

BAB II

TINJAUAN PUSTAKA

2.1 Teori Dasar

2.1.1 Metode Pengembangan Prototype

Metode prototype adalah metode pengembangan aplikasi yang menggunakan prototype atau bentuk asli untuk menggambarkan model fisik aplikasi sekaligus menggambarkan versi awal dari aplikasi yang sedang dikembangkan. Tujuan pembuatan prototype selain untuk menunjukkan model fisik dan sistem kerja aplikasi kepada klien, tujuan lainnya yang tidak kalah penting adalah untuk mengumpulkan informasi yang sebanyak-banyaknya dari pengguna terkait kebutuhan-kebutuhan yang diharapkan oleh pengguna.

Pada hasil penelitiannya Dwi Purnomo (2017, p. 55) menyebutkan dari sudut pandang penyajiannya metodologi yang digunakan dalam Prototyping memiliki 4 (empat) metodologi utama, sebagai berikut :

1. Illustrated, prototype hanya menampilkan contoh tampilan layar.
2. Simulated, prototype dapat mensimulasikan sebagian atau beberapa alur sistem dengan tidak menggunakan data asli (dummy data)
3. Functional, prototype dapat mensimulasikan beberapa atau sebagian alur sistem yang sebenarnya dan dengan menggunakan data asli (real data)
4. Evolutionary, prototype ini biasanya sudah fully functional, meskipun masih berupa prototype tapi sudah dapat digunakan karena sudah menjadi bagian dari sistem.

Artikel BSI Today (2022 : 33) mengungkapkan tahapan-tahapan penerapan metode Prototype, sebagai berikut :

1. Mengumpulkan Informasi dan Analisis Kebutuhan
Sebelum pembuatan prototype, dilakukan proses pengumpulan informasi kebutuhan sistem dan didiskusikan bersama pengguna dan didefinisikan secara rinci yang kemudian dianalisa.
2. Desain Cepat
Proses pembuatan desain sederhana berdasarkan hasil analisa kebutuhan pengguna yang berguna untuk memberikan gambaran sistem yang akan dibuat.
3. Membangun Prototype
Setelah desain cepat yang diajukan sudah disetujui oleh pengguna kemudian desain diserahkan ke tim pengembang (programmer) untuk dilakukan proses pengembangan prototype aplikasi.
4. Evaluasi Pengguna
Pada tahap ini, prototype aplikasi yang telah dibangun akan dipresentasikan ke hadapan pengguna untuk memberikan kesempatan kepada pengguna memberikan komentar dan masukan terkait prototype aplikasi yang telah dibuat.
5. Perbaiki Prototype
Setelah prototype menjalani proses evaluasi dan tidak menghasilkan komentar dan masukan dari pengguna maka aplikasi bisa dikembangkan dan berlanjut ke tahap berikutnya. Apabila saat proses evaluasi menghasilkan catatan dan masukan dari pengguna maka harus dilakukan perbaikan sampai mendapatkan persetujuan dari pengguna.
6. Implementasi Produk dan Pemeliharaan
Pada tahap ini, pengembang akan melakukan pengembangan aplikasi berdasarkan model prototype akhir dan kemudian akan diuji dan diserahkan kepada pengguna (klien) dan disertai dengan fase pemeliharaan untuk memastikan sistem yang

dikembangkan berjalan baik sesuai dengan yang diharapkan oleh pengguna.

Pada tahapan *Desain Cepat (Quick Design)*, penulis akan menggunakan IDE yang telah mendukung *Rapid Application Development (RAD)*. Untuk perancangan desain aplikasi berbasis mobile, penulis akan menggunakan Android Studio IDE. Sedangkan untuk perancangan desain aplikasi berbasis desktop, penulis akan menggunakan Lazarus IDE yang telah mendukung RAD.

Untuk aplikasi yang akan dikembangkan untuk PT Synergy First Logistics saat penyajian presentasi produk prototype yang dihasilkan saat tahapan *Evaluasi Pengguna*, penulis akan memadukan metodologi *Illustrated* dan *Simulated*. Presentasi prototype aplikasi mobile penulis akan menggunakan metodologi *Simulated* dengan mensimulasikan produk prototype yang dihasilkan dihadapan pengguna.

Sedangkan untuk produk prototype aplikasi yang berbasis desktop, penulis akan menggunakan metodologi *Illustrated* dimana penulis akan menyajikan produk prototype yang dihasilkan hanya berupa tampilan gambar *mockup* aplikasi tersebut.

2.1.2 Definisi Presensi

Presensi berdasarkan Kamus Besar Bahasa Indonesia (KBBI) berarti kehadiran, dalam bahasa sehari-hari umumnya penggunaan kata presensi jarang sekali digunakan karena kebanyakan orang lebih banyak menggunakan penyebutan kata *absensi* yang menurut Kamus Besar Bahasa Indonesia, *Absen* berarti “tidak masuk” atau ketidakhadiran seseorang dalam menghadiri kegiatan atau aktivitas dalam sebuah sekolah, tempat kerja dan sebagainya. Menurut Feriandirza (2020 : 124) absensi

bisa diartikan dengan “tidak hadir”, kata absensi secara umum lebih banyak digunakan dibanding kata presensi hingga absensi bisa juga dikatakan sebagai kondisi ketidakhadiran sekaligus kehadiran seseorang yang terlibat pada suatu kegiatan di sebuah organisasi.

Tingkat kehadiran karyawan pada suatu perusahaan merupakan salah satu indikator kedisiplinan mendasar dan biasanya dijadikan salah satu faktor penilaian Key Performance Index (KPI) bagi karyawan. Kebutuhan akan perangkat yang digunakan untuk memantau kehadiran atau presensi karyawan merupakan salah satu kebutuhan utama bagi perusahaan. Penggunaan teknologi informasi salah satunya adalah penggunaan perangkat bergerak (mobile) yang difungsikan sebagai perangkat untuk memantau presensi karyawan.

Penggunaan perangkat mobile yang dilengkapi dengan aplikasi yang berfungsi sebagai perangkat presensi saat ini sangat cocok digunakan selain dapat digunakan di kantor, perangkat mobile yang umumnya memiliki fitur lokasi menggunakan GPS (Global Positioning System) yang dapat memantau lokasi pengguna maka perangkat mobile juga dapat digunakan untuk memantau presensi dan posisi karyawan yang bekerja diluar kantor.

Penerapan presensi secara online akan semakin meningkatkan produktifitas dan efisiensi waktu proses presensi. Penerapan sistem informasi presensi berbasis mobile ini sebenarnya bisa juga diterapkan bagi karyawan yang bekerja di kantor sebab presensi dengan perangkat mobile akan semakin mempermudah karyawan dalam melakukan presensi terutama menghindari antrian di mesin presensi yang disediakan perusahaan.

2.1.3 Sistem Operasi Android

Android adalah sistem operasi yang didesain untuk perangkat bergerak (mobile) dengan layar sentuh. Menurut Android.com, Android adalah sistem operasi yang tidak hanya difokuskan pada perangkat seluler tapi juga digunakan pada perangkat lainnya seperti jam, TV bahkan pada kendaraan mobil.

Berdasarkan artikel dimuat portal berita Kompas yang memuat sejarah Android disebutkan Android pertama kali dirilis ke publik pada bulan November tahun 2007 yang merupakan Android Versi 1.0. Pengembang awal sistem operasi Android adalah *Rich Miner, Nick Sears, Chris White* dan *Andy Rubin* yang kemudian mendirikan Android Inc pada tahun 2003. Sebelum digunakan sebagai sistem operasi perangkat telepon seluler, diawal pengembangannya tujuan awal dari pengembangan Android rancangannya ditujukan meningkatkan fitur sistem informasi kamera digital.

Android diambil alih (diakuisisi) oleh Google Inc pada bulan Agustus 2005, Android mulai dirancang dan versi 1.0 dirilis sebagai sistem operasi telepon seluler secara resmi pada tahun 2008. Uniknya untuk penamaan versi sistem operasi Android, sejak versi 1.5 Google mulai menggunakan nama-nama makanan manis yang berurutan sesuai abjad.

Saat ini Android sudah digunakan pada 2,5 miliar perangkat aktif diseluruh dunia. Android.com menyatakan bahwa keamanan perangkat berbasis Android memiliki tingkat keamanan yang sangat baik salah satunya adalah dengan fitur Google Play protect yang dapat melindungi perangkat mobile dari aplikasi yang dapat mengancam keamanan perangkat.

Perangkat mobile yang menggunakan sistem operasi Android merupakan perangkat yang bersifat terbuka yang dapat digunakan oleh pengembang aplikasi untuk melakukan pengembangan aplikasi dan bereksperimen dengan perangkat Android.

2.1.4 Bahasa Pemrograman Kotlin

Kotlin adalah sebuah bahasa pemrograman yang dikembangkan oleh JetBrains yaitu perusahaan yang juga merupakan pengembang IDE Android Studio. Pemrograman Kotlin secara resmi diumumkan oleh Google pada ajang acara Google I/O pada tahun 2017 yang merupakan ajang konferensi tahunan para pengembang perangkat lunak yang diselenggarakan oleh Google.

Menurut Giffari Munawar (2018, p. 320), Kotlin merupakan bahasa pemrograman berbasis Java Virtual Machine (JVM) untuk sistem operasi Android yang memadukan pemrograman object oriented dan pemrograman fungsional.

Munurut McLaughlin (2021, p. 3) bila Kotlin dibandingkan dengan Java, saat ini Kotlin telah menambahkan dan merubah beberapa fitur penting yang juga terdapat pada Java. Kotlin telah mendukung perluasan fungsi tanpa perlu melakukan *override* pada *parent class* secara keseluruhan dan telah menambahkan beberapa komponen fungsional yang digunakan dalam pemrograman seperti dukungan *lambda*, mendukung *Lazy Evaluation* dan peningkatan fitur lainnya.

Bahasa pemrograman Kotlin memiliki performa yang lebih baik dibanding Java. Hasil penelitian Niko Sumanda Sibarani, Ghifari Munawar dan Bambang Wisnuadhi dari Politeknik Negeri Bandung, ditemukan bahwa hasil pengukuran *CPU Usage*, *Memory Usage* dan

Execution Time antara Kotlin dan Java hasilnya bahwa Kotlin lebih unggul dari segi performa.

Berdasarkan keunggulan performa Kotlin diatas terlihat bahwa penggunaan resource yang digunakan oleh aplikasi yang berbasis Kotlin akan lebih ringan dan tidak memberatkan kinerja perangkat seluler, hal ini jelas sangat menguntungkan karena umur baterai perangkat seluler akan lebih hemat dan perangkat seluler bekerja lebih ringan.

2.1.5 Object Oriented Programming

Object oriented programming (OOP) atau pemrograman berorientasi obyek adalah suatu metode pemrograman yang berorientasi obyek dimana program yang dibuat merupakan gabungan dari program-program kecil yang merupakan obyek-obyek yang saling berkaitan yang disebut dengan Class (kelas).

Penerapan pemrograman berorientasi obyek ini sangat membantu dan mempermudah programmer dalam melakukan pengembangan aplikasi karena dengan metode ini pengembang tidak perlu menulis kode berulang-ulang.

Dalam pemrograman berorientasi obyek mengenal 4 (empat) prinsip dalam penggunaannya, antara lain :

- Encapsulation (Pengkapsulan)
Pengkapsulan adalah proses pengikatan data atau penyatuan data yang berasal dari metode yang berbeda yang dikapsulkan menjadi satu unit data tertentu.
- Inheritance (Pewarisan)
Pewarisan adalah proses pembentukan kelas baru (*subclass*) yang merupakan kelas turunan dari *parent class* dimana

dimana subclass memiliki fungsi-fungsi turunan dari parent class.

- Polymorphism (Banyak Bentuk)

Polymorphism adalah salah satu konsep dalam pemrograman berorientasi obyek yaitu kemampuan dalam pemrosesan data dalam banyak bentuk.

- Abstraction (Menyembunyikan Data)

Ini merupakan salah satu fitur pemrograman berorientasi obyek yang memberikan kemampuan dalam menyembunyikan detail suatu fungsi atau proses.

2.1.6 Definisi Database

Database atau basis data adalah sekumpulan data yang diatur dan dikelola dengan sedemikian rupa dengan aturan tertentu dan saling berhubungan serta saling berkaitan satu sama lain. Dengan pengelolaan data ini maka akan mempermudah pengguna dalam menyimpan, merubah, menghapus dan melakukan pencarian informasi.

Dalam artikelnya, Zaenal Mustofa (2021 : 20) merinci jenis-jenis database sebagai berikut :

1. Database Operasional

Database ini ini adalah Database dengan sistem kerja OLTP (On Line Transaction processing) yang berguna mengelola database secara dinamis yang dapat berubah secara langsung dan secara realtime. Jenis file yang digunakan sebagai media penyimpanan adalah yang berjenis JSON (JavaScript Object Notation) dan XML (Extensible Markup Language)

2. Database Warehouse

Database ini lebih banyak difungsikan sebagai penyimpan data pelaporan dan untuk kepentingan analisis data dan merupakan komponen utama dari Business Intelligence. Contoh dari database yang berjenis ini adalah Microsoft SQL Server.

3. Distributed Database

Database ini dalam penerapannya tidak tersimpan di dalam satu perangkat yang sama. Basis data bisa saja tersimpan di beberapa perangkat yang berbeda dan tersebar melalui jaringan yang saling berhubungan. Contoh dari database yang berjenis ini adalah Microsoft Access.

4. Database Relational

Database relasional adalah database yang mengorganisir datanya berdasarkan model relasi datanya. Untuk bekerja dan mengelola data pada basis data ini umumnya menggunakan bahasa pemrograman Structured Query Language (SQL). Salah satu contoh dari database yang berjenis ini adalah MySQL.

2.1.7 Lazarus IDE

Lazarus IDE adalah Integrated Development Environment (IDE) *Cross-platform* yang dapat dijalankan di beberapa sistem operasi, antara lain sistem operasi Microsoft Windows, Linux, FreeBSD, MacOS dan lain-lain. IDE ini kompatibel dengan *Delphi IDE* dan berbasis bahasa pemrograman Free Pascal yang dirilis oleh team *Lazarus Project*.

Berdasarkan situs resminya lazarus.org, Lazarus Project adalah sebuah proyek yang awalnya didirikan oleh Cliff Baeseman, Shane Miller dan Michael A. Hess pada tahun 1999 dan Lazarus dikembangkan oleh team yang bernaung dalam Megido Project. Karena kurangnya dukungan maka di bentuklah komunitas Lazarus Project yang kemudian semakin menambah

dukungan dari para pengembang seluruh dunia dan terus dikembangkan hingga saat ini.

Lazarus adalah IDE yang sudah mendukung Rapid Application Development dimana pengguna sangat dimudahkan dalam membuat tampilan mockup yang dapat dibuat secara visual seperti halnya penggunaan Microsoft Visual Studio. Lazarus banyak disebut sebagai Delphi-like IDE karena memiliki banyak kesamaan komponen library yang digunakan dan struktur pembuatan kode programnya dengan Delphi IDE dan keduanya juga menggunakan Pascal sebagai basis bahasa pemrogramannya.

2.1.8 Object Oriented Analysis Design (OOAD)




Object Oriented Analysis Design adalah suatu metode analisis untuk memeriksa kebutuhan sistem yang ditemukan pada suatu ruang lingkup permasalahan dengan menggunakan sudut pandang kelas dan objek. Fiera Arnandya Rani, dkk dalam jurnal penelitiannya yang mengutip teori dari Kendall & Kendall (2011) menyatakan bahwa OOAD merupakan suatu metode analisis dan desain sistem berdasarkan konsep dunia nyata.

Analisa dan desain dengan pendekatan OOAD biasanya menggunakan pemodelan Unified Modelling Language (UML) yaitu suatu pemodelan dalam perancangan sistem berorientasi objek secara visual. UML merupakan suatu bahasa standar secara visual, yang digunakan dalam perancangan dan pendokumentasian sistem. Dibawah ini adalah penjelasan dari setiap symbol yang digunakan dalam pemodelan visual berupa diagram yang umum digunakan dalam UML :

1. Use Case Diagram

Use case diagram adalah diagram salah satu jenis diagram yang digunakan UML untuk menggambarkan interaksi antara sistem dengan







actor (Laksamana, 2022) dan mendeskripsikan jenis atau tipe interaksi actor sebagai pengguna sistem dengan sistem yang digunakan.

Simbol	Fungsi
 Actor	Simbol yang menggambarkan seorang pengguna yang berinteraksi dengan sistem
 Use case	Simbol yang menjelaskan fungsi dan aksi yang dilakukan aktor
 Association	Simbol yang menggambarkan hubungan antara use case dengan aktor
<<include>>	Simbol yang menggambarkan use case bagian dari use case yang lain
<<extend>>	Simbol yang menggambarkan suatu use case membutuhkan use case dalam satu kondisi tertentu terpenuhi

Gambar 2.1 Simbol Use Case Diagram

1. Activity Diagram






Activity diagram adalah diagram yang menggambarkan proses-proses yang terdapat pada suatu sistem berjalan yang di visualisasikan lewat model (Laksamana, 2022). Diagram ini menjelaskan jalannya program tanpa pemrograman yang menampilkan aktivitas yang sedang dikerjakan.

Simbol	Fungsi
 Initial/Start State	Simbol yang menggambar awal dari aktifitas
 Final/End State	Simbol yang menggambarkan akhir dari aktifitas
 Activity	Simbol yang menggambar suatu aktifitas
 Decision	Simbol yang menggambar suatu pilihan pengambilan keputusan
 Fork/Join/Synchronization	Simbol yang menggambarkan beberapa aliran kegiatan yang dilakukan secara bersama atau paralel
 Swimlane	Simbol yang digunakan untuk mengelompokkan aktifitas berdasarkan aktor

Gambar 2.2 Simbol Activity Diagram

2. Sequence Diagram

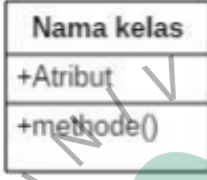


Diagram yang menggambarkan hubungan dan interaksi antara pengguna (aktor) dengan suatu sistem yang berjalan, diagram ini menjelaskan interaksi antar objek berdasarkan urutan dan menggambarkan tahapan apa saja yang harus dilakukan untuk dapat menghasilkan sesuatu dari sistem tersebut (Laksamana, 2022).

Simbol	Fungsi
 Actor	Simbol yang menggambarkan seorang pengguna yang berinteraksi dengan sistem
 Boundary	Simbol yang menggambarkan hubungan kegiatan yang akan dilakukan
 Control	Simbol yang menggambarkan pengatur penghubung antara boundary dengan tabel
 Entity	Simbol yang menggambarkan penyimpan data atau informasi
 Message	Simbol yang menggambarkan komunikasi antar objek

Gambar 2.3 Simbol Sequence Diagram

3. Class Diagram

Class diagram adalah diagram yang menggambarkan kelas-kelas yang terdiri dari paket-paket (Laksamana, 2022). Diagram ini juga yang akan menjelaskan hubungan antara sistem dengan database.

Simbol	Fungsi
 Class	Simbol yang menggambarkan kelas, yang terdiri dari nama kelas pada bagian atas, nama atribut pada bagian dan nama metode pada bagian bawahnya
 Association	Simbol yang menggambarkan hubungan yang bersifat statis antar kelas
 Agregation	simbol yang menggambarkan hubungan antar dan menjelaskan bahwa suatu kelas memiliki atribut kelas lain

Gambar 2.4 Simbol Class Diagram

2.2 Tinjauan Studi

Berbagai hasil penelitian mengenai penggunaan teknologi aplikasi mobile (perangkat bergerak) sudah banyak dilakukan. Beberapa penelitian akan dijadikan sebagai acuan dalam penelitian ini ditambah dengan beberapa literatur yang diambil dari beberapa buku yang berkaitan dengan teknologi dan bahasa pemrograman yang akan digunakan dalam pembangunan aplikasi yang akan dibangun. Berikut adalah beberapa penelitian terdahulu yang pernah dilakukan dan beberapa literatur yang berasal dari artikel yang dijadikan acuan penelitian ini :

1. Penelitian yang dilakukan oleh Niko Sumanda Sibarani, Ghifari Munawar dan Bambang Wisnuadhi dari Politeknik Negeri Bandung yang berjudul “Analisis Performa Aplikasi Android Pada Bahasa Pemrograman Java dan Kotlin” pada tahun 2018. Pada penelitian ini pengujian dilakukan menggunakan tiga parameter yang dijadikan acuan pengukuran yaitu *CPU Usage*, *Memory Usage* dan *Execution Time*. Munawar (2018 : 129) menyimpulkan bahwa berdasarkan hasil pengukuran secara keseluruhan ditemukan bahwa bahasa pemrograman Kotlin dalam hal performa masih lebih unggul dibandingkan bahasa pemrograman Java untuk digunakan dalam pengembangan aplikasi berbasis sistem operasi Android. Berdasarkan penelitian ini penulis memilih Bahasa pemrograman Kotlin untuk perancangan aplikasi pemantau aktivitas karyawan diluar kantor.
2. Penelitian yang dilakukan oleh Siti Nurazizah dari Fakultas Sains dan Teknologi Universitas Islam Negeri Sunan Ampel pada tahun 2018. Penelitian ini terfokus pada pengembangan dan perancangan mockup sebuah sistem informasi manajemen dimana penulis menjadikan hasil penelitian ini sebagai contoh penelitian untuk perancangan dan pengembangan sebuah mockup aplikasi seperti yang akan dibuat oleh penulis. Menurut Nurazizah (2018 : 9) dengan penerapan manajemen yang baik pada perusahaan maka perusahaan dapat mengatur dan mengelola aset mereka sesuai dengan standar. Selain penerapan manajemen yang baik, perusahaan harus didukung juga dengan adanya sumber daya manusia dan memiliki kemampuan berinvestasi dalam pengadaan sumber daya teknologi. Saat ini banyak organisasi telah memanfaatkan sumber daya teknologi yaitu berupa sistem informasi yang dimanfaatkan untuk menjalankan bisnisnya terutama oleh organisasi atau perusahaan yang telah maju dan berskala besar.
3. Hasil penelitian yang berjudul “*Perancangan Aplikasi Absensi Online Dengan Menggunakan Bahasa Pemrograman Kotlin*”, penelitian ini dilakukan oleh Arafat Febriandirza dari Universitas Muhammadiyah Prof. Dr. Hamka pada tahun 2020. Dalam penelitian ini, Febriandirza (2020 : 124)

menyatakan bahwa proses presensi yang dilakukan secara konvensional sudah tidak relevan untuk diterapkan saat ini karena memiliki beberapa kekurangan terutama yang berkaitan dengan efisiensi waktu dalam proses presensinya. Memberikan kemungkinan besar untuk terjadinya kesalahan dalam proses penyimpanan dan pengumpulan datanya yang disebabkan oleh *human error* dan rentan terjadinya kecurangan dalam proses presensinya. Oleh sebab itu, penggunaan sistem presensi yang dikembangkan dengan bahasa pemrograman Kotlin sangatlah dibutuhkan untuk memperbaharui proses presensi konvensional menjadi sistem presensi realtime menggunakan aplikasi berbasis mobile. Penelitian ini menyimpulkan bahwa pengembangan aplikasi presensi online menggunakan bahasa pemrograman Kotlin merupakan solusi terbaik dibandingkan menggunakan presensi secara konvensional atau manual.

4. Artikel yang berkaitan dengan aktivitas “Work From Home” (WFH – bekerja dari rumah). Artikel yang berjudul “Bekerja dari Rumah (Work From Home) Dari Sudut Pandang Unit Kepatuhan Internal” ini disusun oleh Tantri Dewayani, seorang Kepala Seksi Kepatuhan Internal Kanwil DJKN (Direktorat Jendral Kekayaan Negara) Jawa Barat. Tantri menyimpulkan bahwa WFH merupakan sistem kerja yang dipilih oleh pemerintah yang bertujuan untuk mengurangi penyebaran virus SARS-CoV-2 penyebab Covid-19 yang diharapkan dapat membantu mengurangi mobilitas orang, menjaga jarak fisik dan dan mengurangi kerumunan terutama pada aktivitas kerja dikantor. Pada artikelnya, Tantri (2020 : 31-35) menyadari bahwa bekerja dari rumah memiliki tantangan dan kendala yang tidak mudah sebab tidak semua pekerjaan dapat dikerjakan dari rumah, begitu juga dengan ketersediaan alat kerja dan komunikasi, kurangnya koordinasi, gangguan “domestik” lingkungan rumah tangga dan lainnya merupakan tantangan yang dihadapi dalam aktivitas bekerja dari rumah.
5. Jurnal penelitian yang berjudul “Model Prototyping Pada Pengembangan Sistem Informasi” disusun oleh Dwi Purnomo dari Universitas Widyagama Malang, di jurnal ini menjelaskan bahwa bagi pengembang sistem

salah satu tujuan dari prototyping adalah untuk mengumpulkan informasi dari pengguna. Pada metode pengembangan sistem dengan prototyping, aplikasi yang telah dikembangkan sebelum digunakan atau dikembangkan lebih lanjut akan menjalani tahap User Evaluation dimana pada tahap inilah pengembang sistem akan banyak menyerap informasi dari pengguna hingga dapat memastikan produk sistem informasi yang dihasilkan sesuai dengan kebutuhan pengguna. Menurut Purnomo (2017 : 55), penerapan metode prototyping tidak hanya dapat dilakukan pada pengembangan sistem berukuran kecil tapi dapat juga diterapkan pada pengembangan sistem besar. Ketika prototype terbentuk, pengguna dapat terlibat secara penuh dengan memberikan catatan dan masukan terhadap produk prototype yang diajukan oleh pengembang dan hal ini akan menguntungkan seluruh pihak yang terlibat dalam pengembangan sistem.

6. Literatur tentang pemrograman Kotlin yaitu buku yang berjudul “Programming Kotlin Applications : Building Mobile and Server-Side Applications with Kotlin”, karya Bret McLaughlin pada tahun 2021. McLaughlin merekomendasikan bahasa pemrograman Kotlin untuk digunakan dalam pengembangan aplikasi bergerak (mobile) berbasis Android. Kotlin adalah bahasa pemrograman berorientasi objek, bagi pengembang aplikasi Android yang sudah terbiasa menggunakan Java maka tidak akan kesulitan dalam menggunakan Kotlin. Menurut McLaughlin (2021, p. 3) bila Kotlin dikomparasi dengan Java, saat ini Kotlin telah menambahkan dan merubah beberapa fitur penting yang juga terdapat pada Java. Kotlin telah mendukung perluasan fungsi tanpa perlu melakukan *override* pada *parent class* secara keseluruhan, Kotlin juga telah menambahkan beberapa komponen fungsional yang digunakan dalam pemrograman seperti dukungan *lambda*, mendukung *Lazy Evaluation* dan peningkatan fitur lainnya.
7. Literatur yang diambil dari artikel Portal BSI Today yang berjudul “*Mengenal Metode Prototype Kelebihan dan Kekurangan*” membahas tentang metode pengembangan software menggunakan metode prototype, menurut artikel ini

Metode Prototype adalah suatu teknik dalam pengembangan sistem yang menggunakan prototype dalam menggambarkan sebuah sistem dalam suatu pengembangan sistem dengan tujuan agar pelanggan atau client pemilik sistem dapat gambaran yang jelas tentang sistem yang akan dikembangkan oleh pengembang aplikasi (2022 : 6). Keuntungan penerapan metode prototype ini, pengembang dapat lebih mudah dalam mengumpulkan informasi kebutuhan dari pengguna dan memungkinkan pengembang dapat menggali informasi lebih dalam terkait aplikasi yang dibutuhkan oleh pengguna. Selain keuntungan, penerapan metode ini juga memiliki kekurangan yaitu dapat mendatangkan kesulitan bagi pengembang untuk focus pada perencanaan awal yang telah disepakati bersama client karena pada tahap "*User Evaluation*" dimana pengembang menunjukkan mockup aplikasi yang telah dikembangkan biasanya akan memancing permintaan lain diluar perencanaan awal dari pengembangan aplikasi bahkan dapat menambah kompleksitas dan requirement aplikasi.

8. Literatur yang diambil yang berjudul "*Rekayasa Perangkat Lunak : Terstruktur dan Berorientasi Objek*" adalah sebuah buku karya Rosa A.S dan M. Shalahudin pada tahun 2018, yang membahas tentang perancangan perangkat lunak dengan menggunakan pemrograman terstruktur dan pemrograman berorientasi objek. Literatur ini menjelaskan bahwa dengan menggunakan metode prototype akan mempermudah pemahaman klien akan kebutuhannya terutama yang berkaitan dengan hal-hal teknis dan mempermudah klien dalam menjabarkan spesifikasi apa saja yang mereka butuhkan pada aplikasi yang akan dikembangkan (2018, p.31). Metode prototype akan mempermudah pengembang dalam menggali informasi lebih dalam dan detail terkait keinginan dan kebutuhan klien bahkan dengan metode prototype pelanggan bisa mendapatkan lebih jelas tentang aplikasi yang akan dikembangkan karena biasanya pengembang akan menggunakan mockup yang akan didemonstrasikan dihadapan klien. Dalam pengembangan sistem, mockup adalah suatu model desain aplikasi yang biasanya digunakan untuk

pelatihan, demonstrasi, promosi atau lainnya yang bertujuan untuk menunjukkan gambaran detail tentang aplikasi yang akan atau yang telah dikembangkan.

9. Literatur yang diambil dari artikel Portal Universitas Stekom Semarang (2021) yang ditulis oleh Zaenal Mustofa, M. Kom, yang berjudul “*Apa Itu Database ? Contoh Produk dan Fungsinya*” membahas pengertian dan jenis-jenis database yang saat ini banyak digunakan. Database adalah salah satu komponen yang sangat penting dalam sebuah sistem karena dapat memudahkan penggunaannya dalam melakukan proses aktivitas penyimpanan data, perubahan data, penghapusan data dan pencarian data. Jenis-jenis dan fungsi database yang bermacam-macam pemnfaatannya disesuaikan dengan fungsi dan pemanfaatan data yang dibutuhkan pengguna juga disesuaikan dengan karakteristik sistem yang digunakan dalam proses bisnis yang dijalankan oleh pengguna.
10. Jurnal penelitian yang berjudul “*Implementasi Object Oriented Metodologi dan UML pada Pengembangan Sistem Informasi Keuangan Organisasi*” hasil penelitian Ari Satrio Putra dan Kholid Haryono dari Universitas Islam Indonesia Yogyakarta pada tahun 2021. Ari Satrio dan Kholid berpendapat bahwa analisis dan perancangan sistem dengan menggunakan metode Object Oriented Analysis Design lebih menekankan pada aktivitas dan proses daripada data. Pemodelan dilakukan dirancang dengan menggunakan konsep kelas (class) dan pewarisan (inheritance) yang rancangannya kemudian di tuangkan dalam bentuk visual menggunakan Unified Modelling Language (UML).
11. Literatur pada Portal Universitas Stekom Semarang yang ditulis Laksamana Rajendra H.A.F, M.T, M.Kom pada 25 Januari 2022 yang berjudul “*UML*”. Pada artikelnya Laksamana menjelaskan definisi dan pemanfaatan Unified Modelling Language (UML) pada proses perancangan aplikasi. UML versi 1.0 pertama kali diperkenalkan pada bulan Januari 1997 oleh pembuatnya yang bernama Object Management Group. Laksamana mendefinisikan UML

sebagai suatu bahasa standar dalam memvisualisasikan perancangan dan pendokumentasian sistem atau dokumen yang tertuang dalam UML bisa juga disebut sebagai *blueprint* dari sebuah program. Berikut adalah tujuan dan fungsi dari pemanfaatan UML menurut Laksamana, antara lain :

1. UML digunakan sebagai bahasa pemodelan secara visual yang dapat dijadikan informasi kepada pengguna terkait dengan sistem yang akan dikembangkan.
 2. Dokumen UML dapat menyatukan semua informasi-informasi yang terbaik yang dituangkan dalam pemodelan.
 3. UML merupakan gambaran model visual yang ekspresif dalam pengembangan sistem
 4. UML memodelkan sistem secara *object oriented* tidak hanya sekedar menggambarkan model sistem software saja
 5. UML akan mempermudah pengguna dalam memahami rancangan sistem
 6. UML bisa disebut sebagai *blueprint* dari suatu sistem karena di dalamnya banyak memuat detail perancangan suatu sistem
12. Jurnal penelitian yang diterbitkan Universitas Brawijaya Malang pada tahun 2019 yang berjudul “*Analisis dan Perancangan Sistem Informasi Pengajuan Cuti Pegawai Menggunakan Pendekatan Object Oriented Analysis and Design (OOAD) (Studi Pada PT. Pos Indonesia Surabaya Kebonrojo)*”, hasil penelitian Fiera Arnandya Rani, Satrio Agung Wicaksono & Admaja Dwi Herlambang. Pada jurnalnya Fiera Arnandya Rani, dkk mengutip teori dari Kendall & Kendall (2011) menyatakan bahwa OOAD merupakan suatu metode analisis dan desain sistem berdasarkan konsep dunia nyata. Pendekatan metode ini akan menyajikan visualisasi pemodelannya dalam bentuk pemodelan standar yang bernama Unified Modelling Language (UML).