

BAB II

TINJAUAN PUSTAKA

2.1 Teori Dasar

2.1.1 Pengertian Rancang Bangun

Menurut (Rahayu, Fajri, & Hambali, 2019) Kata “rancang” pada rancang bangun merupakan kata sifat dimana berasal dari kata “perancangan” dimana merupakan suatu prosedur dalam menerjemahkan hasil sebuah analisis pada suatu aplikasi atau sistem ke dalam bahasa pemrograman sebagai upaya untuk menerjemahkan secara detail bagaimana bagian-bagian dari sistem tersebut di implementasikan. Sehingga istilah pada rancang bangun yang dimaksud pada penelitian ini antara lain suatu bentuk produk hasil dari penelitian dengan penerjemahan data-data yang telah diperoleh yaitu seperti hasil dari wawancara, observasi serta dokumentasi yang digunakan untuk memudahkan tahapan perancangan yang dilakukan.

2.1.2 Pengertian Sistem Informasi

Sistem adalah suatu kesatuan yang terdiri dari beberapa unsur dapat berupa teknologi seperti perangkat lunak atau prosedur yang saling berkaitan dengan tujuan tertentu. Sehingga sistem informasi merupakan suatu sistem yang menyediakan informasi-informasi dimana dalam hal ini dapat membantu dalam kegiatan operasional juga dalam pengambilan keputusan pada suatu perusahaan (Ginting, Fadlina, & Karim, 2022). Sehingga kemampuan dari sistem informasi cukup banyak seperti menyimpan informasi dalam jumlah yang sangat besar terlebih memungkinkan pengaksesan informasi jumlah besar tersebut, serta tentunya meningkatkan efektivitas dan efisiensi orang-orang yang bekerja pada suatu kelompok atau organisasi.

2.1.3 Pengertian Integrasi Data, Database, dan MySQL

Integrasi data merupakan penggabungan data yang berasal dari *database* yang berbeda-beda kedalam suatu *database* atau basis data yang baru (Sianturi, Hasugian, Simangunsong, & Nadeak, 2019). Sehingga pada penelitian ini data yang di integrasi merupakan data yang berasal dari modul-modul kegiatan dalam penelitian. Integrasi kegiatan sendiri dalam penelitian ini memiliki maksud penggabungan dari modul-modul kegiatan yang ada pada perusahaan terkait.

Menurut (Rachmadi, 2020) *database* atau dapat diartikan dengan basis data dimana terdiri dari dua kata, basis sendiri diterjemahkan sebagai sebuah gudang ataupun markas, data sendiri merupakan suatu keterangan dari fakta-fakta yang mewakili suatu objek dimana terwujud dalam bentuk angka, huruf, teks, gambar, dan lain-lain. Sehingga pengertian dari *database* merupakan kumpulan kelompok data dimana saling terhubung serta di organisasikan sehingga tidak sulit untuk dimanfaatkan nantinya. Kumpulan kelompok data-data tersebut dalam wujud suatu file atau arsip yang saling berhubungan serta tersimpan pada media penyimpanan elektronis yang bertujuan untuk kemudahan dalam mengatur, memilah, serta mengelompokkan data sesuai tujuan. Dalam pengoperasiannya, *database* memiliki sebuah konsep yaitu *Structured Query Language* (SQL).

Menurut (Fitri, 2020) SQL atau *Structured Query Language* adalah suatu konsep dalam mengoperasikan *database* seperti penyeleksian atau pemasukan data sehingga dapat memudahkan pengerjaan operasi data. Dalam penggunaannya SQL memiliki perkembangan konsep yang merupakan turunan dari konsep SQL sendiri yaitu *mySQL*. *MySQL* sendiri merupakan sebuah perangkat lunak pada sistem manajemen basis data SQL atau biasa disebut dengan *Database Management System* (DBMS).

2.1.4 Pengertian Honorarium

Menurut (Yakub & Solihin, 2020) definisi dari kata “honorarium” adalah merupakan upah dimana diperuntukkan sebagai imbalan dari suatu jasa dimana seperti diberikan kepada pegawai, dan lain-lain.

2.1.5 Pengertian SDLC

Menurut (Ahmad, Krismanik, Rupilele, Muliawati, & Syamsiyah, 2022) *Software Development Life Cycles* atau SDLC merupakan suatu siklus rancangan atau pengembangan aplikasi atau perangkat lunak. Menurut (Subakti, Widiastiwi, Syamsiyah, Nugroho, & S, 2022) SDLC sendiri merupakan proses membuat dan modifikasi dalam perangkat lunak berdasarkan metode pengembangan perangkat lunak tersebut. SDLC memiliki fase-fase diantaranya

1. *Planning* (Perencanaan)

Merupakan fase mendasar dalam memahami mengapa suatu sistem informasi harus dibangun hingga menentukan bagaimana akan membangunnya.

2. *Analysis* (Analisis)

Pada fase ini terdapat langkah-langkah seperti menganalisa siapa yang akan menggunakan sistem tersebut, apa yang sistem dapat lakukan, serta melakukan analisa pada sistem yang sedang berjalan. Kemudian juga mengidentifikasi serta mengembangkan konsep untuk sistem yang akan dibangun.

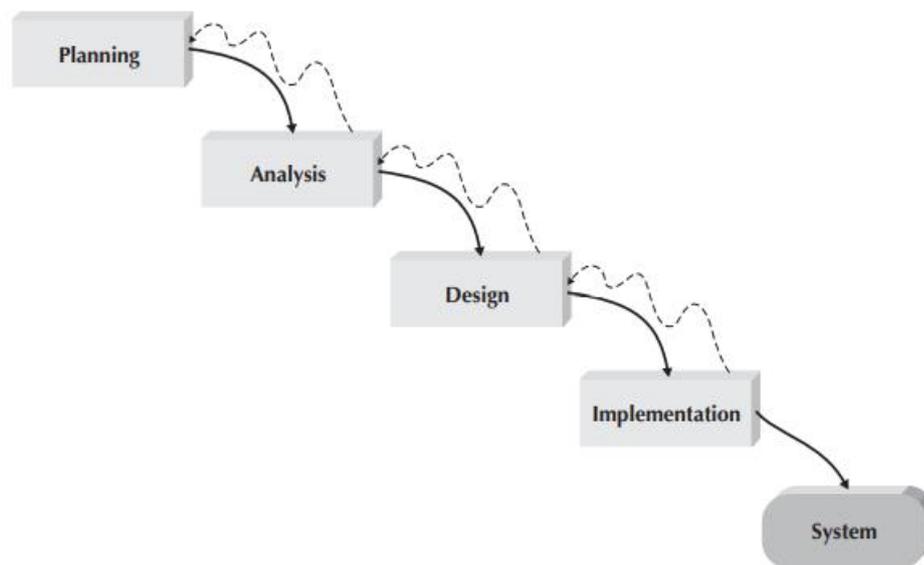
3. *Design* (Desain)

Pada fase ini menentukan seperti apa sistem yang akan dibangun akan beroperasi seperti dalam hal perangkat keras atau *hardware*, perangkat lunak atau *software* sampai infrastruktur jaringan. Selain itu juga menentukan antar muka pengguna (*User Inteface*), kemudian program sampai *database* serta file-file yang dibutuhkan.

4. *Implementation* (Implementasi)

Fase implementasi merupakan fase terakhir pada SDLC dimana merupakan fase terpanjang karena pada fase ini sistem yang baru dibangun berdasarkan desain yang telah dilakukan, sehingga fase implementasi meliputi pemrograman sistem hingga pengujian.

SDLC mendasari beberapa jenis metode utama dalam pengembangan perangkat lunak, pada penelitian ini menggunakan metode pengembangan *waterfall*. Menurut (Dennis, Wixom, & Tegarden, 2015) metode pengembangan *waterfall* atau *waterfall development* merupakan sebuah metode pengembangan pada structure design dimana seluruh fase dilakukan berurutan sesuai dengan alur dari satu fase ke fase berikutnya seperti terlihat pada gambar 2.1. Pada pelaksanaannya juga metode *waterfall* membutuhkan persetujuan di setiap fase pengembangan dari pemilik atau sponsor dari proyek agar dapat lanjut ke fase berikutnya.



Gambar 2. 1 Metode Pengembangan *Waterfall* (Dennis, Wixom, & Tegarden, 2015)

Terdapat beberapa kelebihan dalam menggunakan metode pengembangan *waterfall* dalam membangun suatu aplikasi atau sistem, dimana seluruh fase dilakukan secara berurutan sehingga pengembang dapat mengidentifikasi kebutuhan dari pengguna jauh sebelum membuat program sistem. Hal ini dapat meminimalisir perubahan pada kebutuhan pengguna yang dimana dijadikan acuan dalam mengembangkan aplikasi.

2.1.6 Pengertian UML

Menurut (Dennis, Wixom, & Tegarden, 2015) *Unified Modelling Language* (UML) merupakan diagram-diagram yang bertujuan sebagai kosa kata umum berorientasi objek serta teknik pembuatan diagram dalam memodelkan setiap proyek pengembangan sistem, mulai dari fase analisis hingga dengan fase implementasi. Dalam menganalisis UML memiliki 2 pendekatan diantaranya *Object Oriented Analysis and Design* (OOAD), dan *Structure System and Design* (SSAD) dimana pada penelitian ini menggunakan pendekatan OOAD.

Pada buku (Dennis, Wixom, & Tegarden, 2015) juga dijelaskan bahwa *Unified Modelling Language* (UML) memiliki 3 jenis model dalam penggambaran diagram diantaranya.

1) *Functional Model*

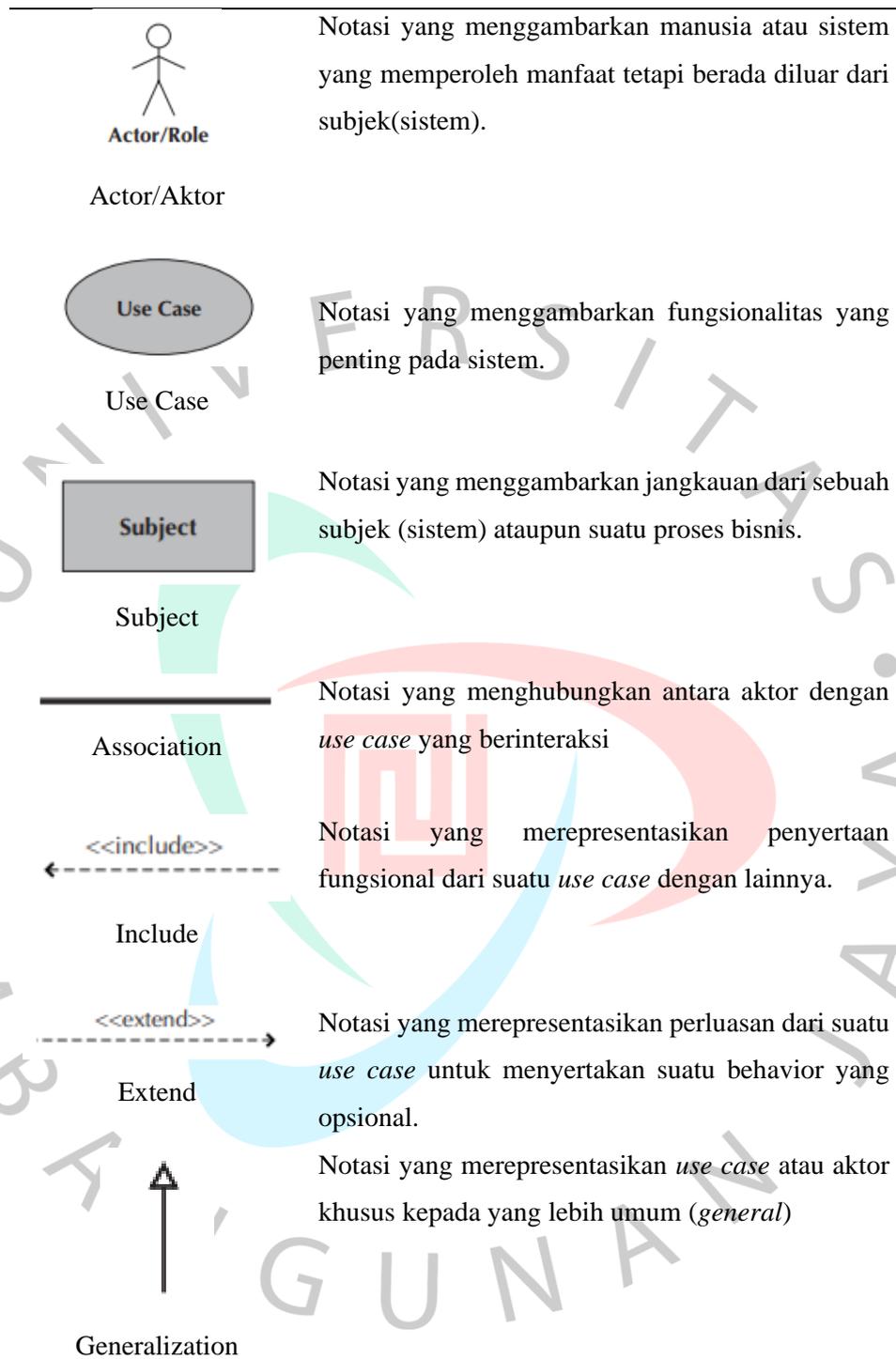
Diagram *functional model* pada UML berfungsi dalam menggambarkan proses bisnis serta interaksi suatu sistem informasi terhadap lingkungannya. Pada pendekatan OOAD *functional model* digambarkan dengan 2 diagram yaitu *use case diagram* dan *activity diagram*.

- *Use Case Diagram*

Use case diagram memuat penggambaran secara dasar yang merepresentasikan bagaimana sistem berinteraksi dengan lingkungannya (aktor). Dalam penggambaran *use case diagram*, sebagai tujuan untuk memudahkan pemahaman tidak diperbolehkan lebih dari tiga sampai sembilan *use case* pada suatu model. Selain itu juga meminimalisir garis yang bersinggungan dengan cara aktor ditempatkan dekat dengan *use case* yang berasosiasi dengan aktor tersebut. Berikut notasi notasi yang digunakan pada diagram *use case*.

Tabel 2. 1 *Tabel Notasi Use Case Diagram*

Notasi	Deskripsi
--------	-----------



Sumber: (Dennis, Wixom, & Tegarden, 2015)

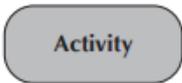
Use case diagram juga memiliki *use case description* untuk menyediakan informasi-informasi yang diperlukan untuk membangun *structural* dan *behavioral* diagram dimana memiliki format yang dikit untuk

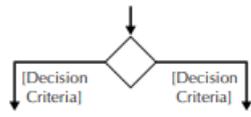
mempermudah pengguna dalam memahami *use case* tersebut. Dalam pembuatannya *use case description* memuat flow atau alur yang berjalan pada satu *use case* dari proses yang memicu sampai kepada alur alternatif yang memungkinkan.

- *Activity Diagram*

Activity Diagram digunakan untuk memodelkan apapun dari level tertinggi alur kerja bisnis dimana terdiri dari *use case* yang berbeda-beda, sampai detail dari suatu individu *use case*. Sehingga *activity diagram* dapat menggambarkan atau memodelkan suatu proses sampai alur kerja apapun. Dalam menggambarkan *activity diagram* diperlukan melihat kembali kebutuhan dari sistem yang akan dibangun serta dokumentasi yang telah dibuat seperti *use case diagram* dan lainnya. Selain itu juga menentukan aktifitas yang perlu dilakukan dalam mendukung suatu proses bisnis, dalam penggambarannya berikut merupakan notasi *activity diagram*.

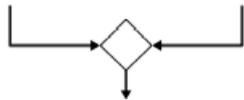
Tabel 2. 2 Tabel Notasi Activity Diagram

Notasi	Deskripsi
 Activity	Notasi yang menggambarkan manusia atau sistem yang memperoleh manfaat tetapi berada diluar dari subjek (sistem).
 Control Flow	Notasi yang menggambarkan urutan dari eksekusi aktifitas.
 Initial node	Notasi yang menggambarkan awal dari sekumpulan aktifitas (<i>activity diagram</i>).
 Final activity node	Notasi yang menggambarkan berhentinya control flow pada sebuah <i>activity diagram</i>



Decision node

Notasi yang merepresentasikan kondisi sebuah pengujian agar memastikan alur hanya menyusuri satu jalan.



Merge Node

Notasi yang digunakan untuk menyatukan alur keputusan yang berbeda dimana berasal dari decision node.



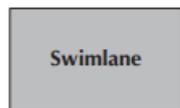
Fork Node

Notasi yang digunakan untuk memisah suatu behavior menjadi berbeda alur yang berjalan secara bersamaan.



Join node

Notasi yang digunakan untuk menyatukan alur berbeda yang berjalan secara bersamaan.



Swimlane

Notasi yang digunakan untuk memecah activity diagram menjadi baris dan kolom dimana bertujuan untuk menetapkan suatu aktivitas individu/objek yang bertanggung jawab dalam menjalankan aktivitas tersebut

Sumber: (Dennis, Wixom, & Tegarden, 2015)

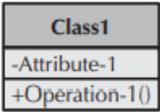
2) Structure Model

Diagram *structure model* pada UML menggambarkan struktur dari objek yang mendukung suatu proses bisnis pada perusahaan. Pada *functional model* merepresentasikan bagaimana sistem akan berperilaku, setelah itu tentunya perlu memahami informasi yang digunakan dan dibuat oleh sistem seperti informasi pengguna, transaksi dan sebagainya. Dengan itu maka pemodelan struktur merepresentasikan objek yang digunakan dan dibuat berdasarkan sistem bisnis. *structure model* pada pendekatan OOAD digambarkan dengan *class diagram*.

- *Class Diagram*

Class diagram adalah diagram model yang menyajikan *class-class* dan relasi antar *class* yang tetap konstan di dalam sistem dari waktu ke waktu. Pada *class diagram* juga digambarkan *class* beserta behavior dari *class* tersebut (*method/operation*). Berikut notasi-notasi yang digunakan dalam menggambarkan *class diagram*.

Tabel 2. 3 Tabel Notasi Class Diagram

Notasi	Deskripsi
 <p>Class</p>	Notasi yang merepresentasikan terkait orang, tempat, atau benda dimana dibutuhkan sistem dalam menangkap dan mengumpulkan informasi.
<p>attribute name /derived attribute name</p> <p>Attribute</p>	Notasi yang merepresentasikan properti yang menggambarkan pernyataan dari suatu objek.
<p>operation name ()</p> <p>Operation</p>	Notasi yang merepresentasikan suatu fungsi atau method yang ada pada suatu class.
<p>AssociatedWith</p> <p>0..* 1</p> <p>Association</p>	Notasi yang merepresentasikan relasi atau hubungan antar class atau dengan <i>class</i> itu sendiri.
<p>Generalization</p>	Notasi yang merepresentasikan jenis relasi sejenis terhadap beberapa class.
<p>0..* IsPartOf 1</p> <p>Aggregation</p>	Notasi yang merepresentasikan logika dari bagian hubungan antara beberapa <i>class</i> juga dengan class itu sendiri.
<p>1..* IsPartOf 1</p> <p>Composition</p>	Notasi yang merepresentasikan secara fisik dari bagian hubungan antara beberapa class juga dengan class itu sendiri.

Sumber: (Dennis, Wixom, & Tegarden, 2015)

3) Behavior Model

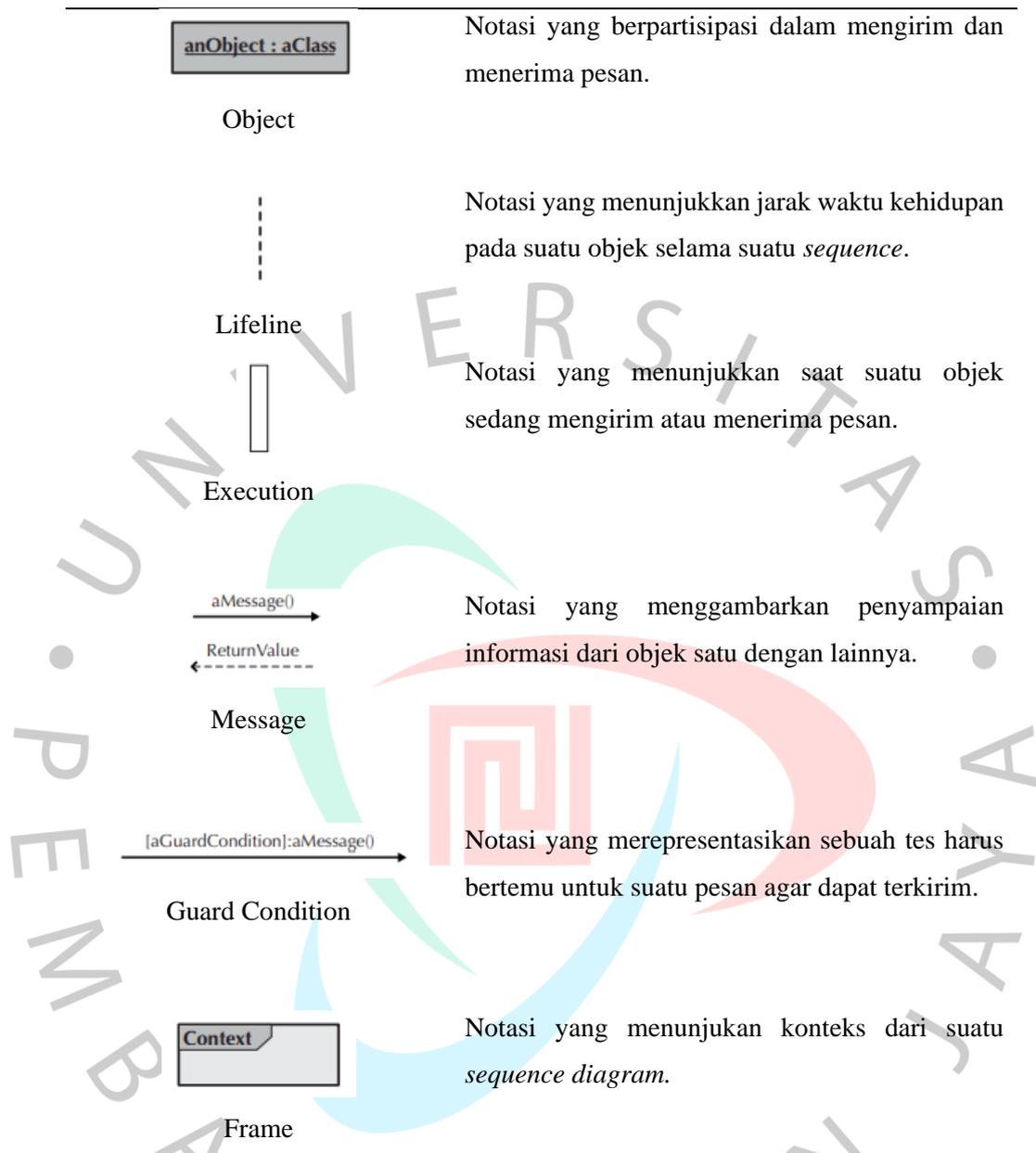
Diagram *behavioral model* pada UML bertujuan menggambarkan aspek dinamis internal dari suatu sistem informasi dimana mendukung proses bisnis dari suatu perusahaan. Penggambaran behavioral modelling menjadikan *use case diagram* pada *functional modelling* sebagai acuan, dimana karena tujuan dari *behavioral modelling* yang menggambarkan interaksi sistem dengan pengguna dari sistem informasi terkait. Penggambaran behavioral sendiri menggunakan *interaction diagram* yaitu *sequence diagram* dan *communication diagram*, dimana pada penelitian ini menggunakan *sequence diagram* pada *behavioral modelling*.

- *Sequence Diagram*

Sequence diagram mengilustrasikan objek-objek yang terlibat pada *use case* serta pesan-pesan yang lewat antara mereka selama satu proses *use case* berjalan. Sehingga *sequence diagram* merupakan model dinamis yang memperlihatkan urutan pesan-pesan yang lewat tersebut diantara objek dalam suatu interaksi yang ditentukan. Dalam penggambaran *sequence diagram* juga dapat ditentukan berdasarkan *use case diagram* yang telah dibuat. Notasi-notasi yang digunakan pada *sequence diagram* adalah sebagai berikut.

Tabel 2. 4 Tabel Notasi Sequence Diagram

Notasi	Deskripsi
 anActor	Notasi yang menggambarkan manusia atau sistem yang memperoleh manfaat tetapi berada diluar dari subjek (sistem).
Actor/Aktor	



Sumber: (Dennis, Wixom, & Tegarden, 2015)

2.1.7 Pengertian User Requirement

Menurut (Maniah & Hamidin, 2017) *User Requirement* atau *Requirement Analysis* merupakan tahapan dimana terjadinya interaksi antara pengguna akhir (*end user*) dengan analis atau pengembang sistem. Pada pengumpulan dokumen *user requirement* dilakukan diskusi panjang bertujuan untuk mendapatkan hasil kesepakatan antara pihak pengguna sistem dengan pihak pengembang dari

sistem tersebut dalam menentukan batasan-batasan dari sistem yang akan dibangun. Dokumen dari *user requirement*.

Menurut (Martono, Sucipto, & Maulana, 2019) salah satu metode dalam menganalisa kebutuhan dalam membangun sistem atau perangkat lunak menggunakan metode elisitasi. Metode elisitasi sendiri didapatkan melalui proses wawancara atau diskusi antara pengguna dengan pengembang, elisitasi sendiri dilakukan melalui tiga tahap diantaranya

- Elisitasi tahap pertama memuat usulan-usulan rancangan untuk rancangan sistem/aplikasi yang akan dibangun oleh pihak pengguna
- Elisitasi pada tahap kedua berisi hasil dari pengelompokkan dari elisitasi tahap pertama yang telah diperoleh berdasarkan metode MDI yaitu *Mandatory, Desireable, Inessential* (MDI), dimana bertujuan dalam memilah antara kebutuhan rancangan sistem/aplikasi yang penting serta harus ada (*Mandatory*) dengan kebutuhan-kebutuhan yang lainnya.
- Elisitasi pada tahap ketiga merupakan hasil dari pengerucutan dari elisitasi pada tahapan kedua dengan mengeliminasi seluruh *user requirement* dengan kategori I (*Inessential*) pada tahap MDI, setelah itu diklasifikasikan kembali pada tahap 3 menggunakan metode *Technical, Operational, Economic* (TOE)
- Elisitasi tahap final adalah hasil akhir dari tahapan-tahapan elisitasi yang akan dijadikan acuan dalam pengembangan sistem atau perangkat lunak oleh pihak pengembang dimana terdiri dari kebutuhan fungsional dan non fungsional. Kebutuhan fungsional sendiri meliputi kebutuhan yang difasilitasi internal atau didalam sistem sementara non fungsional difasilitasi diluar sistem

2.1.8 Pengertian Interaksi Manusia Komputer

Menurut (Nidhom, 2019) interaksi manusia komputer merupakan sebuah ilmu pengetahuan yang mempelajari hubungan antara manusia (pengguna/user) dengan komputer dimana memiliki kesamaan serta saling membutuhkan

sehingga dapat mencapai suatu tujuan yang diharapkan, dengan cara menjalankan suatu sistem yang dapat melakukan perancangan antarmuka sistem (*interface*). Interaksi manusia komputer memiliki tiga faktor dalam pembuatannya diantaranya:

- a) Manusia, dimana manusia merupakan alasan interaksi dapat terjadi karena adanya suatu keinginan dari manusia akan dapat memudahkan pekerjaannya.
- b) Komputer, komputer sebagai media dimana digunakan untuk berinteraksi dalam hal ini merupakan bahasa pemrograman, masukan (*input*) dan keluaran (*output*).
- c) Fungsi, pembuatan interaksi manusia komputer didasari dari fungsi dimana hasil akhir yang didapat untuk membantu pekerjaan manusia.

2.1.9 Pengertian Black Box Testing

Menurut (Bierig, Brown, Galván, & Timoney, 2021) Black box testing atau juga biasa disebut *functional testing* merupakan metode *testing* yang dilakukan berdasarkan dengan spesifikasi dari program tanpa perlu memiliki pengetahuan tentang program yang berjalan didalam aplikasi tersebut. Sehingga *black box testing* lebih mengarah untuk melakukan verifikasi bahwa suatu program sesuai dengan spesifikasi kebutuhan yang ditetapkan tanpa perlu mengetahui implementasi dari program tersebut. *Blackbox testing* dikarakteristikan berdasarkan kemungkinan kombinasi nilai *input* pada suatu *subset*. Pada penelitian ini menggunakan *blackbox testing* bertujuan untuk memudahkan dalam melakukan implementasi karena dapat dilakukan oleh personil selain dari tim pengembang sistem, dimana tentunya tanpa mengetahui tentang program yang berjalan pada aplikasi tersebut.

2.2 Tinjauan Studi

- 1) Jurnal hasil penelitian yang dilakukan oleh Ahmad Afandi, Lutfi Ali Muharram, Eko Fajar Y yang berjudul “INTEGRASI SISTEM INFORMASI AKADEMIK DAN E-LEARNING” dan diterbitkan oleh

Universitas Muhammadiyah Jember pada *website repository* pada tahun 2016(<http://repository.unmuhjember.ac.id/446/1/ARTIKEL%20JURNAL.pdf>). Sistem Informasi Akademik (SIA) sendiri adalah suatu sistem pengelolaan data yang dimiliki Universitas Muhammadiyah Jember. Sistem *E-Learning* sendiri merupakan suatu media untuk pembelajaran jarak jauh dimana memanfaatkan peran dari teknologi seperti computer atau jaringan komputer serta internet. Namun 2 sistem ini belum terintegrasi sehingga masalah yang terjadi yaitu, seperti contoh data yang sudah masuk ke dalam *database* pada Sistem Informasi Akademik (SIA) tidak secara langsung atau otomatis tersimpan juga pada *database E-learning* melainkan data tersebut harus diinput secara manual di *E-learning*. Masalah didefinisikan dengan jelas dan cukup mendalam yaitu karena proses yang cenderung kurang efisien karena tidak adanya integrasi antara sistem, menurut saya masalah yang dijabarkan sudah sesuai dan jelas. Kesimpulan dari jurnal ini yaitu dengan integrasi yang dilakukan pada Sistem Informasi Akademik (SIA) dapat menghubungkan data yang dikirim pada *E-Learning* sehingga mempermudah pengelolaan data pada kedua sistem tersebut, pada rancangan sistem juga cukup dijelaskan dari DFD dan *flowchart* dimana alur proses saat menghubungkan *database* dari kedua sistem sehingga dapat terintegrasi.

- 2) Penelitian yang berupa jurnal yang dilakukan oleh Miki Mania, Bambang Eka Purnama, Sukadi yang berjudul “SISTEM INFORMASI PENGGAJIAN KARYAWAN MITRA KARYA PRIMA DI PEMBANGKIT LISTRIK TENAGA UAP 1 PACITAN” dan diterbitkan oleh IJSE – Indonesian Journal on Software Engineering 2016 ini menjelaskan tentang perancangan sistem informasi untuk penggajian karyawan Mitra Karya Prima. Pengolahan gaji pada karyawan dilakukan masih belum terautomasi atau masih belum seperti penyimpanan pada data masih berbentuk arsip yang menyebabkan perlunya waktu dan tenaga lebih serta sering terjadi kesalahan ketik sampai hilangnya arsip. Maka dari itu diperlukan membangun Sistem Informasi untuk pengolahan gaji karyawan

dimana tentunya dapat meningkatkan kualitas dari kinerja karyawan yang lebih efisien. Sehingga masalah yang ditekankan pada jurnal ini adalah penggajian yang tidak efisien baik secara waktu dan belum lagi jika terdapat kesalahan karena *human error*, menurut saya sudah dijabarkan dengan jelas. Selain itu upaya dari penelitian ini adalah membangun sistem penggajian untuk karyawan Mitra Karya Prima di pembangkit tenaga listrik uap 1 Pacitan, sehingga dapat disimpulkan dengan dibangunnya sistem penggajian karyawan ini dapat mempermudah serta mempercepat kinerja admin di perusahaan tersebut juga akan mempercepat proses pembuatan laporan rekap dalam setiap periode tertentu. Pada rancangan sistem cukup dijelaskan detail dari DFD dimana proses-proses hingga DFD level 3 dijelaskan sehingga terlihat bahwa sistem cukup menunjang kegiatan penggajian.

- 3) Jurnal yang disusun oleh M Harist Murdani dan Bilal Luqman Bayasut yang berjudul “INTEGRASI SISTEM INFORMASI PADA TINGKATAN SYSTEM SPECIFICATION INTEGRATION PADA KOPERASI KARYAWAN COCA-COLA SIER” dan diterbitkan pada Jurnal Ilmiah Edutic /Vol.5, No.2 pada *website* di tahun 2019 (<https://journal.trunojoyo.ac.id/edutic/article/download/4738/3577>) menjelaskan tentang permasalahan yang terjadi pada koperasi Coca-Cola Sier dimana pada koperasi coca-cola Sier proses bisnis yang berjalan antara lain, ketika karyawan mendaftar sebagai anggota koperasi akan mendapatkan hak seperti mengajukan pinjaman dengan jumlah tertentu dimana pembayaran dapat melalui potong gaji. Karyawan juga harus melaporkan hasil untuk semua operasional, termasuk pemotongan gaji, pemotongan pembelian toko, pemotongan pembayaran pinjaman, semua proses ini dilakukan secara terpisah, sehingga tidak ada integrasi antara ketiga sistem informasi tersebut. Sehingga dengan itu masalah sudah didefinisikan dengan jelas diantaranya proses bisnis yang berjalan pada koperasi masih berjalan secara terpisah karena belum adanya integrasi antara ketiga sistem informasi tersebut yaitu sistem informasi simpan

pinjam, sistem informasi buku besar, dan sistem informasi ritel/toko. Kontribusi dari penelitian ini antara lain membangun sistem untuk integrasi antara 3 sistem informasi yang berjalan pada koperasi untuk menjaga keintegritasan data. Dapat disimpulkan dengan dibangunnya sistem untuk mengintegrasikan antara ketiga sistem informasi yang berjalan pada koperasi karyawan dapat menggunakannya dengan mudah karena mengingat bahwa integrasi ini hanya mengubah mekanisme basis data sehingga terhubung dengan satu sama lain, terlihat juga pada rancangan basis data terlihat cukup detail dimana menggabungkan data-data dari ketiga sistem informasi yang dimana data-data tersebut juga saling berkaitan.

- 4) Penelitian yang dilakukan oleh Christian Yulianto Rusli, Muhammad Faizal Kurniawan, Wahyu Setianto dimana berjudul “ANALISA DAN DESAIN INTEGRASI DATA PENDIDIKAN KOTA PEKALONGAN” diterbitkan oleh Jurnal Litbang Pekalongan Vol. 13 Tahun 2017 pada *website* (<https://jurnal.pekalongankota.go.id/index.php/litbang/article/view/54/52>) menganalisis tentang rancangan infrastruktur *database*. Beberapa sekolah pada Pekalongan telah menggunakan sistem Data Pokok Pendidikan Dasar dan Menengah atau dapat disingkat (DAPODIKDASMEN) dimana termasuk sumber data yang utama mengenai pendidikan tingkat dasar ataupun menengah nasional. Permasalahan kali ini dihadapi oleh dinas pendidikan kota Pekalongan adalah dimana data yang telah dilaporkan oleh sekolah pada pekalongan melalui DAPODIKDASMEN memiliki kesalahan dalam otoritas karena otoritas data tersebut dimiliki oleh Kementerian Pendidikan. Berdasarkan kendala-kendala tersebut, maka akan dibuat usulan model integrasi sistem kependidikan di kota Pekalongan. Sehingga masalah yang ditekankan pada artikel ini adalah ketidakmampuan untuk akses pengolahan data karena tidak memiliki otoritas, dan menurut saya sudah cukup jelas masalah yang dijabarkan. Upaya dari penelitian ini antara lain membuat usulan rancangan infrastruktur dan rancangan *database* untuk mempermudah terwujudnya integrasi pada sistem kependidikan di Kota Pekalongan. Dapat disimpulkan dengan desain infrastruktur ini dapat

diterapkan data integrasi yang selaras dengan Permendagri No. 61 tahun 2015 yang dimana tentang Hak Akses Serta Pemanfaatan NIK, KTP serta sesuai dengan keadaan dimana DAPODIK sudah tersedia di Dinas Pendidikan. Pada rancangan infrastruktur juga sudah dijelaskan cukup mendalam bahkan dengan skema 2 alternatif sehingga rancangan yang telah dibuat cukup membantu dalam pengusulan pembuatan integrasi *database*.

- 5) Jurnal yang disusun oleh Auji Hamizan, Mayasari, Rahayu Saputri, Randy Novhendra Pohan berjudul “SISTEM INFORMASI PENGGAJIAN DI PT. PERKEBUNAN NUSANTARA IV” pada website (<https://ojs.unikom.ac.id/index.php/jamika/article/download/2656/1827/>) tahun 2020 berisi tentang perancangan sistem informasi untuk penggajian karyawan pada PT. Perkebunan Nusantara IV karena belum adanya sistem informasi untuk penggajian, mengingat banyaknya karyawan yang bekerja hal ini menyebabkan ketidak efisienan waktu menyulitkan karyawan seperti pada penginputan, penyusunan, sampai peminjaman arsip. Sehingga dapat saya tekankan bahwa penjelasan masalah sudah cukup lengkap antara lain tidak efektifnya kegiatan penggajian dalam manajemen waktu serta keamanan data itu sendiri. Upaya yang dilakukan pada jurnal untuk menangani masalah tersebut yaitu membangun sistem informasi untuk penggajian karyawan dengan harapan untuk meningkatkan efektivitas serta keamanan data dalam kegiatan penggajian. Dapat disimpulkan bahwa dengan dibangunnya sistem informasi penggajian ini tentunya dapat meningkatkan efisiensi dari kinerja karyawan dan lain lain, pada rancangan sistem juga sudah cukup dijelaskan terlihat pada flowchart dan *use case* diagram yang menggambarkan alur kerja sistem.