

BAB II

LANDASAN TEORI

2.1 Teori Dasar

Mobile application merupakan sebuah aplikasi perangkat lunak yang dirancang untuk digunakan di telepon genggam. Aplikasi ini berbeda dengan aplikasi yang memang dirancang hanya untuk dijalankan di perangkat komputer saja. Pada awalnya aplikasi ini dirancang hanya untuk mempermudah dalam pengiriman email, kalender, dan basis data.

Namun tingginya permintaan pengguna akan aplikasi perangkat lunak untuk mendukung berbagai aktifitas sehari-hari, membuat para pengembang aplikasi berinovasi lebih dalam menambahkan fitur-fitur baru didalam *mobile aplikasi*. Aplikasi juga sangat membantu di sektor penjualan secara online, dan mempermudah berbagai pihak. Dalam hal ini beberapa perusahaan juga sudah memberlakukan jual beli di dalam koperasi karyawan secara online, koperasi karyawan ini ada karena kesepakatan bersama dan memiliki tujuan untuk menambah kesejahteraan karyawan dan memajukan umkm.

2.1.1 Karyawan

Seseorang yang bekerja untuk suatu perusahaan, organisasi, ataupun individu lainnya dalam kapasitas sebagai pekerja biasa juga disebut sebagai karyawan. Dalam sebuah perusahaan sendiri karyawan memiliki tingkatan dalam sebuah organisasi, contohnya karyawan eksekutif, karyawan manajerial, karyawan staff, maupun sebagai karyawan produksi. Semua itu tergantung daripada peran mereka dalam perusahaan tersebut. Dalam kata lain karyawan adalah seseorang yang menjual jasa baik tenaga maupun pikiran untuk kemudian mendapatkan kompensasi. (Widyaningrum, n.d.)

Seorang karyawan juga memiliki peraturan yang tercantum dalam Undang-Undang tenaga kerja Nomor 13 Tahun 2003 tentang ketenagakerjaan, tenaga kerja (*manpower*) diartikan setiap orang yang mampu melakukan pekerjaan guna menghasilkan barang atau jasa untuk memenuhi target dari perusahaan. Peningkatan produktivitas kerja karyawan diharapkan dapat mendukung tercapainya tujuan dan menjaga kelangsungan hidup perusahaan. (Putu, Laksmiari, & Ekonomi, 2019)

Setiap perusahaan selalu menginginkan SDM yang sudah direkrut memiliki kinerja yang cekatan. Hal ini sejalan dengan perusahaan yang selalu memiliki target produksi. Untuk menghasilkan SDM yang berkualitas tentunya berbagai cara dilakukan oleh perusahaan, seperti mengadakan pelatihan dasar, menengah ataupun tingkat lanjut bagi karyawan agar memiliki keahlian dan kinerja yang memuaskan. (Juanita Nurwin & Negeri Surabaya, n.d.)

Pekerjaan merupakan hal yang mendasar dan sangat bervariasi, seperti mengumpulkan sumber daya alam menggunakan cara tradisional sampai menggunakan cara yang *modern*. Hingga pengoperasian teknologi yang banyak menggunakan fisik dan mental yang dilakukan oleh manusia atau yang menggunakan mesin *robotic*. Banyak pekerjaan hanya dapat dilakukan oleh orang-orang yang memiliki *skill* khusus, dan menggunakan berbagai jenis alat kerja, serta sumber daya lainnya.

Berikut adalah gambar karyawan di area produksi:



Gambar 2. 1 Foto Karyawan
Sumber : (Citra Tubindo Tbk, 2024)

2.1.2 Koperasi

Koperasi karyawan atau sering disebut dengan istilah *employee cooperative* merupakan koperasi yang ada dan di bentuk di dalam sebuah perusahaan oleh para karyawan. Adapun pengurus dan anggota nya dari karyawan perusahaan itu sendiri. Koperasi karyawan juga berbadan hukum karena beranggotakan karyawan yang terikat kontrak dengan perusahaan. Koperasi ini memiliki manfaat mensejahterakan ekonomi dan sosial kepada anggotanya. Pada umumnya kegiatan koperasi ini bergerak di bidang penjualan bahan pokok dan lain-lain. Berdasarkan Pasal 15 Undang-Undang No.25 tahun 1992 koperasi dibagi menjadi dua jenis, yaitu koperasi primer dan koperasi sekunder.(Riswanto & Laluma, 2020a)

Dimana koperasi primer merupakan koperasi yang bersifat perorangan dan beranggotakan paling sedikit 25 orang. Sedangkan koperasi sekunder adalah koperasi yang dibentuk oleh sekumpulan koperasi primer. Dalam hal ini, koperasi karyawan juga termasuk ke dalam koperasi primer, karena koperasi ini dibuat oleh sekumpulan orang

dengan kesamaan visi misi koperasi karyawan. Dalam mengikuti perkembangan revolusi industri 5.0, digitalisasi dalam koperasi karyawan juga sangat dibutuhkan untuk memunjang berbagai aktifitas dari koperasi karyawan ini.(Dasuki et al., 2023)

Tujuan dibentuknya koperasi karyawan yaitu untuk berkontribusi menambah kesejahteraan karyawan dan mengangkat produktifitas umkm karyawan. Adapun beberapa manfaat dari adanya koperasi karyawan, diantaranya:

1. Meningkatkan kesejahteraan serta menggali potensi usaha dari karyawan yang menjadi anggota koperasi.
2. Berperan aktif dalam meningkatkan kualitas bisnis anggota koperasi.
3. Mengembangkan dan memajukan perekonomian nasional dengan dasar yang kuat dan ilmu pengetahuan bisnis yang luas.

Pada umumnya koperasi karyawan sangat transparasi dan fleksibel, sehingga tidak diwajibkan ataupun paksaan bagi seorang karyawan untuk mendaftar jadi anggota koperasi atau dalam artian bagi seluruh karyawan boleh mendaftar sebagai anggota koperasi dengan syarat karyawan aktif di perusahaan. (Riswanto & Laluma, 2020b)

Koperasi karyawan juga memiliki prinsip-prinsip dalam menjalankan aktifitas seputar koperasi yang meliputi:

1. Pembagian sisa hasil usaha dilakukan secara adil.
2. Dikelola dengan demokratis.
3. Mendapatkan pengarahan serta wawasan tentang koperasi.
4. Bersifat mandiri dan sukarela.
5. Memajukan UMKM di ruang lingkup perusahaan.

Berikut merupakan gambar koperasi karyawan ditempat penelitian:



Gambar 2. 2 Koperasi Karyawan
Sumber : (Citra Tubindo Tbk, 2024)

2.1.3 Aplikasi

Aplikasi ialah subkelas perangkat lunak komputer yang memakai langsung kemampuan computer dalam melakukan tugas sesuai keinginan *user*. Aplikasi bisa dikatakan sebagai command interpreter dan dijalankan oleh pengguna komputer untuk diproses oleh perangkat keras. Aplikasi ialah program yang langsung menjalankan proses yang digunakan oleh pengguna di komputer. (Priandika, 2023)

Aplikasi adalah sekumpulan file dan berisikan kode program dan menghubungkan *user* ke perangkat keras komputer. Aplikasi artinya subkelas perangkat lunak komputer yang memakai langsung kelebihan komputer untuk menjalankan perintah yang diinginkan oleh pengguna.(Kasamilale & Candra, 2023)

Aplikasi bisa dikategorikan sebagai penafsir perintah yang dilakukan user untuk diproses oleh perangkat keras. Aplikasi, dikenal sebagai perangkat lunak, ialah program komputer dengan perintah yang bisa dimodifikasi dan tidak terlalu sulit. Aplikasi pada umumnya berguna untuk mengontrol perangkat keras (driver perangkat), menghitung, serta berinteraksi dengan aplikasi lain (seperti OS & bahasa pemrograman). Pada umumnya, aplikasi terbagi jadi 3 tingkatan: program aplikasi (seperti Microsoft Office), OS (seperti Microsoft Windows) dan bahasa pemrograman (seperti *Python, PHP, C++, Java*).

Berikut merupakan logo aplikasi *mobile*:



Gambar 2. 3 Aplikasi Pesan
Sumber : (WhatsApp LLC, 2024)

Di zaman yang serba canggih saat ini, seseorang akan semakin mudah dalam menjalankan kegiatan sehari-hari ataupun bekerja dengan bantuan aplikasi. Dan seseorang juga dapat menambah relasi dengan lingkungan baru melalui sosial media. Berbagai aplikasi sudah tersedia, pengguna hanya tinggal mengunduhnya saja. Jadi dengan berbagai kemudahan yang didapat tentu user akan semakin terbantu dan menggunakannya dengan baik.

2.1.3.1 *Android*

Berdasarkan penelitian (Sulistiyorini et all, 2022) *Android* merupakan sistem operasi yang memiliki biaya terjangkau. Sehingga bisa dijadikan pilihan bagi perusahaan yang bergerak di bidang teknologi, manufaktur, maupun *oil* dan *gas*. *Android* merupakan sistem operasi berbasis *Linux* dengan lisensi *Apache 2.0* yang dirancang untuk perangkat layar sentuh seperti *smartphone*, jam tangan, tablet, PC, dan TV. Seiring dengan perkembangannya, sistem operasi *android* telah mengalami banyak perubahan dan perbaikan. Seperti nama-nama *Android* versi baru yang diberi nama-nama makanan, yang menjadi daya tarik tersendiri bagi para penggunanya.

Berdasarkan penelitian ((Monica Santoso et all, 2020) Sistem operasi *Android* sendiri awal mulanya dikembangkan oleh *Android Inc*. Kemudian diambil alih oleh *Google* pada tahun 2005, *Google* sendiri menjadikannya sebagai sistem operasi yang bersifat "*Open Source*". Sistem operasi ini bisa didapatkan secara gratis. Bukan hanya untuk ponsel saja, tetapi sistem *Android* juga tersedia untuk perangkat lainnya.

Selain itu, *Android* juga memiliki komunitas pengembang aplikasi dengan jumlah yang sangat besar sehingga dapat memperluas fungsional sebuah perangkat dan pada umumnya ditulis dengan bahasa pemrograman Java. Beberapa faktor di ataslah yang berkontribusi terhadap perkembangan *Android* dan menjadikannya sebagai sistem operasi *smartphone* yang paling banyak digunakan di dunia. (Sulistiyorini & Listiadi, 2022)

Berikut merupakan logo sistem operasi *Android*:



Gambar 2. 4 Logo *Android*
Sumber : (Google, 2023)

2.1.3.2 *My Structured Query Language (MySql)*

My Structured Query Language (MySQL) adalah sistem manajemen basis data relasional (RDBMS) *open source* yang begitu populer. Ini adalah perangkat lunak *open source* yang digunakan untuk mengelola dan mengambil data dalam *database*. *MySQL* adalah salah satu RDBMS yang paling banyak digunakan di dunia dan digunakan dalam pengembangan web, perangkat lunak perusahaan, perangkat lunak *open source*, dan banyak lagi.(Ramadhan & Mukhaiyar, 2020)

Alasan para pengembang memilih untuk menggunakan *MySQL* sebagai server *database* mereka bukan hanya dari segi fungsionalitasnya saja, melainkan fleksibilitas dari *MySQL* sendiri yang mampu beradaptasi dengan beberapa bahasa pemrograman yang digunakan oleh para pengembang seperti *PHP*, *Java*, *C++* dan lain-lain, dengan menyediakan *plugin* dan *driver* khusus untuk setiap *plugin* nya. Selain itu, *MySQL* dipuji karena kesederhanaan sintaksisnya yang singkat dan mudah dipahami.(Dwi Praba & Safitri, n.d. 2020)

Berikut merupakan logo *MySql*:



Gambar 2. 5 Logo *MySQL*
Sumber : (Oracle, 2024)

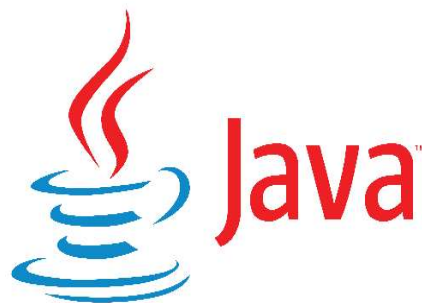
2.1.3.3 Java

Bahasa pemrograman *Java* pertama kali dikembangkan pada tahun 1991 oleh Sun Microsystems dengan proyek yang diberi nama “The Green Project”. Proyek yang melibatkan James Gosling, Patrick Naughton, dan Mike Sheridan ini bertujuan untuk menciptakan bahasa pemrograman baru yang lebih baik dibandingkan *C* dan *C++*. Setelah 18 bulan mengerjakan proyek tersebut, pengembang James Gosling akhirnya mampu mengembangkan bahasa pemrograman baru bernama *Oak*. Sayangnya, nama ini terdaftar sebagai merek dagang *Oak Technology*. (Sugandi et all, 2022)

Java adalah bahasa pemrograman yang dikembangkan pada tahun 1995 oleh *Sun Microsystems* (saat dimiliki oleh *Oracle Corporation*). *Java* dikembangkan dengan beberapa fitur penting yang menjadikannya unik dan sangat populer. *Java* terkenal dengan motonya "Tulis sekali, jalankan di mana saja". Artinya, aplikasi apa pun yang ditulis dalam *Java* dapat langsung dijalankan di *platform* berbeda tanpa harus mengkompilasi ulang kodenya. (Fajri Ali, 2019)

Berdasarkan penelitian (Zain Arif et al., 2022) *Java* juga memiliki berbagai versi yang dirilis sejak awal, dengan yang terbaru. *Java* juga memiliki sejumlah alat pengembangan, seperti Java Development Kit (JDK) dan Integrated Development Environments (IDEs) seperti Eclipse, IntelliJ IDEA, dan NetBeans, yang memudahkan pengembangan perangkat lunak dengan bahasa ini.

Berikut merupakan logo bahasa pemrograman *Java*:



Gambar 2. 6 Logo *Java*
Sumber : (Java, 2024)

2.1.3.4 Visual Studio Code

Visual Studio Code atau yang biasa dikenal dengan nama *VS Code* merupakan sebuah perangkat lunak (*software*) yang dikembangkan oleh *Microsoft*. *VS Code* merupakan salah satu dari sekian banyak editor kode yang digunakan oleh para pengembang perangkat lunak untuk menulis, mengedit, dan juga mengelola kode program. *VS Code* merupakan editor kode gratis yang dapat berjalan di berbagai perangkat desktop dengan sistem operasi *Windows*, *Linux*, dan *MacOS*.(Romzi & Kurniawan, 2020)

Visual Code adalah program pengeditan yang kompleks, dan mudah digunakan. *VS Code* juga dapat digunakan untuk menghasilkan dan memodifikasi *source code* untuk berbagai bahasa pemrograman. Misalnya seperti *JavaScript*, *TypeScript* dan *Node.js*.(Ra'uf Al Farras et al., 2023)

Berikut merupakan logo dari *Visual Studio Code*:



Gambar 2. 7 Logo *Visual Studio Code*
Sumber : (Visual Studio Code, 2024)

2.1.3.5 *Android studio*

Android Studio adalah *tools Integrated Development Environment (IDE)* resmi dari kemitraan antara *Google* dan *Jetbrains*, yang dibuat khusus untuk pengembangan aplikasi *Android*. *IDE* ini dikembangkan oleh *Google* dan merupakan alat utama yang digunakan oleh pengembang *Android* untuk membuat, menguji, dan menerbitkan aplikasi *Android*.(Nasution et all, 2019)

Android Studio didukung oleh *IntelliJ IDEA*, sebuah *IDE* yang dikembangkan oleh *Jetbrains* yang juga mengembangkan program seperti *PHPStorm* dan *PyCharm*. Tentu saja sebagai *IDE*, *Android Studio* memiliki komponen yang komprehensif termasuk *source code editor*, *compiler*, hingga *debugger*.(Roihan et all, 2019)

Berikut merupakan gambar *android studio*:



Gambar 2. 8 Logo *Android Studio*
Sumber : (Google, 2024)

2.1.3.6 XAMPP

XAMPP adalah perangkat lunak *server web* asli untuk membuat situs *web*, aplikasi, dan *database offline*. *XAMPP* merupakan singkatan dari "X" yang menunjukkan bahwa *XAMPP* bersifat lintas *platform (Cross-Platform)*, "*Apache*" sebagai *server web*, "*MySQL*" sebagai sistem manajemen basis data (*Database*), "*PHP*" sebagai bahasa pemrograman *server*, dan "*Perl*" sebagai bahasa pemrograman skrip. (Suryana, 2020)

XAMPP merupakan perangkat lunak *open-source* yang dikembangkan oleh *Apache Friends*, dengan tujuan menyediakan paket perangkat lunak yang lengkap untuk membuat dan mengelola lingkungan pengembangan *web* lokal. *XAMPP* adalah perangkat lunak sumber terbuka gratis dan dapat diinstal pada berbagai platform seperti *Windows*, *Linux* dan *OS X*.(Subekhan & Wati, 2020)

Berikut merupakan logo *XAMPP*:



Gambar 2. 9 Logo *XAMPP*
Sumber : (Apache Friends, 2024)

2.1.3.7 FLUTTER

Flutter adalah jenis *framework* yang digunakan untuk membantu pengembang membuat aplikasi seluler *multiplatform*. *Flutter* sendiri telah digunakan secara luas untuk membangun aplikasi seluler bagi perusahaan besar seperti *Alibaba*, *Ebay*, dan *BMW*. Tidak

heran jika banyak pengembang perangkat lunak yang tertarik untuk mencoba menggunakan *Flutter*.(Hakim et all, 2019)

Flutter adalah kerangka pengembangan perangkat lunak (*framework*) *open source* yang dikembangkan oleh *Google*. Ini digunakan untuk membangun aplikasi seluler (*mobile apps*) yang kaya dan menarik minat para pengguna untuk berbagai *platform*, termasuk *Android*, *iOS*, *web*, dan *desktop*, semuanya dari satu kode sumber (*single codebase*). *Flutter* memiliki dua komponen penting, yaitu, *Software Development Kit (SDK)* dan juga *framework user interface*.(Sumarni, 2022.)

Berikut merupakan logo *Flutter*:



Gambar 2. 10 Logo *Flutter*
Sumber : (Flutter, 2024)

2.1.3.8 *StarUML*

StarUML merupakan perangkat lunak pemodelan yang berorientasi pada objek yang mendukung *UML (Unified Modeling Language)*. *StarUML* unggul dalam hal kustomisasi dilingkungan kerja bagi para pengguna. *StarUML* sendiri menasbihkan diri sebagai salah satu pemodelan perangkat lunak yang fleksibel dan memaksimalkan produktifitas perangkat lunak pengembang.(Betesda, 2020)

Berikut merupakan logo *StarUML*:



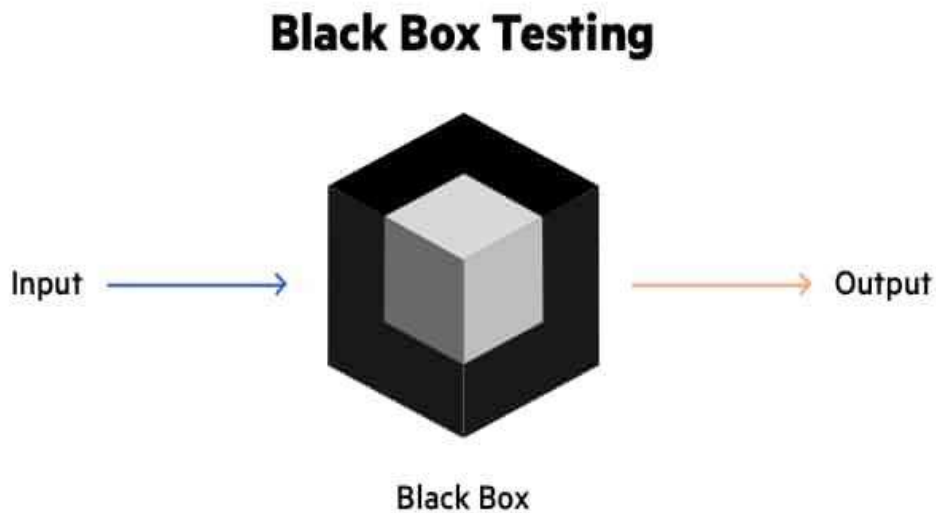
Gambar 2. 11 Logo *StarUML*
Sumber : (StarUML.io, 2024)

2.1.3.9 Black Box

Black box testing atau dikenal juga dengan *behavioral testing* merupakan pengujian yang dilakukan untuk mengamati hasil *input* dan *output* perangkat lunak tanpa mengetahui struktur kode perangkat lunak tersebut. Pengujian ini dilakukan pada akhir pengembangan perangkat lunak untuk mengetahui apakah perangkat lunak dapat berfungsi dengan baik atau tidak.(Alarie et all, 2021)

Tujuan dari black box testing adalah untuk memastikan bahwa perangkat lunak berfungsi dengan benar sesuai dengan spesifikasi fungsional yang telah ditentukan. Dalam pengujian ini, pengujian dilakukan tanpa pengetahuan mendalam tentang struktur kode sumber atau desain internal perangkat lunak. Pengujian black box berfokus pada masukan yang diberikan ke sistem dan keluaran yang dihasilkan, serta bagaimana sistem merespons berbagai situasi atau kondisi yang mungkin terjadi.(Febrian et all, 2020)

Berikut merupakan logo *Black Box*:



Gambar 2. 12 Logo *Black Box*
Sumber : (Blackbox, 2024)

2.1.4 Software Development Life Cycle (SDLC)

Software Development Life Cycle (SDLC) atau siklus hidup pengembangan perangkat lunak merupakan pendekatan sistematis untuk pengembangan perangkat lunak. *SDLC* menjelaskan tentang langkah-langkah atau tahapan yang harus diikuti oleh tim pengembangan perangkat lunak mulai dari pembuatan ide hingga pengiriman hingga pemeliharaan produk perangkat lunak. Setiap fase *SDLC* mempunyai tujuan dan aktivitas spesifik, yang membantu dalam mengelola risiko dan menjamin kualitas produk yang dihasilkan. (Apriliah, Subekti, & Haryati, 2019)

Didalam penelitian ini peneliti sendiri menggunakan *SDLC* model *waterfall* dengan berbagai pertimbangan, salah satunya karena model *waterfall* merupakan model yang paling mudah dan juga sesuai dengan kebutuhan objek penelitian yang dilakukan.

Berikut ini merupakan gambar tahapan tahapan *Software Development Life Cycle (SDLC)*:



Gambar 2. 13 Gambar *SDLC*
Sumber : (Amazon Web Services, 2024)

2.1.4.1 Model Waterfall

Model *waterfall* (air terjun) adalah salah satu model pengembangan perangkat lunak yang paling sederhana dan tertua. Dalam model ini, pengembangan perangkat lunak dibagi menjadi serangkaian tahapan linier, dan setiap tahapan bergantung pada hasil tahap sebelumnya. Misalnya air mengalir dari satu lantai ke lantai lain, sehingga disebut dengan *waterfall*. (Ardiansah, Hidayatullah, 2023)

Model *waterfall* menyediakan pendekatan alur hidup perangkat lunak secara *sequencial* atau lebih terorganisir. Berikut ini merupakan tahapan utama dalam metode *waterfall*:

1. Perencanaan

Dalam tahap ini penulis melakukan perencanaan terlebih dahulu tentang sebuah sistem yang akan dibuat.

2. Analisis

Kemudian melakukan analisis tentang sistem yang dibutuhkan dilapangan, agar dapat beroperasi dengan maksimal.

3. Desain

Setelah dilakukannya analisis, selanjutnya menentukan desain yang cocok dan sesuai dengan sistem.

4. Implementasi

Implementasi sendiri adalah tahap yang dilakukan saat sistem sedang dibangun untuk mengetahui bagaimana fungsi dari sistem tersebut berjalan.

5. Pengujian

Untuk pengujian sendiri, sistem yang sudah dirancang diuji terlebih dahulu dengan metode *Black Box*, dan kemudian di ujikan ke karyawan langsung.

6. Pemeliharaan

Pemeliharaan sendiri nantinya akan dilakukan untuk melakukan pembaharuan daripada fitur maupun fungsinya.

2.1.4.2 Unified Modeling Language (UML)

Unified Modelling Language (UML) merupakan suatu metode dalam pemodelan secara visual yang digunakan sebagai sarana perancangan sistem berorientasi objek. Awalnya, *Unified Modelling Language (UML)* diciptakan oleh Object Management Group dengan versi awal 1.0 pada bulan Januari 1997. *Unified Modeling Language (UML)* memiliki standar khusus agar dapat menspesifikasikan serta membangun sistem perangkat lunak, dan juga mempunyai alat pemodelan berbasis visual yang berguna untuk membuat diagram dan model yang mewakili sistem *software*. (Nistrina & Sahidah, 2022)

Unified Modeling Language (UML) adalah bahasa standar yang sangat membantu *software developer, engineer, dan stakeholder* lain agar dapat berkomunikasi satu sama lain dan juga berkolaborasi dengan efektif selama proses pengembangan *software*. (Sonata, 2019)

Unified Modeling Language (UML) sendiri mempunyai standar dalam penulisan sebuah sistem yang disebut *blueprint*, yang meliputi konsep bisnis, penulisan sub-sub dalam bahasa pemrograman, *database*, dan komponen lain yang diperlukan oleh sistem. (Wira, Putra, & Andriani, 2019)

Unified Modeling Language (UML) mempunyai beberapa diagram dan notasi yang bisa dipakai untuk memodelkan beberapa macam aspek sistem, struktur, perilaku dan lain-lain. Dengan menggunakan *UML*, *developer dan engineer* bisa membuat representasi sistem dengan jelas dan ringkas sehingga dapat memecahkan masalah.

Secara logika, terciptanya *Unified Modeling Language (UML)* bersamaan dengan konsep yang telah ada yaitu konsep permodelan *Object Oriented (OO)*, karena konsep ini mencontohkan sistem seperti kehidupan nyata dimana banyak didominasi oleh obyek dan maupun simbol-simbol yang cukup spesifik maka *Object Oriented (OO)* memiliki proses standard dan bersifat independen. (Rinaldi, 2019)

Adapun beberapa model dari *UML*, berikut ini merupakan pendekatan dengan *Use Case Diagram, Activity Diagram, Class Diagram, dan Sequence Diagram*.

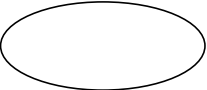
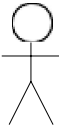


2.1.4.2.1 Use Case Diagram

Use Case Diagram merupakan salah satu jenis diagram yang ada di dalam *Unified Modeling Language (UML)* yang berfungsi untuk menggambarkan interaksi antara sebuah sistem perangkat lunak dengan aktor-aktor eksternal yang berinteraksi dengan sistem tersebut. *Use Case Diagram* membantu dalam merencanakan, memahami, dan mengkomunikasikan fungsionalitas sistem dengan menunjukkan cara pengguna atau aktor-aktor lainnya berinteraksi dengan sistem tersebut.

Use Case Diagram memiliki beberapa komponen penting diantaranya, untuk dapat memahami dari tiap-tiap komponen penting di dalamnya, dibawah ini merupakan contoh dari *use case diagram* beserta simbol dan penjelasannya untuk mempermudah kita dalam memahami arti dan maksud yang di tuju.

Berikut ini penjelasan tabel dan juga simbol-simbol yang ada di dalam *Use case diagram*:

Tabel 2.1 Simbol-simbol *use case diagram*

Simbol	Nama	Keterangan
	<i>Use Case</i>	Membantu dalam mendefinisikan kebutuhan sistem Batas sistem yang ditempatkan di dalam
	<i>Actor</i>	Mewakili orang atau alat untuk berkomunikasi dengan sistem
	<i>Association</i>	Membantu dalam menggambarkan hubungan atau interaksi antara <i>actor</i> dengan <i>use case</i>
	<i>Generalization</i>	Membantu <i>use case</i> umum untuk berkomunikasi dengan <i>use case</i> khusus
<<include>>	<i>Include</i>	Relasi tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan bantuan untuk menjalankan fungsinya
<<extend>>	<i>Extend</i>	Menunjukkan bahwa suatu <i>use case</i> adalah tambahan fungsional dari <i>use case</i> yang lain



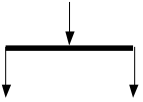
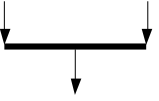
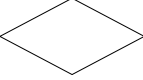
Sumber : (Susilowati & Pakusadewa, 2023)


2.1.4.2.2 Activity Diagram

Activity diagram atau diagram aktivitas adalah jenis diagram yang digunakan untuk menggambarkan alur kerja atau proses dalam suatu sistem. Diagram ini menunjukkan langkah-langkah yang dilakukan dalam suatu aktivitas dan hubungan antara langkah-langkah tersebut. (Susilowati & Pakusadewa, 2023)

Dalam *activity diagram*, setiap langkahnya akan gambarkan dengan bentuk geometris, seperti persegi panjang atau lingkaran, biasa disebut dengan simpul. Node-node ini dihubungkan oleh garis yang disebut dengan garis tepi yang mewakili alur kerja atau urutan langkah. Berikut ini merupakan tabel dan simbol-simbol yang ada pada *activity diagram* beserta penjelasan tentang fungsi dari masing-masing simbol nya:

Tabel 2.2 Simbol-simbol *activity diagram*

Simbol	Nama	Keterangan
	<i>Start point</i>	<i>Start point</i> merupakan simbol untuk memulai sebuah aktifitas <i>Start point</i> hanya boleh digunakan 1 simbol di setiap aktifitas
	<i>Activity</i>	Menggambarkan sebuah aktifitas yang biasanya diawali dengan kata kerja Hanya memiliki satu aktifitas yaitu alur masuk dan keluar
	<i>Fork</i>	Satu aliran yang dikerjakan bersamaan
	<i>Join</i>	Aliran yang disatukan untuk kemudian melanjutkan aktifitas
	<i>Decision point</i>	Asosiasi percabangan yang menunjukkan jika ada aktifitas lebih dari satu

<i>Swimlane</i>	<i>Swimlane</i>	Memisahkan organisasi bisni/ <i>actor</i> yang menunjukkan adanya aktifitas
	<i>End point</i>	Status akhir yang menunjukkan akhir dari sebuah aktifitas dalam sebuah <i>diagram</i>

Sumber : (Susilowati & Pakusadewa, 2023)

2.1.4.2.3 Class Diagram

Class diagram merupakan salah satu jenis diagram yang digunakan dalam pemodelan objek dalam pengembangan perangkat lunak. *Class diagram* memberikan pandangan tingkat tinggi tentang bagaimana objek dalam sistem berinteraksi dan berkomunikasi satu sama lain. *Class diagram* juga bisa disebut dengan diagram struktur karena menggambarkan berbagai hal yang diperlukan dalam sistem untuk di modelkan dengan berbagai komponen. (Susilowati & Pakusadewa, 2023)





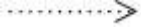

Class diagram mempunyai fungsi utama yaitu menggambarkan struktur dari sistem pemrograman, namun ada beberapa fungsi lain dari *class diagram* seperti:

1. Menggambarkan struktur kelas
2. Mengidentifikasi hubungan
3. Dapat dijadikan *business analyst* untuk membuat model dari sebuah bisnis.

Berikut ini merupakan tabel dan simbol-simbol yang ada pada *activity diagram* beserta penjelasan tentang fungsi dari masing-masing simbol nya:

Tabel 2.3 Simbol-simbol *class diagram*

Simbol		Keterangan			
<table border="1" style="width: 100%;"> <tr> <td>Nama_Kelas</td> </tr> <tr> <td>+atribut</td> </tr> <tr> <td>+operasi()</td> </tr> </table>	Nama_Kelas	+atribut	+operasi()	<i>Class</i>	Struktur sistem yang ada pada kelas
Nama_Kelas					
+atribut					
+operasi()					

	Antarmuka/ <i>interface</i>	Konsep <i>interface</i> di dalam pemrograman berorientasi objek
	Asosiasi / <i>association</i>	Relasi antar kelas dengan makna umum, biasanya disertai <i>multiplicity</i>
	Asosiasi berarah / <i>directed association</i>	Relasi antar kelas atau makna dengan relasi yang lain, biasanya terdapat <i>multiplicity</i> .
	<i>Generalization</i>	Generalisasi antar kelas dengan makna khusus
	<i>Dependency</i>	Kebergantungan relasi antar kelas
	<i>Agregation</i>	Relasi antar kelas dengan semua bagian (<i>whole-part</i>)







Sumber : (Susilowati & Pakusadewa, 2023)

2.1.4.2.4 Sequence Diagram

Sequence diagram atau diagram urutan adalah salah satu jenis diagram yang ada didalam *Unified Modeling Language (UML)* yang digunakan untuk menggambarkan dan menjelaskan bagaimana cara kerja objek-objek atau entitas-entitas didalam suatu sistem. Selain itu *sequence diagram* menggambarkan urutan waktu dari pesan atau panggilan metode antara satu objek, yang membantu dalam memahami interaksi dan alur eksekusi antara komponen-komponen dalam sistem perangkat lunak atau yang diurutkan dari arah kiri ke arah kanan. Diagram ini sendiri memiliki dua dimensi, yaitu vertikal dan horizontal. Dimana vertikal adalah dimensi yang menunjukkan waktu sementara horizontal adalah yang menunjukkan sebuah objek.(Susilowati & Pakusadewa, 2023)

Berikut ini merupakan tabel dan simbol-simbol yang ada pada *sequence diagram* beserta penjelasan tentang fungsi dari masing-masing simbol nya:

Tabel 2. 4 Simbol-simbol *sequence diagram*

Simbol	Nama	Keterangan
	<i>Actor</i>	Menjelaskan orang yang berinteraksi dengan sistem
	<i>Boundary</i>	Sebagai penghubung antara <i>actor</i> dengan sistem
	<i>Control</i>	Perilaku suatu sistem yang telah di <i>setting</i> cara kerjanya oleh sistem lain
	<i>Entity</i>	Menggambarkan struktur data pada sebuah sistem
	<i>Activation</i>	Menggambarkan kondisi aktifitas suatu objek
	<i>Message</i>	Simbol yang menggambarkan pengiriman pesan

Sumber : (Susilowati & Pakusadewa, 2023)

2.2 Penelitian Terdahulu

Adapaun penelitian terdahulu yang digunakan sebagai sumber referensi dalam pembuatan laporan dan pengembangan aplikasi berbasis *android* dalam penelitian ini, diantaranya :

1. Dalam penelitian (Alfisyahri & Simanjuntak, 2020) yang berjudul **“Aplikasi pembelajaran bahasa latin tumbuh-tumbuhan berbasis android”** ISSN: 2715-6265 VOL.0 3 NO. 02 (2020). Penelitian ini membahas tentang media pembelajaran berbasis *android* dengan tujuan membantu para siswa untuk mengetahui nama-nama latin dari tumbuh-tumbuhan, yang sudah mereka pelajari saat duduk di bangku sekolah dasar (SD) dan sekolah menengah pertama (SMP) dengan menggunakan buku. Proses manual sendiri akan membosankan bagi anak-anak dan remaja. Dengan adanya sistem pembelajaran menggunakan aplikasi berbasis *android* akan memberikan manfaat yang baik bagi anak-anak saat menggunakan *smartphone*.
2. Dalam penelitian (Prasetyo et all, 2020) yang berjudul **“Penerapan aplikasi android pengenalan huruf Hijaiyah metode Waterfall pada PAUD AlFina”** E-ISSN:2581-0588 Vol.09 N0. 03(2020)412-419 DOI:10.32736/sisfokom.v9i3.883. Penelitian ini membahas tentang penerapan aplikasi berbasis *android* dalam membantu anak usia dini untuk mempelajari huruf hijaiyah dengan metode yang lebih menarik, karena huruf hijaiyah ditampilkan dengan visualisasi dalam bentuk gambar, teks dan suara yang menarik minat anak usia dini dalam belajar. Dengan metode ini minat belajar anak usia dini jadi lebih bersemangat dibandingkan dengan menggunakan metode manual dan lebih efisien bagi tenaga pendidik itu sendiri.
3. Dalam penelitian (Jannah et all, 2021) yang berjudul **“Perancangan media pembelajaran sejarah menggunakan mit app inventor berbasis android di SMKN 2 Panyabungan”** E-ISSN: 2798-5210 Volume 1 No. 7 November 2021.

Penelitian ini dirancang menggunakan aplikasi berbasis *android* agar mempermudah siswa dan guru dalam melakukan aktifitas belajar mengajar secara daring demi menghidupkan suasana kelas agar lebih kompetitif dan menarik.

4. Dalam penelitian (Tampubolon & Maslan, 2022) yang berjudul **“Perancangan aplikasi e-galon berbasis *android*”** ISSN: 2715-6265 VOL. 06 NO. 05 (2022). Penelitian yang terkait saat ini yaitu sebuah penelitian yang bertujuan untuk mempermudah pelanggan dalam melakukan pemesanan air galon, yang mana sebelum adanya penelitian ini pemesanan air galon dilakukan secara manual yang mengharuskan pelanggan datang ke toko, saat adanya penelitian ini dengan berbasis *android* sangat membantu para pelanggan dan memajukan UMKM.
5. Dalam penelitian (Saiful Isnandar, Islam Mahdi, 2023) yang berjudul **“Pemodelan sistem informasi sekolah berbasis web mobile pada sekolah SMK Darussalam”** E-ISSN: 2961-7189 Vol. 3, No. 1, Januari 2023. Penelitian ini terkait sistem informasi yang digunakan oleh SMK Darussalam masih menggunakan sistem manual, baik dalam hal pendataan maupun pemberian informasi tentang SMK Darussalam sendiri. Dimana masih menggunakan media-media tercetak seperti brosur, pamflet, spanduk, ataupun informasi yang disebarkan dari mulut ke mulut, sehingga proses pemberian informasi tersebut tidak efisien dalam hal kecepatan waktu saat mencari data ataupun informasi.
6. Dalam penelitian (Alfajri, Faizah, & WP, 2023) yang berjudul **“Aplikasi sistem persediaan barang gudang pt. berkah pena ilmu menggunakan *anroid studio* dan metode *rapid application development*”** E-ISSN: 2723-7079 Vol 4 No 1, Januari (2023). Penelitian ini membahas tentang sinkronisasi stok barang antara

gudang dengan toko *online* nya, dimana sering terjadi salah komunikasi antara karyawan gudang dengan *admin* toko tentang stok barang di gudang. Hal ini terjadi saat seorang pelanggan membutuhkan suatu barang dalam jumlah banyak sementara karyawan gudang harus melakukan pengecekan secara manual, dari segi waktu tentu tidak efisien dan hasilnya tidak akurat. Dengan adanya penelitian ini *monitoring* stok barang dilakukan dengan lebih efisien dan akurat karena dilakukan dengan menggunakan sistem berbasis *android*.

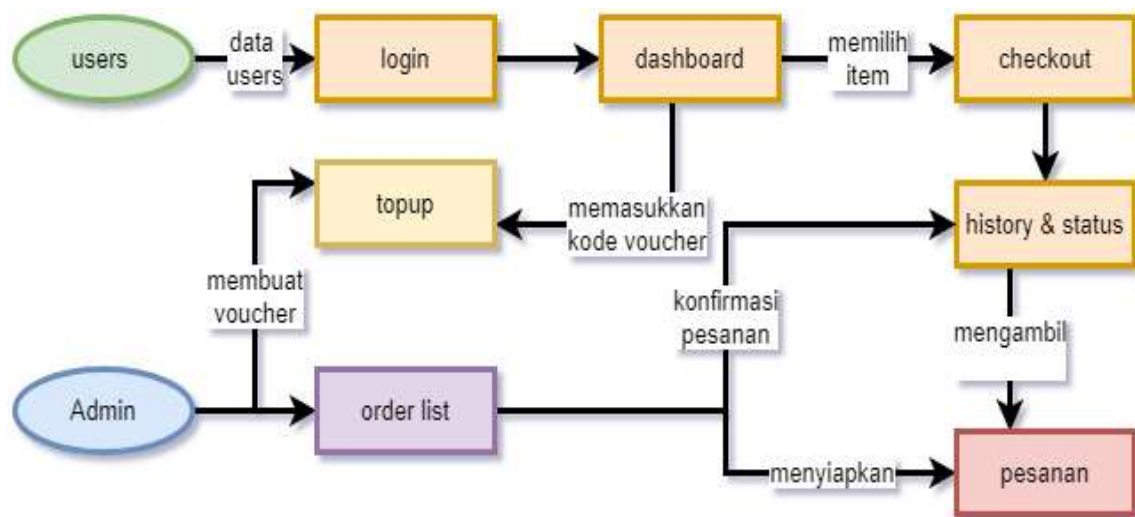
7. Dalam penelitian (Hussudur & Pangaribuan, 2023) yang berjudul **“Implementasi augmented reality sebagai media pengenalan aksara arab melayu berbasis *android*”** ISSN: 2715-6265 VOL. 09 NO. 02 (2023). Penelitian ini bertujuan untuk mengenalkan bahasa arab melayu dengan tampilan yang lebih menarik yaitu dengan penggunaan augmented reallity berbasis *android*. Hal ini bertujuan untuk mengantisipasi menurunnya minat baca dari anak-anak tentang bahasa arab melayu sehingga bahasa arab melayu tetap lesatari sampai ke generasi seterusnya.
8. Dalam penelitian (Umami et all, 2023) yang berjudul **“Designing a mobile application to assist micro entrepreneurs in understanding the food business legality process”** E-ISSN: 2502-3357 Vol. 9 No. 1(2023): January 68-85 DOI: 10.26594/register.v9i1.3061. Penelitian bertujuan untuk mengatasi kesenjangan pembinaan dan pengembangan usaha mikro terbatas menjadi lebih fleksibel dan efektif, yaitu dengan mengembangkan aplikasi *mobile* yang dapat menunjang bisnis para pengusaha. Dengan harapan dapat membantu pertumbuhan ekonomi baik untuk pemerintahan maupun para pengusaha dan lingkungan sekitarnya.

9. Dalam penelitian (Kusnadi et all, 2023) yang berjudul **“Designing halal product traceability system using UML and integration of blockchain with ERP”** E-ISSN: 2502-3357 Vol. 9 No. 1(2023): January 29-41 DOI: 10.26594/register.v9i1.3045. Penelitian ini membahas tentang bagaimana para pengembang aplikasi berusaha mengembangkan sebuah aplikasi untuk menunjang kegiatan transaksi dari produsen ke konsumen untuk menjamin status halal dari produk mereka. Dengan aplikasi ini sebuah informasi jadi lebih transparan, keamanan data lebih terjamin, serta efektifitas waktu tercapai. Sehingga produsen dapat memenuhi banyaknya permintaan dari konsumen dengan memonitoring lewat sistem.
10. Dalam penelitian (Nugroho, Izzah, & Eliyen, 2023) yang berjudul **“Pengembangan aplikasi seluler untuk memecahkan masalah *routing* kendaraan dalam pemasaran atau perencanaan perjalanan”** ISSN Media Electronic: 2580-0760 Vol. 7 No.1 (2023)27-33 DOI: 10.29207/resti.v7il.4552. Penelitian ini membahas tentang optimalisasi rute kendaraan dalam bidang jasa pengiriman yang sering mengalami masalah dalam rute perjalanan maupun jumlah biaya. Dengan mengembangkan aplikasi *mobile* akan mempermudah dalam menghitung biaya yang dikenakan kepada pelanggan dan tahu jarak dari rute pengiriman yang akan disematkan dalam aplikasi *mobile* ini nantinya.

2.3 Kerangka Pemikiran

Metode kerangka pemikiran yaitu menjelaskan tentang bagaimana alur ataupun sebuah pemikiran dari adanya penelitian. Kerangka ini memiliki konsep yang menggambarkan ataupun penghubung antara satu variabel dengan variabel yang lainnya.

Beberapa latar belakang yang di dapat dan di identifikasikan beberapa masalah adalah tidak terkontrolnya antrian yang terjadi di koperasi, khususnya saat di jam masuk dan jam istirahat karyawan. Yang bisa saja menyebabkan terjadinya kecurangan ataupun kesalahan pada *admin* pada saat melakukan pencatatan belanja karyawan. Hal itu terjadi karena transaksi jual beli yang masih manual dan memakan waktu yang cukup lama. Dan dengan adanya penelitian ini tentunya akan mempermudah kerja admin koperasi dan juga para karyawan sebagai pelanggan. Berikut ini merupakan bagan dari kerangka pemikiran penelitian ini:



Gambar 2. 14 Kerangka Pemikiran
Sumber : (Data Penelitian, 2024)

Kerangka pemikiran diatas menggambarkan bahwa transaksi dari koperasi karyawan yang sebelumnya masih menggunakan cara manual yang sangat memakan waktu di tengah meningkatnya target produksi. Dengan adanya *digitalisasi* ini salah satu tujuannya yaitu mengatasi masalah yang terjadi dilapangan tentang antrian yang sering terjadi. Sementara untuk mendapatkan informasi lain dari koperasi adalah dengan melakukan *interview* langsung dengan admin koperasi dan juga beberapa karyawan, serta mencari dokumen

pendukung seperti halnya buku ataupun jurnal referensi yang memiliki keterkaitan tentang koperasi karyawan. Langkah yang kedua yaitu aplikasi akan diterapkan dengan menggunakan sistem operasi berbasis *android (smartphone)*.