-Kita.Net
	Metode Penelitian	Pemodelan Prototype
Hasil Penelitian	Aplikasi ini dapat meminimalisir kesalahan yang terjadi dalam pencatatan laporan penjualan, serta meningkatkan efisiensi kerja, memastikan efisiensi waktu dan efisiensi energi di Kita-Kita.Net	
4	Judul	Analisis Penerapan Aplikasi Keuangan Berbasis <i>Android</i> Pada Laporan Keuangan Umkm Mekarsari, Depok
	Penulis	Anita Ria
	Tanggal/Tahun	(2018)
	Tujuan Penelitian	Penelitian ini bertujuan untuk menganalisis penerapan aplikasi keuangan berbasis android pada hasil laporan keuangan Usaha Mikro Kecil Menengah (UMKM) di wilayah Mekarsari Depok Jawa Barat.
	Permasalahan	Permasalahan yang umumnya dihadapi dalam meningkatkan keberhasilan UMKM antara lain ketersediaan bahan baku, kurangnya pengetahuan dalam berbagai aspek seperti aspek manajemen dan sumber daya manusia, serta lemahnya pelaporan keuangan yang akurat sehingga menyulitkan mengetahui laporan keuangan yang baik dan benar.
	Subjek Penelitian	UMKM Mekarsari, Depok

**Tabel 2.1 Studi Literature (lanjutan)**

	Metode Penelitian	Deskriptif kualitatif
	Hasil Penelitian	Pencatatan keuangan menggunakan Aplikasi Android Financial pada UMKM pemilik smartpone di kawasan Mekarsari Depok terbukti dapat mempermudah transaksi berupa jual-beli, hutang, pembayaran biaya operasional dan lain-lain, laporan keuangan pada smartpone berbasis Android tersedia mudah digunakan kapan saja. Informasi real-time sistem ini membantu membuat operasi lebih efektif dan efisien
5	Judul	Sistem Informasi Penjualan Barang pada Toko Candra Berbasis <i>Android</i>
	Penulis	Hartati Dyah Wahyuningsih, Paryanta, Hetri Candra Winoto
	Tanggal/Tahun	(2019)
	Tujuan Penelitian	Untuk mempermudah proses pelayanan, transaksi pembelian, laporan keuangan dan pesan antar pelanggan dalam lingkup Toko Candra
	Permasalahan	Toko Candra masih menggunakan sistem manual yaitu mengambil barang, kemudian menyerahkannya kepada kasir setelah mencatat barang menjadi catatan dan secara mental harga barang yang harus dibayar oleh pelanggan, dan dalam hal laporan tidak sesuai dengan sistem yang ada
	Subjek Penelitian	Toko Candra
	Metode Penelitian	Waterfall
	Hasil Penelitian	Aplikasi Penjualan Toko Candra berbasis <i>Android</i> , telah diimplementasikan pada Toko Candra dan mampu meningkatkan pelayanan, kemudahan bertransaksi dan menghemat waktu pelanggan.

Berdasarkan tinjauan pustaka diatas maka perbedaan antara penelitian terdahulu dengan penelitian yang dilakukan adalah:

1. Pengembangan sistem yang akan digunakan adalah berbasis *Android*
2. Sistem yang akan dibuat menggunakan *Android Studio* dan *Mysql* sebagai *Database Management System*
3. Metode yang digunakan metode Interatif

4. Pengujian sistem menggunakan pengujian *Black box* dan ISO 9126.

## 2.2 Pengertian UMKM

Menurut Undang-Undang Nomor 20 Tahun 2008 tentang Usaha Mikro, Kecil dan Menengah (UMKM), didefinisikan sebagai berikut:

1. Usaha Mikro adalah usaha produktif milik orang perseorangan dan atau badan usaha perorangan yang memenuhi kriteria sebagai berikut:
  1. sebuah. Aset Rp50.000.000 Memiliki kekayaan bersih kurang dari atau sama dengan Rp50.000.000,00 (lima puluh juta rupiah)
  2. Omzet Rp300.000.000 Memiliki hasil penjualan tahunan kurang dari Rp300.000.000 (tiga ratus juta rupiah).
2. Usaha Kecil adalah usaha ekonomi produktif yang berdiri sendiri, yang dilakukan oleh orang perseorangan atau badan usaha yang bukan merupakan anak perusahaan atau cabang perusahaan yang dimiliki, dikuasai, atau menjadi bagian baik langsung maupun tidak langsung dari usaha menengah atau usaha besar yang memenuhi kriteria berikut:
  1. sebuah.  $Rp50.000.000 < \text{Aset } Rp500.000.000$  Memiliki kekayaan bersih lebih dari Rp50.000.000,00 (lima puluh juta rupiah) sampai dengan paling banyak Rp500.000.000,00 (lima ratus juta rupiah) tidak termasuk tanah dan bangunan tempat usaha.
  2.  $Rp300.000.000 < \text{Omzet } 2.500.000.000$  Memiliki penjualan tahunan lebih dari Rp300.000.000 (tiga ratus juta rupiah) sampai dengan paling banyak Rp2.500.000.000 (dua miliar lima ratus juta rupiah).

3. Usaha menengah adalah usaha ekonomi produktif yang berdiri sendiri dan dijalankan oleh orang atau badan yang bukan merupakan anak perusahaan atau cabang dari perusahaan yang dimiliki, dikuasai, atau bagian dari usaha kecil atau besar yang:
  - a. standar. :
    1. sebuah. Rp500.000.000 < Aset Rp10.000.000.000,00 Memiliki kekayaan bersih lebih dari Rp500.000.000,00 (lima ratus juta rupiah) sampai dengan paling banyak Rp10.000.000.000,00 (sepuluh miliar rupiah) tidak termasuk tanah dan bangunan tempat usaha; atau
    2. Rp2.500.000.000 < Omzet Rp50.000.000 Omzet tahunan di atas Rp2.500.000.000 (dua miliar lima ratus juta rupiah) sampai dengan Rp50.000.000.000,00 (lima puluh miliar rupiah).

### **2.3 Pengertian Penjualan**

Menurut Hermawan (2006) bahwa penjualan adalah “kegiatan utama perusahaan dalam menghasilkan sesuatu dari suatu proses pertukaran yang terjadi di pasar.

Menurut Sadeli (2005) “Penjualan adalah tindakan menukarkan barang atau jasa dengan uang dengan mempengaruhi orang lain agar mau memiliki barang yang ditawarkan sehingga kedua belah pihak diuntungkan dan dipuaskan”.

Menurut Dharmmesta (2014) penjualan adalah ilmu dan seni dalam mempengaruhi pribadi yang dilakukan oleh penjual untuk mengajak orang lain agar bersedia membeli barang atau jasa yang ditawarkan.

Dari beberapa pengertian menurut para ahli di atas, dapat dikatakan bahwa penjualan adalah transaksi barang atau jasa yang menghasilkan pendapatan bagi penjual yang dapat dilakukan baik secara tunai maupun secara kredit.

#### **2.4 Penegertian Kasir**

Menurut Winardi (1991) Kasir adalah seseorang yang pekerjaannya menerima uang pembayaran saat pembelian produk barang atau jasa dan melakukan pengembalian uang sisa pembayaran, sekaligus menyerahkan produk barang atau jasa kepada pelanggan (customer) di loket-loket kasir di suatu toko, super market, mini market, hotel, restoran, rumah sakit, ataupun department store.

#### **2.5 Pengertian Pendapatan**

Menurut Kieso, Warfield dan Weygantd (2016) pendapatan adalah arus masuk bruto dari manfaat ekonomi yang timbul dari aktivitas normal entitas selama suatu periode, jika arus masuk tersebut menghasilkan peningkatan ekuitas yang tidak berasal dari kontribusi investasi.

Menurut Skousen dan Stice (2011) pendapatan adalah arus masuk atau penyelesaian (atau kombinasi keduanya) dari penyerahan atau produksi barang, penyediaan jasa atau pelaksanaan kegiatan utama atau kegiatan pusat yang sedang berlangsung.

Dari Pengertian diatas bahwa pendapatan adalah peningkatan manfaat perekonomian selama suatu periode akuntansi tertentu berupa pendapatan atau peningkatan aset atau penurunan kewajiban yang mengakibatkan peningkatan ekuitas, yang tidak berasal dari investasi. bertambah jumlahnya aset atau penurunan

kewajiban dapat berasal dari barang/jasa atau kegiatan usaha lainnya dalam suatu periode.

## **2.6 Aplikasi Mobile**

### **2.6.1 Pengertian Aplikasi Mobile**

Menurut Buyens (2011) aplikasi *mobile* adalah berasal dari kata aplikasi dan *mobile*. Aplikasi yang berarti aplikasi, aplikasi, penggunaan. Dari segi aplikasi adalah program siap pakai yang dirancang untuk menjalankan suatu fungsi bagi pengguna atau aplikasi lain dan dapat digunakan oleh sasaran yang dituju, sedangkan *mobile* dapat diartikan sebagai perpindahan dari satu tempat ke tempat lain.

Kata *mobile* memiliki arti bergerak atau bergerak, sehingga aplikasi *mobile* menurut Rangsang Purnama (2010) adalah sebutan untuk aplikasi yang berjalan pada perangkat bergerak. Dengan menggunakan aplikasi *mobile*, Anda dapat dengan mudah melakukan berbagai aktivitas mulai dari hiburan, berjualan, belajar, mengerjakan pekerjaan kantor, browsing, dan bermain game.

Penggunaan aplikasi *mobile* untuk hiburan paling disukai oleh pengguna ponsel, karena dengan memanfaatkan fitur game, music player, dan video player, kita menjadi lebih mudah untuk menikmati hiburan kapanpun dan dimanapun.

Perangkat seluler datang dalam banyak variasi dalam hal ukuran, desain, tata letak, tetapi mereka memiliki karakteristik yang sangat berbeda dari sistem desktop. Perangkat seluler memiliki memori kecil.

### 2.6.2 Karakteristik Perangkat Mobile

Perangkat *mobile* datang dalam banyak variasi dalam hal ukuran, desain, dan tata letak, tetapi mereka memiliki karakteristik yang sangat berbeda dari sistem desktop, termasuk:

1. Ukuran kecil

Perangkat *mobile* memiliki ukuran yang kecil. Konsumen menginginkan perangkat terkecil untuk kenyamanan dan mobilitas mereka.

2. Memori terbatas

Perangkat *mobile* juga memiliki memori yang kecil, yaitu primer (RAM) dan sekunder (disk). Keterbatasan ini merupakan salah satu faktor yang mempengaruhi penulisan program untuk berbagai jenis perangkat tersebut. Dengan membatasi jumlah memori, pertimbangan khusus harus diambil untuk mempertahankan penggunaan sumber daya yang mahal ini.

3. Kekuatan pemrosesan terbatas

Sistem seluler tidak sekuat rekan desktop mereka. Ukuran, teknologi, dan biaya adalah beberapa faktor yang mempengaruhi status sumber daya ini. Seperti hard drive dan RAM, Anda dapat menemukannya dalam ukuran yang tepat dalam paket kecil.

4. Konsumsi data rendah

Perangkat *mobile* mengkonsumsi lebih sedikit daya dibandingkan dengan mesin desktop. Perangkat ini harus menghemat daya karena beroperasi dalam keadaan di mana suplai dibatasi oleh baterai.

#### 5. Kuat dan dapat diandalkan

Karena perangkat seluler dibawa ke mana-mana, mereka harus cukup kuat untuk menahan benturan, gerakan, dan tetesan air sesekali.

#### 6. Konektivitas terbatas

Perangkat *mobile* memiliki bandwidth rendah, beberapa di antaranya bahkan tidak terhubung. Kebanyakan dari mereka menggunakan koneksi nirkabel.

#### 7. Umur Pendek

Perangkat konsumen ini menyala dalam hitungan detik, sebagian besar selalu menyala. Ambil kasus ponsel, mereka boot dalam hitungan detik dan kebanyakan orang tidak mematikan ponsel mereka bahkan di malam hari. PDA akan menyala jika Anda menekan tombol dayanya.

## 2.7 Android

Menurut Sifaat (2014) “*Android* adalah sebuah sistem operasi untuk perangkat *mobile* berbasis *linux* yang mencakup sistem operasi, *middleware* dan aplikasi. *Android* menyediakan *platform* terbuka bagi para pengembang untuk menciptakan aplikasi mereka”. *Android* mengalami perkembangan dari versi ke versi disesuaikan dengan kebutuhan, beberapa versi *android* yang telah dirilis hingga saat ini diantaranya *Android* versi 1.1 yang dirilis pada tahun 2008 lalu pada tahun yang sama diluncurkan *Android* versi 1.5 (*Cupcake*) dan kemudian *Android* versi 1.6 (*Donut*), *Android* versi 2.1 yang diberi nama *Eclair*, kemudian pada tahun berikutnya dirilis *Android* versi 2.2 (*Froyo*), *Android* versi 2.3 (*Gingerbread*), lalu dirilis *Android* versi 3.0/3.1 (*Honeycomb*) yang dirancang khusus untuk *tablet*. Selanjutnya diluncurkan pada akhir tahun 2011 *Android* versi 4.0 (*Ice Cream*

*Sandwich*), kemudian pada tahun 2012 dirilis *Android* versi 4.1 (*Jelly Beans*) dan *Android* versi 4.4 (*Kitkat*) pada tahun 2013. Lalu diluncurkan *Android* versi 5.1 (*Lollipop*), *Android* versi 6.0 (*Marshmallow*), *Android* versi 7.0 (*Nougat*), *Android* versi 8.0 (*Oreo*), *Android* versi 9.0 (*Pie*), *Android* 10, hingga versi terbaru yaitu *Android* 11. Banyaknya versi *android* yang dirilis adalah pembaharuan dan penambahan fitur-fitur sebagai penyempurnaan dari versi sebelumnya. Mulai dari *android* versi 1.1 sampai *android* versi 1.6 (*Donut*) yang dilengkapi dengan pembaharuan pada kamera, galeri, *bluetooth*, kontak telepon, resolusi *display* dan jaringan VPN. Kemudian pada *android* versi 2.2 (*Eclair*) sampai *android* versi 2.3 (*Gingerbread*) dilakukan perubahan pada pengoptimalan *hardware*, dukungan *flash* pada kamera, dukungan HTML5, optimasi kecepatan, memori dan dukungan USB tethering atau *Wi-Fi hotspot*, peningkatan manajemen daya dan peningkatan performa serta adanya penambahan sensor. Berbeda dengan *android* versi sebelumnya, *android* versi 4.0 (*Ice Cream Sandwich*) dan *android* versi 4.1 (*Jelly Beans*) memiliki kelebihan yaitu adanya *platform* yang dapat berjalan di *tablet* dan *mobile*, konsumsi baterai yang lebih hemat serta peningkatan performa yang lebih signifikan. Sedangkan *android* versi 4.4 (*Kitkat*) lebih ramah terhadap perangkat dengan spesifikasi seadanya. Perubahan besar yang ada pada *android* versi 5.1 (*Lollipop*) adalah adanya dukungan arsitektur 64 bit sehingga memungkinkan penggunaan RAM diatas 3 GB pada perangkat *hardware*. *Android* versi 6.0 (*Marshmallow*) apabila dilihat dari segi tampilan hampir sama dengan *android* versi 5.1 (*Lollipop*) hanya saja pada versi ini fitur keamanan di OS sudah lebih ditingkatkan. *Android* versi terbaru adalah *android* versi 7.0 (*Nougat*), pembaharuan pada versi ini yaitu adanya penambahan *emoji* dan fitur *google*

*assistant* serta OS memiliki kemampuan 2 bahasa dalam satu waktu. *Android* versi 8.0 (Oreo), Pembaharuan pada versi Oreo membawa beberapa fitur seperti fitur Autofill yang memberikan kemudahan dalam mengisi formulir misal, dukungan gambar dalam gambar dan pengoptimalan booting agar lebih cepat. *Android* versi 9.0 (Pie), pembaharuan pada versi ini yaitu adanya kemampuan AI atau kecerdasan buatan yang dinamakan *AI Smartphone*. *Android* 10, pembaharuan pada versi ini yaitu adanya peningkatan fitur sound amplifier untuk mengatur kualitas audio. hingga versi terbaru yaitu *Android* 11, pembaharuan pada versi ini yaitu adanya fitur keamanan lebih canggih dan fitur notifikasi, screen recorder tanpa aplikasi tambahan, picture-in-picture, hingga mengambil screenshot panjang.

### **2.7.1 *Android Software Development Kit (Android SDK)***

Menurut Safaat (2014) menyatakan bahwa, “*Android SDK* merupakan alat yang dibutuhkan untuk mengembangkan aplikasi berbasis *Android* dengan menggunakan bahasa pemrograman Java”. Saat ini *Android SDK* telah menjadi alat dan Application Programming Interface (API) untuk mengembangkan aplikasi berbasis *Android*.

### **2.7.2 *Android Studio***

Menurut Herdi (2014), “*Android Studio* adalah IDE yang dapat digunakan untuk pengembangan aplikasi *Android* dan dikembangkan oleh Google. *Android Studio* adalah singkatan dari Eclipse IDE dan didasarkan pada Java IDE IntelliJ IDEA yang populer. *Android Studio* merupakan kepanjangan dari Eclipse IDE dan berdasarkan Java IDE IntelliJ IDEA yang populer. *Android Studio* bertujuan untuk

menggantikan Eclipse sebagai IDE resmi untuk pengembangan aplikasi *Android*. *Android Studio* menyertakan Eclipse IDE Termasuk, ada banyak fitur baru dibandingkan dengan *Android Studio* yang menggunakan Gradle sebagai lingkungan build. dari Eclipse. Fitur lainnya termasuk:

1. Menggunakan *Gradle-based build system* yang *fleksibel*.
2. Bisa mem-*build multiple APK*.
3. *Template support* untuk *Google Services* dan berbagai macam tipe perangkat.
4. *Layout editor* yang lebih bagus.
5. *Built-in support* untuk *Google Cloud Platform*, sehingga mudah untuk integrasi dengan *Google Cloud Messaging* dan *App Engine*.
6. Import library langsung dari *Maven repository*.

## **2.8 Pengertian UML (Unified Modeling Language)**

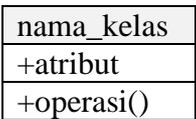
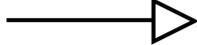
Menurut Rosa A.S. dan M. Shalahudin (2018) UML (*Unified Modeling Language*) adalah salah standar bahasa yang banyak digunakan dalam dunia industri untuk mendefinisikan requirement, membuat analisis & desain, serta mengembangkan arsitektur dalam pemograman berorientasi objek.

Adapun pengembangan diagram-diagram antara lain :

### **1. Class Diagram**

Diagram kelas menggambarkan struktur suatu sistem dalam hal mendefinisikan kelas-kelas yang dibuat untuk membangun sistem. Berikut adalah simbol – simbol yang ada pada diagram kelas di tabel 2.2 :

Tabel 2.2 Simbol dan Deskripsi dari Diagram Kelas

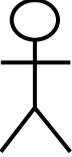
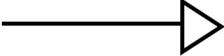
Simbol	Deskripsi
Kelas 	Kelas pada struktur sistem
Antarmuka / <i>interface</i> 	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek
Asosiasi / <i>association</i> 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
Asosiasi berarah / <i>directed association</i> 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
Generalisasi 	Relasi antarkelas dengan makna generalisasi-spesialisasi (umum khusus)
Kebergantungan / <i>dependency</i> 	Relasi antarkelas dengan makna kebergantungan antarkelas
Agregasi / <i>aggregation</i> 	Relasi antarkeals dengan makna semua-bagian ( <i>whole-part</i> )

Sumber : Rosa A.S dan M. Shalahudin (2018)

## 2. Use Case Diagram

*Use case* atau diagram *use case* adalah pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Simbol - simbol yang digunakan untuk membuat *use case* diagram dapat dilihat pada tabel 2.3 sebagai berikut :

Tabel 2.3 Simbol dan Deskripsi dari *Use Case Diagram*

Simbol	Deskripsi
Use Case 	Fungsionalitas yang disediakan sistem sebagai unit – unit yang bertukar pesan antar unit atau aktor; biasanya digunakan menggunakan kata kerja di awal di awal fase nama <i>use case</i>
Aktor / <i>actor</i>  Nama aktor	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuatnya sendiri, sehingga meskipun simbol aktor adalah gambaran seseorang, aktor tersebut belum tentu orang; biasanya diekspresikan menggunakan kata benda di awal frasa nama aktor
Asosiasi / <i>association</i> 	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor
Ekstensi / <i>extend</i> <<extend>> 	Hubungan <i>use case</i> tambahan ke <i>use case</i> di mana <i>use case</i> yang ditambahkan dapat berdiri sendiri bahkan tanpa <i>use case</i> tambahan; mirip dengan prinsip pewarisan dalam pemrograman berorientasi objek; Biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan, misalnya
Generalisasi / <i>generalization</i> 	hubungan generalisasi dan spesialisasi (umum – khusus) antara dua kasus penggunaan di mana satu fungsi lebih umum daripada yang lain.

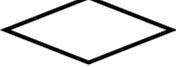
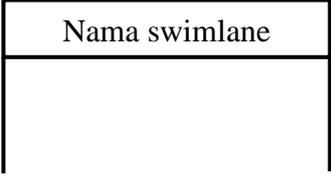
<p>Menggunakan / <i>include</i> / <i>uses</i></p> <p style="text-align: center;">&lt;&lt;uses&gt;&gt; -----▶</p> <p style="text-align: center;">&lt;&lt;include&gt;&gt; ————▶</p>	<p>Hubungan <i>use case</i> tambahan ke <i>use case</i> di mana <i>use case</i> yang ditambahkan membutuhkan <i>use case</i> ini untuk meningkatkan kemampuan atau sebagai syarat agar <i>use case</i> ini dieksekusi</p>
---	---

Sumber : Rosa A.S dan M. Shalahudin (2018)

### 3. Activity Diagram

Activity diagram atau diagram aktivitas menggambarkan alur kerja (*workflow*) atau aktivitas dari suatu sistem atau proses atau menu dalam perangkat lunak. Simbol yang digunakan untuk membuat diagram aktivitas dapat dilihat pada tabel 2.4 sebagai berikut

**Tabel 2.4 Simbol dan Deskripsi dari Diagram Aktivitas**

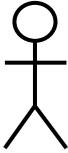
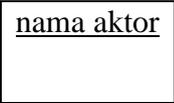
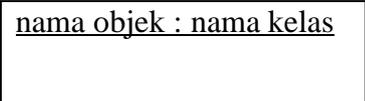
Simbol	Deskripsi
Status awal 	Status awal aktivitas sistem, diagram aktivitas memiliki status awal
Aktivitas 	Kegiatan yang dilakukan oleh sistem, kegiatan biasanya dimulai dengan kata kerja
Percabangan / <i>decision</i> 	Percabangan asosiasi di mana jika ada lebih dari satu pilihan kegiatan
Penggabungan / <i>join</i> 	Menggabungkan asosiasi di mana lebih dari satu aktivitas digabungkan menjadi satu
Status akhir 	Keadaan akhir yang dilakukan sistem, diagram aktivitas memiliki keadaan akhir
<i>Swimlane</i>  Atau 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

Sumber : Rosa A.S dan M. Shalahudin (2018)

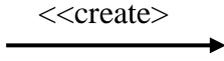
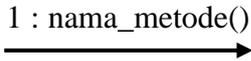
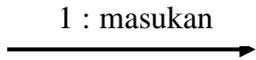
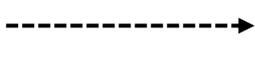
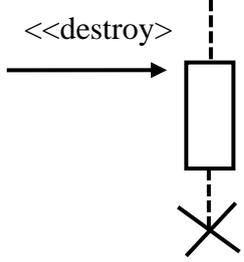
## 4. Sequence diagram

Sequence diagram menggambarkan objek dalam *use case* dengan menggambarkan waktu hidup objek dan pesan yang dikirim dan diterima antar objek. Simbol yang digunakan untuk membuat diagram urutan dapat dilihat pada tabel 2.5 sebagai berikut :

**Tabel 2.5 Simbol dan Deskripsi dari Diagram Sequence**

Simbol	Deskripsi
<p>Aktor</p>  <p>nama_aktor</p> <p>Atau</p>  <p>Tanpa waktu aktif</p>	<p>Orang, proses, atau sistem lain yang berinteraksi dengan situs informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, sehingga meskipun simbol aktor adalah citra seseorang, aktor belum tentu orang; biasanya diekspresikan menggunakan kata benda di awal frasa nama aktor</p>
<p>Garis hidup / <i>lifeline</i></p> 	<p>Menyatakan kehidupan suatu objek</p>
<p>Objek</p> 	<p>Deklarasikan objek yang berinteraksi dengan pesan</p>
<p>Waktu aktif</p> 	<p>Menyatakan suatu objek dalam keadaan aktif dan berinteraksi, semua yang berhubungan dengan waktu aktif ini merupakan langkah yang dilakukan di dalamnya.</p>

**Tabel 2.5 Simbol dan Deskripsi dari Diagram Sequence (lanjutan)**

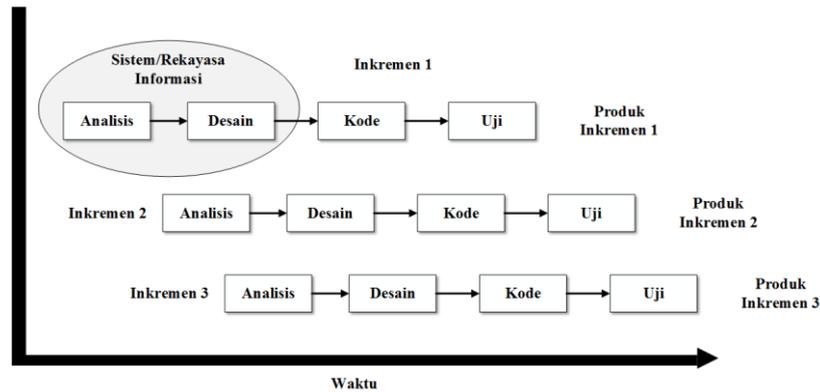
Pesan tipe create 	Menyatakan objek untuk membuat objek lain, arah panah menunjuk ke objek yang sedang dibuat
Pesan tipe call 1 : nama_metode() 	Menyatakan bahwa suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri
Pesan tipe send 1 : masukan 	Menyatakan bahwa suatu objek mengirimkan data/input/operasi atau metode menghasilkan return ke objek tertentu, arahnya mengarah ke objek yang menerima return
Pesan tipe return 1 : keluaran 	Menyatakan bahwa suatu objek yang telah melakukan operasi atau metode menghasilkan pengembalian ke objek tertentu, arahnya menunjuk ke objek yang menerima pengembalian
Pesan tipe destroy 	Menyatakan bahwa suatu objek mengakhiri kehidupan objek lain, arah panah menunjuk ke objek yang dihentikan, sebaliknya jika ada buat maka ada penghancuran ( <i>destroy</i> )

Sumber : Rosa A.S dan M. Shalahudin (2018)

## 2.9 Metode Pengembangan Sistem

### 2.9.1 Metode Interatif

Menurut Rosa A.S-M.Shalahudin (2018), Model Interatif (*iterative model*) menggabungkan proses pada model air terjun dan interatif pada model prototipe. Model inkremental akan menghasilkan versi perangkat lunak yang telah mengalami fungsionalitas tambahan untuk setiap penambahan (inkremen/*increment*), berikut adalah gambar dari model inkremental, dapat dilihat pada Gambar 2.1 :



**Gambar 2.1 Ilustrasi model *Interatif***  
 Sumber : Rosa A.S dan M. Shalahudin (2018)

Model inkremental dibuat untuk mengatasi model *waterfall* yang tidak mengakomodasi iterasi, dan mengatasi kelemahan metode *prototype* yang memiliki proses yang terlalu pendek dan setiap iterasi proses tidak selalu menghasilkan produk/aplikasi untuk setiap tahapannya.

### 2.9.2 Black Box Testing

Menurut Rosa A.S-M.Shalahudin (2018), *Black Box Testing* (pengujian kotak hitam) yaitu pengujian perangkat lunak dalam hal spesifikasi fungsional tanpa menguji desain dan kode program.

*Black Box Testing*, dilakukan tanpa pengetahuan rinci tentang struktur internal dari sistem atau komponen yang diuji juga disebut sebagai *behavioral testing*, *specification-based testing*, *input/output testing* atau *funcional testing*, *Black Box Testing* berfokus pada kebutuhan fungsional pada software, berdasarkan pada spesifikasi kebutuhan dari *software*. Dengan adanya *black box testing*, perancang *software* dapat menggunakan satu set kondisi input yang dapat sepenuhnya memeriksa persyaratan fungsional keseluruhan program. Pengujian kotak hitam bukanlah teknik alternatif selain *white box testing*. Lebih dari itu, ia

merupakan pendekatan pelengkap dalam mencakup *error* dengan kelas yang berbeda dari metode *white box testing*.

Kategori *error* yang diketahui melalui black box testing :

1. Fungsi yang hilang atau tak benar
2. Error dari antar-muka
3. Error dari struktur data atau akses eksternal database
4. Error dari kinerja atau tingkah laku
5. Error dari inisialisasi dan terminasi.

Adapun kerangka yang akan digunakan untuk melakukan pengujian pada sebagai berikut.

**Tabel 2.6** Kerangka Pengujian

<b>Kasus dan Hasil Uji</b>			
<b>Data Masukan</b>	<b>Data Diharapkan</b>	<b>Pengamatan</b>	<b>Kesimpulan</b>
			Diterima ( ) Ditolak ( )
			Diterima ( ) Ditolak ( )

Rumus dari kuesioner pengujian *black box* menggunakan analisa deskriptif, ditunjukkan sebagai berikut:

$$\% Skor = \frac{Skor\ Aktual}{Skor\ Ideal} \times 100\%$$

Keterangan:

Skor Aktual : Jawaban diterima responden

Skor Ideal : Total jumlah butir soal yang telah diujikan kepada responden

Adapun kriteria hasil perhitungan kelayakan sistem dari pengujian ini adalah sebagai berikut:

**Tabel 2.7** Kriteria Persentase Skor Tanggapan Responden

<b>Jumlah Skor</b>	<b>Kriteria</b>
0.00 – 36.00	Tidak Baik / Tidak Layak
36.01 – 52.00	Kurang Baik / Kurang Layak
52.01 – 68.00	Cukup Baik / Cukup Layak
68.01 – 84.00	Baik / Layak
84.01 – 100	Sangat Baik / Sangat Layak

### 2.9.3 ISO 9126

Weber (1999) ISO 9126 adalah salah satu kerangka kerja standar internasional yang digunakan untuk melakukan pengujian kualitas perangkat lunak, yang dibuat oleh *International Organization for Standardization* (ISO) dan *International Electrotechnical Commission* (IEC). Standar internasional memiliki kemampuan untuk mendefinisikan kualitas produk perangkat lunak, karakteristik kualitas, model, dan metrik terkait untuk mengembangkan dan menentukan kualitas produk perangkat lunak. Model ISO 9126 memiliki 6 (enam) faktor dan beberapa subfaktor.

Enam karakteristik model ISO 9126 meliputi *Functionality*, yaitu kemampuan perangkat lunak untuk menyediakan fungsi sesuai kebutuhan pengguna saat digunakan dalam kondisi tertentu. Situs web harus dapat diakses oleh pengguna dengan lingkungan sistem yang berbeda tanpa mengorbankan fungsionalitas yang ada;

*Reliability* (Kehandalan) adalah kemampuan perangkat lunak untuk mempertahankan kinerjanya saat digunakan dalam kondisi tertentu. Keandalan suatu perangkat lunak dapat ditentukan dari jumlah input yang dapat menyebabkan kegagalan saat dijalankan. Hal ini dapat diamati oleh pengguna.

*Usability* (Kebergunaan) berkaitan dengan apakah produk tersebut dapat mencapai tujuan tertentu secara efektif, efisien, dan memperoleh kepuasan setelah

digunakan. Aspek usability dapat diukur dengan menggunakan instrumen berupa angket. Kuesioner ini akan diisi oleh pengguna setelah mereka mencoba menggunakan aplikasi. Kemudahan penggunaan adalah sejauh mana perangkat lunak mudah digunakan, yang sering ditunjukkan dengan menggunakan sub-atribut kemudahan pemahaman, kemudahan belajar, dan pengoperasian..

*Efficiency* (Efisiensi) berkaitan dengan kemampuan perangkat lunak untuk memberikan kinerja yang sesuai terhadap jumlah sumber daya yang digunakan pada saat itu. Jika ingin membuat software yang handal, aspek efisiensi harus benar-benar diperhatikan. Penggunaan sumber daya yang tidak efisien, misalnya penggunaan algoritma yang tidak tepat dapat menyebabkan kinerja perangkat lunak menjadi lamban.

*Maintainability* (Pemeliharaan) berkaitan dengan perangkat lunak untuk mengelola. Modifikasi termasuk perbaikan, peningkatan atau penyesuaian terhadap perubahan lingkungan, persyaratan, dan spesifikasi fungsional; dan

*Portability* (Portabilitas) terkait dengan kemampuan perangkat lunak untuk beroperasi atau bekerja di lingkungan yang berbeda. Untuk menguji tingkat portabilitas aplikasi berbasis android, aplikasi dicoba dijalankan menggunakan smartphone yang berbeda. Selain itu, eksperimen menggunakan perangkat yang berbeda juga dapat menentukan tingkat portabilitas perangkat lunak.

Adapun kerangka yang akan digunakan untuk melakukan pengujian pada sebagai berikut.

**Tabel 2.8** Kriteria Persentase Skor Tanggapan Responden

<b>Aspek</b>	<b>Skor Aktual</b>	<b>Skor Ideal</b>	<b>% Skor Aktual</b>	<b>Kriteria</b>
<i>Functionality</i>				
<i>Reliability</i>				
<i>Usability</i>				
<i>Efficiency</i>				
<i>Maintainability</i>				
<i>Portability</i>				
<b>Total</b>				

Skor aktual diperoleh dari hasil perhitungan setiap aspek yang diujikan, sedangkan skor ideal adalah skor maksimal untuk setiap aspek sedangkan persentase skor aktual diperoleh dengan menggunakan persamaan

$$\% \text{ Skor Aktual} = \frac{\text{Skor Aktual}}{\text{Skor Ideal}} \times 100\%$$

Untuk menilai kriteria berdasarkan persentase skor aktual dengan menggunakan acuan pada Tabel 2.9.

### 2.9 Kriteria Penilaian

<b>Jumlah Skor</b>	<b>Kriteria</b>
20,00% - 36,00%	Tidak Baik / Tidak Layak
36,01% - 52,00%	Kurang Baik / Kurang Layak
52,01% - 68,00%	Cukup Baik / Cukup Layak
68,01% - 84,00%	Baik / Layak
84,01% - 100%	Sangat Baik / Sangat Layak