

BAB II

TINJAUAN PUSTAKA

2.1 Teori Dasar

Pada bab ini akan menjelaskan mengenai teori-teori yang akan menjadi acuan dalam melakukan perancangan aplikasi berbasis web. Teori-teori dasarnya adalah sebagai berikut :

2.1.1 Rancang Bangun

Pada kamus besar bahasa indonesia, rancang dapat diartikan sebagai mengatur segala sesuatu (sebelum bertindak, mengerjakan, atau melakukan sesuatu) atau merencanakan. Menurut (Pressman, 2002). Rancang adalah serangkaian tahap-tahap untuk menerjemahkan hasil dari analisis sebuah sistem ke dalam bahasa pemrograman untuk menjabarkan dengan detail bagaimana komponen-komponen suatu sistem dapat diimplementasikan. Sedangkan bangun pada kamus besar bahasa indonesia dapat diartikan sebagai cara menyusun atau susunan yang merupakan bentuk wujud atau struktur. Dari pengertian yang sudah ada maka penulis dapat menyimpulkan bahwa rancang bangun adalah sebuah proses yang melibatkan dua tahap utama yaitu rancang sebagai tahap perencanaan dan penyusunan berdasarkan analisa awal dan bangun sebagai tahap implementasi dari rencana yang sudah dibuat pada tahapan rancang dengan demikian rancang bangun adalah sebuah proses keseluruhan dari perencanaan hingga implementasi sistem yang mencakup analisi, perancangan detail, dan pembangunan struktur sistem sehingga menjadi sebuah bentuk (aplikasi) yang berfungsi. Maka, untuk dapat membuat sebuah aplikasi IT *Helpdesk* yang bisa dioperasikan, rancang bangun harus dilakukan secara menyeluruh dan terstruktur.

2.1.2 Sistem Informasi

Menurut (Dedy Rahman Prehanto dkk., 2020) bahwa Sistem Informasi merupakan sebuah proses pengumpulan, penyimpanan, analisis sebuah informasi dengan tujuan tertentu. Sistem informasi yang terdiri dari data (*input*) dan menghasilkan laporan (*output*) sehingga diterima oleh sistem lainnya. Selain itu, buku berikut juga menyoroti kegiatan strategis dalam suatu

organisasi yang melibatkan pengambilan tindakan atau keputusan berdasarkan informasi yang telah diolah oleh sistem informasi. Maka, dengan demikian sistem informasi dianggap sebagai elemen kritis dalam mendukung kegiatan dan pengambilan keputusan di berbagai organisasi atau perusahaan. Menurut (Laudon & Laudon, 2020) sistem informasi merupakan seperangkat komponen yang saling berhubungan yang mengumpulkan, memproses, menyimpan, dan mendistribusikan informasi untuk mendukung pengambilan keputusan, koordinasi, dan kontrol dalam sebuah organisasi. Sistem ini dapat membantu perusahaan untuk mendapatkan keunggulan kompetitif, meningkatkan efisiensi operasional, serta mendukung inovasi.

Ciri dari sistem informasi antara lain adalah baru yang berarti pada suatu sistem informasi yang didapatkan merupakan informasi yang baru dan segar bagi penerima, tambahan (yang mana sistem informasi tersebut dapat diperbaharui atau memberikan tambahan terhadap informasi yang sudah ada sebelumnya.), Kolektif (di mana informasi yang dapat menjadi suatu koreksi dari informasi yang salah sebelumnya.), Penegas (informasi yang dapat mempertegas informasi yang sudah ada sebelumnya) Fungsi sebuah sistem informasi adalah mampu untuk menjadi tingkat ketersediaan kualitas dan pengalaman dalam mengelola sebuah sistem informasi secara kritis dan logis, mampu meningkatkan produktifitas kerja pada seluruh client dan stakeholders, mampu menganalisa dan juga meminimalisir terjadinya kerugian dari sisi ekonomi, memberikan aksesibilitas yang baik untuk pengguna yang mana maksudnya adalah sistem dapat digunakan di manapun dan kapanpun dengan mudah, dan dapat mencapai tujuan perusahaan secara cepat berdasarkan dengan dukungan data yang dapat dipertanggung jawabkan.

1. Komponen Sistem Informasi

- a) Perangkat Keras (*Hardware*), merupakan komponen fisik dari teknologi informasi, menyediakan infrastruktur fisik untuk menjalankan perangkat lunak dan menyimpan data. Contohnya dapat berupa komputer, server, perangkat jaringan, dan media penyimpanan

- b) Perangkat Lunak (*Software*), merupakan sebuah aplikasi, sistem informasi dan utilitas yang menjalankan perangkat keras dan memproses sebuah data menjadi sebuah informasi
- c) Data, merupakan sebuah data mentah seperti angka, teks, gambar atau bentuk lain yang dapat diolah menjadi informasi yang berguna oleh sistem informasi
- d) Prosedur, merupakan sebuah aturan dan proses yang digunakan untuk mengoperasikan dan mengelola sebuah sistem informasi. Prosedur dapat mencakup tata cara penggunaan sistem, kebijakan keamanan, dan alur kerja operasional.
- e) Manusia (*People*), *End User* dan profesional TI yang mengelola dan mengoperasikan sebuah sistem informasi. Seperti staf IT, manajer, dan analisis sistem

2. Fungsi Sistem Informasi

- a) Pengumpulan Data
Mengumpulkan data mentah dari berbagai sumber, termasuk input manual oleh pengguna dan otomatisasi melalui sensor ataupun perangkat lain
- b) Pemrosesan Data
Pemrosesan data adalah mengubah data mentah menjadi sebuah informasi yang berguna melalui proses perhitungan, penyortiran, pengelompokan, dan agregasi
- c) Penyimpanan Data
Menyimpan data dan informasi secara efisien dan aman untuk penggunaan di masa depan. Penyimpanan bisa dalam bentuk sebuah *database*, sebuah file, ataupun sistem penyimpanan lainnya.
- d) Pengelolaan Informasi
Pengelolaan informasi adalah mengelola sebuah informasi untuk mendukung operasi, manajemen, dan pengambilan suatu keputusan. Dapat mencakup pengelolaan *database*, keamanan informasi, dan manajemen pengetahuan.
- e) Distribusi Informasi

Distribusi Informasi merupakan sebuah proses di mana dapat menyediakan informasi kepada pihak yang membutuhkan dalam format yang tepat dan waktu yang sesuai. Distribusi informasi dapat berupa sebuah laporan, *Dashboard*, email, ataupun sistem komunikasi lainnya.

3. Jenis Sistem Informasi

- a) Transaction Processing Systems (TPS)
Menurut (Laudon & Laudon, 2020) *Transaction Processing Systems* (TPS) berfungsi untuk memproses transaksi bisnis rutin seperti pemesanan, pembelian, penggajian karyawan, dan inventaris. TPS membantu dalam mencatat, memproses, menyimpan dan mengelola data transaksi.
- b) Management Information System (MIS)
Information System berfungsi untuk menghasilkan laporan periodik untuk manajemen, seperti laporan penjualan mingguan, bulanan atau laporan keuangan tahunan. MIS dapat membantu manajer dalam melakukan pengambilan keputusan berdasarkan data yang terstruktur dan ringkasan dari TPS.
- c) Decision Support System (DSS)
Decision Support System berfungsi sebagai pendukung pengambilan keputusan yang lebih kompleks dan tidak terstruktur dengan menyediakan analisis data, simulasi data, dan model prediksi
- d) Executive Information Systems (EIS)
Executive Information Systems dapat memberikan akses mudah dan cepat kepada informasi penting bagi eksekutif perusahaan. Biasanya menyediakan sebuah *Dashboard* dengan indikator kinerja utama dan analisis tingkat tinggi
- e) Customer Relationship Management Systems (CRM)
Customer Relationship Management Systems berfungsi untuk mengelola interaksi perusahaan dengan pelanggan, menyimpan data pelanggan, dan mendukung penjualan serta layanan pelanggan.

CRM mampu membantu untuk meningkatkan hubungan dengan pelanggan dan meningkatkan kepuasan pelanggan

f) Supply Chain Management Systems (SCM)

Supply Chain Management Systems (SCM) berfungsi untuk mengelola aliran barang, informasi, dan uang melalui rantai pasokan dari pemasok hingga *end pengguna*.

g) Enterprise Resource Planning Systems (ERP)

ERP berfungsi untuk mengintegrasikan semua fungsi bisnis utama ke dalam suatu sistem yang terpadu.

h) Knowledge Management Systems (KMS)

KMS berfungsi untuk mendukung penciptaan, pengumpulan, penyimpanan dan juga distribusi pengetahuan dalam organisasi. KMS membantu perusahaan dalam berbagi pengetahuan dan pengalaman antar anggota organisasi.

Contoh Implementasinya pada aplikasi IT *Helpdesk* adalah Transaction Processing System (TPS) yaitu dengan mencatat dan melacak setiap tiket yang masuk dan Management Information System (MIS) untuk menghasilkan laporan tentang kinerja tim dukungan seperti laporan jumlah tiket yang telah selesai pada periode tertentu.

2.1.3 Helpdesk

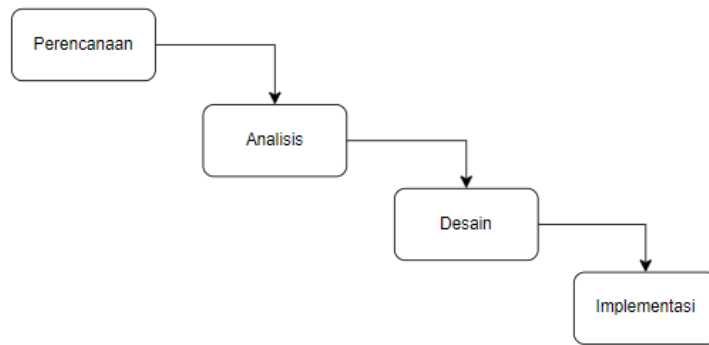
Menurut (Blokdyk, 2018) *Helpdesk* merupakan sebuah sistem manajemen untuk membantu dalam menangani kebutuhan pengguna terkait dengan pertanyaan, pelayanan, bantuan teknis, dan keluhan terhadap layanan pada perusahaan dengan memanfaatkan sistem penomoran untuk memudahkan penelusuran terhadap tindakan penyelesaian yang telah dikoordinasikan oleh Tim IT. Sistem ini dapat mempermudah karyawan divisi terkait dalam menangani masalah dan laporan yang disampaikan oleh pengguna dan juga manajemen dalam memonitoring kinerja dan memberikan suatu solusi untuk memecahkan masalah yang dihadapi oleh pengguna. Manfaat dari adanya sistem *Helpdesk* ini adalah dapat meningkatkan kepuasan pengguna, dapat meningkatkan kualitas dari layanan digital, dan juga dapat mengembangkan

bisnis di mana apabila kepuasan pengguna dan kualitas dari layanan digital maka bisnis pun juga menjadi berkembang karena tidak adanya permintaan-permintaan pengguna akibat tidak diberlakukannya sistem *Helpdesk*. Berikut adalah kelebihan dari penggunaan *Helpdesk*:

1. *Helpdesk* dapat memberikan sebuah solusi atas keluhan yang masuk dalam waktu yang singkat
2. *Helpdesk* dapat memberikan laporan terhadap kinerja staf IT kepada manajer IT
3. *Helpdesk* dapat mengatur pembagian kerja staf berdasarkan tiket isu yang diajukan oleh *End User*.

2.1.4 System Development Life Cycle (SDLC)

Menurut (Muharni, 2021) yang mengutip pengertian dari (Sukanto & Shalahuddin, 2018) bahwa *System Development Life Cycle*(SDLC) merupakan proses mengembangkan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan orang untuk mengembangkan sistem-sistem perangkat lunak sebelumnya (berdasarkan *best practice* atau cara-cara yang sudah teruji baik). Seperti halnya proses metamorphosis pada kupu-kupu yang indah maka dibutuhkan beberapa tahap untuk dilalui, sama halnya dengan membuat perangkat lunak, memiliki alur tahapan yang dilalui agar menghasilkan perangkat lunak yang bagus dan berkualitas. Dengan demikian dapat disimpulkan bahwa *System Development Life Cycle*(SDLC) adalah serangkaian tahapan yang harus dilalui dalam pengembangan perangkat lunak untuk mencapai hasil yang baik dan berkualitas. Menurut (Dennis dkk., 2015) bahwa *System Development Life Cycle*(SDLC) memiliki empat fase utama yaitu adalah perencanaan, analisis, desain, dan implementasi yang masing-masing mencakup langkah-langkah *deliverables* atau produk konkret yang dihasilkan sebagai bagian dari suatu proses yang menjelaskan proyek secara rinci.



Gambar 2.1. Tahap-tahap SDLC (Sumber : Dennis, Dkk (2015))

1. Fase Perencanaan

Tahap ini merupakan proses mendasar untuk memahami mengapa sistem informasi harus dibangun dan menentukan bagaimana tim proyek akan membangunnya. Tahap perencanaan memiliki dua langkah :

- a) Selama inisiasi proyek, nilai bisnis sistem untuk organisasi akan diidentifikasi. Sebagian besar ide untuk sistem baru datang dari departemen-departemen lain dalam bentuk permintaan sistem. Permintaan sistem menyajikan ringkasan singkat tentang kebutuhan tersebut akan menciptakan nilai bisnis. Departemen IT melakukan kerjasama dengan sponsor proyek untuk melakukan analisis kelayakan. Permintaan sistem dan analisis kelayakan disajikan kepada komite persetujuan untuk memutuskan apakah proyek layak untuk dibangun dan dilaksanakan.
- b) Setelah proyek disetujui, proyek memasuki fase manajemen proyek. Selama manajemen proyek, manajer proyek membuat rencana kerja, menugaskan staf proyek dan menerapkan teknik teknis untuk membantu tim proyek mengendalikan dan mengarahkan proyek melalui seluruh SDLC. Hasil dari manajemen proyek adalah rencana proyek yang menjelaskan bagaimana tim proyek akan mengembangkan sistemnya.

2. Fase Analisis

Fase analisis menjawab pertanyaan siapa yang akan menggunakan sistem, apa yang akan dilakukan oleh sistem, dan di mana serta kapan sistem akan digunakan. Selama fase ini, tim proyek menganalisa sistem saat ini,

mengidentifikasi peluang untuk perbaikan, dan mengembangkan konsep untuk sistem baru. Fase ini memiliki tiga langkah :

- a) Strategi analisis dikembangkan untuk membimbing upaya tim proyek. Strategi tersebut biasanya mencakup analisis sistem saat ini dan masalahnya serta cara-cara untuk merancang sistem baru
- b) Langkah berikutnya adalah pengumpulan persyaratan bisa melalui wawancara atau kuesioner. Analisis informasi ini bersama dengan masukan dari sponsor proyek dan banyak pihak lainnya mengarah kepada pengembangan konsep untuk sistem baru. Konsep sistem kemudian digunakan sebagai dasar untuk mengembangkan serangkaian model analisis bisnis, yang menggambarkan bagaimana bisnis akan beroperasi jika sistem baru dikembangkan
- c) Kemudian analisis, konsep sistem, dan model digabungkan ke dalam dokumen yang disebut proposal sistem yang disajikan kepada sponsor proyek dan pembuat keputusan lainnya yang memutuskan apakah proyek akan terus berjalan.
Proposal sistem merupakan *deliverable* awal yang menjelaskan persyaratan bisnis apa yang harus dipenuhi oleh sistem baru.

3. Fase Desain

Fase Desain memutuskan bagaimana sistem akan beroperasi dari segi perangkat keras, perangkat lunak, infrastruktur jaringan, antarmuka pengguna, basis data, dan program yang diperlukan. Fase desain memiliki empat langkah :

- a) Strategi desain dikembangkan untuk menentukan apakah sistem akan dibuat secara internal, dialihdayakan, atau membeli perangkat lunak yang sudah ada
- b) Desain arsitektur dasar kemudian dibuat, mencakup perangkat keras, perangkat lunak, dan infrastruktur jaringan yang akan digunakan. Desain antarmuka menentukan navigasi pengguna serta formulir dan laporan yang akan digunakan
- c) Spesifikasi basis data dan file dibuat untuk mendefinisikan data yang akan disimpan dan lokasinya

- d) Tim analisis mengembangkan desain program yang menjelaskan program-program yang perlu ditulis secara tepat apa yang akan dilakukan oleh setiap program.

Kumpulan *deliverables* ini (desain arsitektur, desain antarmuka, spesifikasi basis data dan desain program) adalah spesifikasi sistem yang diserahkan kepada tim pemrograman untuk implementasi. Pada akhir fase desain, analisis kelayakan dan rencana proyek akan diperiksa kembali, direvisi apabila ada yang perlu direvisi, keputusan lain dibuat oleh sponsor proyek dan komite persetujuan tentang apakah proyek harus dihentikan atau dilanjutkan.

4. Fase Implementasi

Fase terakhir pada SDLC adalah fase implementasi, di mana pada fase ini sistem akan benar-benar dibangun. Fase ini biasanya mendapat perhatian yang paling besar karena merupakan fase yang paling panjang dan paling mahal dalam proses pengembangan. Fase ini memiliki tiga langkah :

- a) Langkah pertama adalah konstruksi sistem di mana sistem dibangun dan diuji untuk memastikan bahwa sistem berfungsi sesuai desain. Pengujian menjadi salah satu langkah paling kritis dalam implementasi.
- b) Langkah selanjutnya adalah instalasi sistem. Instalasi adalah proses di mana sistem lama dimatikan dan sistem baru diaktifkan. Salah satu aspek paling penting adalah pengembangan rencana pelatihan untuk mengajarkan pengguna cara menggunakan sistem baru dan membantu mengelola perubahan yang disebabkan oleh sistem baru
- c) Tim analisis kemudian menetapkan rencana dukungan untuk sistem. Rencana ini biasanya mencakup tinjauan pasca-implementasi formal atau informal serta sistematis untuk mengidentifikasi perubahan besar dan kecil yang diperlukan untuk sistem.

Model SDLC yang dijelaskan berikut ini merupakan model pengembangan sistem air terjun (*waterfall*) yang dapat disebut juga sebagai model sekuensi liner (*sequential linear*) karena pendekatan alur hidup perangkat lunaknya secara sekuensial atau berurutan dimulai dari perencanaan,

analisis, desain, hingga implementasi. Model SDLC ini sangat cocok untuk pengembangan perangkat lunak yang tidak berubah-ubah.

2.1.5 Object-Oriented Analysis and Design

OOAD adalah pendekatan yang berfokus pada penggunaan objek untuk membangun sistem. Objek adalah entitas yang menggabungkan data dan perilaku. Pendekatan ini menggunakan konsep enkapsulasi, pewarisan, dan polimorfisme yang membantu menciptakan kode yang lebih modular dan mudah dikelola. Tahapan utama pada OOAD adalah :

1. Analisis Kebutuhan: Mengidentifikasi dan mendokumentasikan kebutuhan pengguna.
2. Desain Sistem: Membuat model sistem yang mencakup struktur objek dan hubungan antar objek
3. Desain Objek: Merinci desain setiap objek dan interaksinya
4. Implementasi: Menerjemahkan desain ke dalam bentuk kode pemrograman

Salah satu tools yang digunakan OOAD adalah UML atau *Unified Modeling Language*.


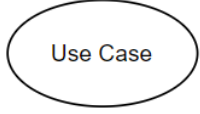

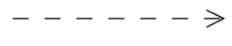
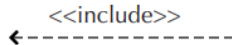
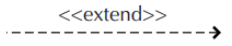
2.1.6 Unified Modelling Language (UML)

(Muharni, 2021) menjelaskan bahwa diperlukan adanya terminologi untuk memodelkan perangkat lunak yang akan dibuat dan perlu adanya standarisasi agar orang di berbagai negara dapat mengerti pemodelan perangkat lunak. Maka pada perkembangan teknik pemrograman berorientasi objek, muncul sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemrograman berorientasi objek, yaitu *Unified Modelling Language (UML)*. Pada buku ini mengutip penjelasan dari (Sukanto & Shalahuddin, 2018) bahwa UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun dan dokumentasi dari sistem perangkat lunak. Pengertian UML adalah sebagai berikut “UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung”. Berikut ini adalah macam – macam UML :

1. Use Case Diagram

Use Case Diagram merupakan urutan transaksi dari actor pada sistem. *Use Case* dapat menggambarkan apa yang sistem lakukan dari perspektif pengguna. *Use Case* merupakan sekumpulan skenario yang menjadi satu untuk tujuan pengguna.



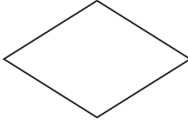



Tabel 2.1. Simbol Diagram Use Case

Simbol	Nama	Keterangan
 Actor	Actor	Mewakili peran orang, sistem yang lain ataupun alat ketika berinteraksi dengan sistem.
 Use Case	Use Case	Mewakili fungsi yang disediakan oleh sistem untuk aktor. <i>Use Case</i> menggambarkan tujuan yang dicapai oleh aktor melalui interaksinya dengan sistem
	Association (Hubungan)	Penghubung antara aktor dengan <i>Use Case</i> , yang menunjukkan bahwa aktor terlibat di dalam <i>Use Case</i> tersebut
	Generalization (Generalisasi)	Relasi antar kelas dengan makna umum-khusus dimana fungsi yang satu adalah fungsi yang lebih umum dari yang lainnya.
	Include	Menunjukkan bahwa <i>Use Case</i> akan selalu menyertakan (include) <i>Use Case</i> lain sebagai bagian dari prosesnya. <i>Use Case</i> (include) adalah bagian yang tidak terpisahkan dan akan selalu dijalankan setiap <i>Use Case</i> utama dijalankan.
	Extend	Menunjukkan bahwa <i>Use Case</i> dapat memperluas fungsionalitas dari <i>Use Case</i> lain. <i>Use Case</i> yang dapat di extend hanya dijalankan dalam situasi spesifik saja

2. Activity Diagram

Activity Diagram adalah teknik untuk menggambarkan logika prosedural, proses bisnis yang dalam beberapa hal mirip dengan sebuah diagram alir.

Tabel 2. 2 *Activity Diagram*

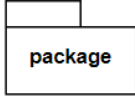
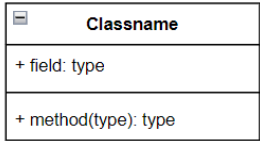

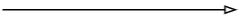




Simbol	Nama	Keterangan
	Initial Node / Status Awal	Status awal pada diagram <i>Activity</i>
	<i>Activity</i> / Aktivitas	Kegiatan yang dilakukan oleh suatu sistem
	Decision / Percabangan	Percabangan adalah ketika suatu sistem memiliki pilihan lebih dari satu.
	Join / Penggabungan	Digunakan untuk menggabungkan kembali <i>Activity</i> atau action yang paralel
	End Point	Akhir dari aktivitas yang dilakukan sistem
	Swimlane	Memisahkan organisasi bisnis yang bertanggung jawab dari aktivitas yang terjadi

3. Class Diagram

Class Diagram menggambarkan suatu struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sebuah sistem. Class dibagi menjadi 3 bagian yaitu nama kelas, atribut, dan operasi. Properti mewakili fitur-fitur struktural dari sebuah kelas juga sebuah konsep tunggal, tetapi tampak seperti dua notasi yang berbeda yaitu atribut dan

asosiasi. Atribut mendeskripsikan properti dengan sebaris teks di dalam kotak kelas. Asosiasi adalah sebuah garis solid antara dua kelas di tarik dari kelas sumber ke kelas target, nama properti bergerak sampai tujuan akhir sebuah asosiasi bersama dengan *multiplicity*.

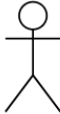
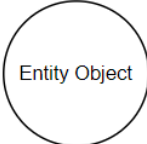
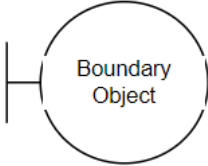
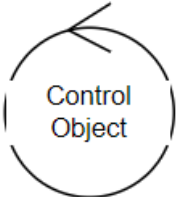

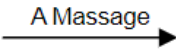
Tabel 2.3. Simbol Class Diagram

Simbol	Nama	Keterangan
	Package	Package adalah sebuah bungkus dari satu atau lebih class
	Class	Kelas pada struktur sistem
	Association (Hubungan)	Relasi antar class dengan makna umum
	Generalization (Generalisasi)	Relasi antar class dengan makna generalisasi spesialisasi (umum-khusus)
	Dependency (Ketergantungan)	Relasi antar class dengan makna ketergantungan antar class
	Aggregation	Relasi antar class dengan makna semua bagian
	Composite	Relasi antar class yang memiliki ketergantungan yang kuat
	Directed Association	Relasi antar class dengan arti bahwa kelas yang satu digunakan oleh kelas yang lain

4. Sequence Diagram

Sequence Diagram menggambarkan kelakuan objek pada *Use Case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek

Tabel 2.4. Simbol Sequence Diagram

Simbol	Nama	Keterangan
 Actor	Actor	Menggambarkan orang yang berinteraksi dengan sistem
 Entity Object	Entity Class	Menggambarkan hubungan aktivitas yang akan dilakukan
 Boundary Object	Boundary Class	Sebuah penggambaran dari form
 Control Object	Control Class	Penghubung antara boundary dengan tabel
	A Focus Control & A Life Line	Menggambarkan tempat mulai dan berakhirnya sebuah message
 A Message	A Message	Pengiriman pesan

2.1.7 Database

Basis data (*Database*) adalah kumpulan informasi yang disimpan selama periode waktu yang panjang, sering kali bertahun – tahun. Basis data sangat penting untuk semua jenis bisnis karena perusahaan menyimpan data –

data penting mereka ke dalam basis data. Kekuatan basis data terletak pada pengetahuan dan teknologi yang selalu berkembang dan diwujudkan ke dalam perangkat lunak khusus yang disebut sebagai sistem manajemen basis data (*Database Management System*). Sistem manajemen basis data adalah kumpulan data yang saling terkait dan sebuah program untuk mengakses data yang saling terkait tersebut. Tujuan adanya sistem basis data adalah sebagai tanggapan terhadap metode awal dalam pengelolaan data komersial terkomputerisasi (Putri, 2022)

2.1.8 Black Box Testing

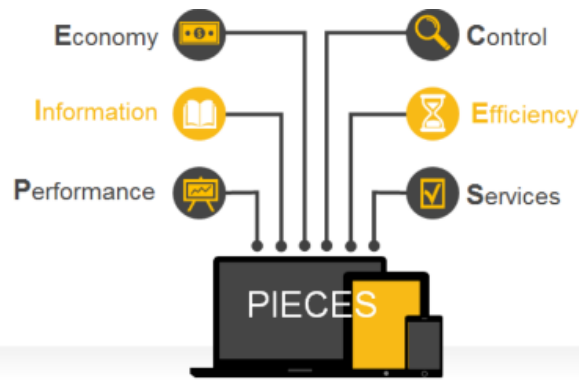
Black Box Testing adalah sebuah metode pengujian perangkat lunak dengan tes secara fungsionalitas aplikasi yang bertentangan dengan alur kerja. Uji kasus menggunakan deskripsi eksternal aplikasi termasuk spesifikasi, persyaratan, dan desain untuk menurunkan uji kasus (Whitten, 2004). Beberapa kategori untuk menemukan masalah dalam pengujian black box menurut (Azminuddin I. S. Azis, 2019) adalah :

1. Fungsi-fungsi yang salah atau hilang;
2. Kesalahan tampilan (*interface*);
3. Kesalahan dalam struktur data;
4. Kesalahan performa;
5. Kesalahan inisialisasi dan terminasi;

Salah satu cara metode black box testing adalah dengan mengeksekusi berbagai macam inputan setiap method dengan menjalankan event-nya pada pemograman berorientasi objek. Jika outputnya sesuai dengan fungsinya maka, method tersebut dianggap sudah bebas dari kesalahan fungsinya.

2.1.9 Analisis PIECES

Analisis PIECES menurut (Baijuri dkk., 2023) digunakan untuk mengidentifikasi masalah dan menganalisis enam komponen utama yang ada pada sistem, yaitu *Performance* (Kinerja), *Information* (Informasi), *Economy* (Ekonomi), *Control* (Kontrol), *Efficiency* (Efisiensi) dan *Services* (Layanan). Dari hasil analisis ini, dapat ditemukan beberapa masalah utama yang memerlukan perhatian lebih lanjut.



Gambar 2. 2 Komponen PIECES (Sumber : Baijuri dkk., 2023)

1. *Performance* (Kinerja)
 Pada tahap ini, dibutuhkan data tentang kinerja sistem yang sedang berjalan. Fokus dari analisis kinerja adalah untuk mengetahui sejauh mana sistem dapat memenuhi kebutuhan pengguna.
2. *Information* (Informasi)
 Pada tahap ini, analisa berfokus untuk mengevaluasi kemampuan sistem dalam menjamin informasi yang diperlukan tersedia dengan tepat, akurat, dan tepat waktu.
3. *Economy* (Ekonomi)
 Pada tahap ini, melibatkan analisa biaya dari sistem yang sedang berjalan yang bertujuan untuk mendapatkan pemahaman tentang penggunaan sumber daya yang diperoleh dari sistem
4. *Control* (Pengendalian)
 Analisis kontrol ini bertujuan untuk memastikan bahwa sistem memiliki kontrol yang memadai untuk mencegah kesalahan atau kegagalan
5. *Efficiency* (Efisiensi)
 Pada tahap ini, melibatkan bagaimana menganalisis penggunaan sumber daya sistem dengan tujuan untuk memastikan bahwa sumber daya sudah digunakan secara efektif
6. *Services* (Layanan)
 Tahap ini mencakup evaluasi bagaimana sistem dapat memberikan layanan kepada pengguna yang bertujuan untuk mengetahui sejauh mana sistem dapat memberikan layanan yang memadai kepada pengguna.

2.1.10 Metode Kualitatif

Metode kualitatif menurut (Sugiyono, 2020) adalah sebuah pendekatan penelitian yang bertujuan untuk memahami fenomena sosial dari perspektif partisipan. Sugiyono menjelaskan bahwa metode ini digunakan untuk mendapatkan data yang mendalam, memahami makna di balik data, dan meneliti proses sosial yang kompleks.

1. Pendekatan dan Tujuan

Metode kualitatif lebih menekankan pada pemahaman secara mendalam tentang fenomena yang diteliti daripada generalisasi hasil. Penelitiannya berfokus pada proses, makna, dan pemahaman kontekstual.

2. Desain Penelitian

Penelitian bersifat fleksibel dan berkembang sesuai dengan data yang ditemukan di lapangan. Desain penelitian sering kali tidak terstruktur secara ketat di awal, melainkan lebih bersifat dinamis dan berkembang seiring dengan proses penelitian.

3. Teknik Pengumpulan Data

Teknik pengumpulan data merupakan langkah yang paling strategis di dalam penelitian, hal itu dikarenakan bahwa tujuan utama dari penelitian adalah untuk mendapatkan data.

Teknik umum dalam penelitian kualitatif yang disebutkan oleh (Sugiyono, 2020) antara lain :

a) Observasi

Pengertian observasi yang dikutip oleh sugiyono dari beberapa ahli bahwa observasi merupakan dasar semua ilmu pengetahuan di mana peneliti dapat belajar tentang perilaku, dan makna dari perilaku tersebut. Observasi dapat diklasifikasikan menjadi dua yaitu observasi partisipatif, di mana peneliti terlibat dengan kegiatan sehari-hari orang yang sedang diamati atau digunakan sebagai sumber data dan peneliti ikut serta pada kegiatan tersebut untuk merasakan apa yang dirasakan oleh sumber data. Observasi partisipatif membuat data yang diperoleh menjadi lengkap, tajam, dan mengetahui makna dari setiap perilaku yang nampak. Lalu, yang kedua adalah observasi terstruktur atau

tersamar di mana dalam hal ini peneliti ketika melakukan pengumpulan data berterus terang bahwa sedang melakukan penelitian dan yang terakhir adalah observasi tak berstruktur di mana pada observasi ini dilakukan dengan tidak berstruktur karena fokus dalam penelitian belum jelas. Fokus observasi akan selalu berkembang seiring berjalannya kegiatan observasi.

Objek observasi dapat berupa tempat di mana interaksi sedang berlangsung, pelaku yang sedang memainkan peran tertentu, dan aktivitas yang sedang berlangsung.

b) Wawancara

Wawancara merupakan sebuah pertemuan dua orang atau lebih untuk bertukar informasi melalui tanya jawab, sehingga membangun makna mendalam terhadap topik yang dibicarakan. Ada tiga macam wawancara yaitu wawancara terstruktur di mana peneliti sudah menyiapkan instrumen penelitian berupa pertanyaan – pertanyaan yang akan ditanyakan kepada responden, wawancara semiterstruktur yang dalam pelaksanaannya lebih bebas bila dibandingkan dengan wawancara terstruktur dengan tujuan untuk menemukan permasalahan secara lebih terbuka, dan wawancara tak berstruktur di mana pada wawancara jenis ini, peneliti tidak menggunakan pedoman wawancara secara terstruktur namun hanya pertanyaan garis besarnya saja.

c) Dokumentasi

Dokumentasi adalah teknik pengumpulan data dari buku, arsip, dokumen, gambar, tulisan angka yang berupa sebuah laporan untuk mendukung penelitian. Peneliti dapat memanfaatkan data yang diperoleh dari dokumentasi untuk melakukan analisis yang lebih mendalam.

2.2 Tinjauan Studi

Penelitian ini penulis mengambil dan melakukan riset terhadap beberapa penelitian-penelitian yang sudah ada sebelumnya untuk dijadikan sebagai bahan referensi dan acuan penulis dalam merancang aplikasi dan penulisan laporan tugas akhir yang sedang penulis susun.

1. Dalam penelitian yang dilakukan oleh (Fitriansyah & Hasibuan, 2022) yang berjudul **PENERAPAN SERVICE REQUEST MANAGEMENT PADA TATA KELOLA LAYANAN TEKNOLOGI INFORMASI** yang dipublikasikan di JRIS: Jurnal Rekayasa Informasi Swadharma, Vol. 2, No. 1, Agustus 2022. Penelitian ini bertujuan untuk meningkatkan dan memperbaiki proses *Service Request Management* di PT. Mitra Integrasi Informatika berdasarkan *Information Technology Service Management*. Metodologi penelitian bersifat kualitatif dengan pengumpulan data melalui observasi di departemen IT, wawancara, dan analisis terhadap *database* yang ada. Hasilnya berupa perancangan aplikasi *Service Request Management* yang mencakup notifikasi email, keamanan data, pemantauan oleh *Top Level Management*, dan kemudahan akses melalui website serta *mobile* Android. Penelitian ini merujuk pada konsep *Information Technology Infrastructure Library (ITIL)* sebagai kerangka kerja untuk manajemen layanan teknologi informasi. Sistem yang diusulkan diharapkan dapat meningkatkan tata kelola teknologi informasi di perusahaan dan memberikan solusi efisien terhadap kendala formulir kertas dalam permintaan *hardware* dan *software*.
2. Dalam Penelitian yang dilakukan oleh (Barreto dkk., 2019) yang berjudul ***Ensuring Efficient IT Service Management to Increase Information Systems Availability*** yang dipublikasikan di *Journal of Information Systems Engineering and Management*, Vol. 4, Issue. 4, No. em0108, Desember 2019. Penelitian ini menyoroti adanya urgensi implementasi kerangka kerja Manajemen Layanan IT (ITSM) dalam menghadapi kompleksitas teknologi informasi (IT) saat ini. Dengan menitikberatkan pada kesadaran, akurasi, kelengkapan, dan kelolaan, penelitian ini mengusulkan suatu kerangka manajemen IT yang dapat mengatasi masalah tersebut. Penelitian ini memberikan dominan dengan menekankan integrasi alat-alat manajemen dan fokus pada implementasi yang efisien yang diharapkan dapat meningkatkan ketersediaan Sistem Informasi dengan memastikan pemahaman menyeluruh terhadap Sistem Informasi.
3. Dalam Penelitian yang dilakukan oleh (Amri & Ula, 2019) yang berjudul **APLIKASI SISTEM INFORMASI DALAM TATA KELOLA DATA PERBAIKAN *HARDWARE*** yang dipublikasikan di Sisfo: Jurnal Ilmiah

Sistem Informasi, Vol. 3, No. 2, Agustus 2019. Penelitian tersebut membahas tentang implementasi aplikasi sistem informasi dalam tata kelola data perbaikan *hardware* di Kantor Direksi PT Resources Regional Riau. Sistem pengelolaan data perbaikan *hardware* yang sebelumnya manual diubah menjadi sistem terkomputerisasi guna meningkatkan efisiensi dan mengurangi kesalahan data. Analisis sistem lama menunjukkan bahwa kesulitan dalam pencarian dan pemrosesan data dapat diatasi dengan perancangan sistem baru. Sistem ini mencakup input data *asset*, data perbaikan *hardware*, dan laporan, memberikan kemudahan bagi pegawai dalam pekerjaan sehari-hari. Melalui diagram konteks, data flow diagram, dan entity relationship diagram, penulis merancang tata kelola data yang terstruktur. Hasil implementasi menunjukkan beberapa fitur antarmuka, seperti halaman login, menu utama, input data *asset*, input data perbaikan *hardware*, dan laporan data perbaikan *hardware*. Kesimpulan yang didapat dari penelitian ini menyoroti adanya peningkatan efisiensi dan akurasi dalam pengelolaan data perbaikan *hardware* melalui sistem informasi aplikasi yang terkomputerisasi.

4. Penelitian kelima diteliti oleh (Rachmawati dkk., 2019) yang berjudul **“Pembangunan Helpdesk Ticketing System Berbasis Web Studi Kasus Universitas Yarsi”** yang dipublikasikan di Jurnal Teknologi Informasi YARSI, Vol. 6, No. 1, Maret 2019. Latar belakang adanya penelitian adalah bahwa penggunaan *Helpdesk Ticketing* sistem juga sangat diperlukan untuk diterapkan pada intuisi. Dengan adanya sistem *Helpdesk* maka dapat mempermudah dalam menyelesaikan masalah dan mengimbangi kebutuhan akan penyajian informasi yang cepat, tepat, dan akurat. Adanya sistem *Helpdesk* dapat memberikan dampak yang lebih baik dan efisien dibandingkan dengan monitoring secara manual. Rumusan masalah yang diperoleh adalah bagaimanakah cara merancang *Helpdesk Ticketing* system berbasis web, bagaimana sistem tersebut dapat membantu, dan bagaimana cara kerja *Helpdesk Ticketing* system yang baik dan menarik sehingga mempermudah pengguna untuk mengaksesnya. Adapun tujuan penulis meneliti penelitian tersebut adalah untuk membangun *Helpdesk Ticketing* system berbasis web pada institusi yang digunakan sebagai media penyampaian informasi masalah teknis yang di alami oleh pengguna

kepada tim pendukung dan melakukan pengujian terhadap *Helpdesk Ticketing* system berbasis web menggunakan metode usability untuk mengetahui tingkat kepuasan pengguna terhadap *Helpdesk Ticketing* system berbasis web. 21 Metode penelitian yang digunakan oleh penulis adalah metode penelitian pengembangan (Research & Development) di mana sumber data yang digunakan adalah data sekunder. Penulis menggunakan sumber data yang diperoleh melalui media perantara internet, buku, jurnal. Data yang didapat berupa tahapan dalam pembangunan *Helpdesk Ticketing* system berbasis web dan juga cara kerja *Helpdesk* tersebut. Kesimpulan secara keseluruhan, penelitian ini telah menghasilkan sebuah *Helpdesk Ticketing* System yang terintegrasi dengan baik berbasis web yang dapat digunakan oleh berbagai perusahaan atau institusi untuk menyelesaikan masalah dengan efisien. Berdasarkan hasil pengujian usability menggunakan metode yang telah disebutkan, tingkat kepuasan pengguna dalam menggunakan atau menjalankan *Helpdesk Ticketing* System berbasis web adalah 88,4%. Hal ini menunjukkan bahwa sistem ini dapat memudahkan dalam menyelesaikan masalah dan menjaga keseimbangan antara kebutuhan penyajian informasi yang cepat, tepat, dan efektif

5. Penelitian keenam dilakukan oleh (Mazia dkk., 2021) yang berjudul “**Rancang Bangun Sistem Informasi *Helpdesk Ticketing* berbasis Web pada PT Mitra Tiga Berlian Bekasi**” yang dipublikasikan di Jurnal Informatika Universitas Pamulang, Vol. 6, No. 1, Maret 2021. Penelitian ini ditujukan saat melakukan aktivitas kerja yang menggunakan teknologi pada perusahaan tersebut terdapat permasalahan yang muncul maka penulis melakukan observasi dan wawancara terlebih dahulu sebelum merancang sistem informasi *Helpdesk* berbasis web agar sistem yang dirancang dapat digunakan oleh divisi IT dalam laporan masalah dengan cepat dan dapat menyimpan informasi masalah dari setiap masalah dalam pengaduan oleh pengguna yang terjadi sehingga dapat meminimalisir kesalahan komunikasi di IT *Helpdesk*. Metode penelitian yang dilakukan oleh penulis adalah dengan menggunakan metode SDLC (*Software Development Life Cycle*) dengan metode waterfall di mana metode tersebut menyediakan pendekatan siklus hidup perangkat lunak secara sekuensial atau

berurutan. Penulis memulai dari pengumpulan data terlebih 20 dahulu di mana penulis melakukan observasi, melakukan wawancara mengenai permasalahan yang terjadi agar mendapatkan informasi yang diperlukan lalu mengumpulkan data melalui sumber-sumber bacaan seperti buku referensi dan juga jurnal yang terkait dengan sistem informasi IT *Helpdesk* sebagai sebuah acuan teori dalam melakukan penulisan penelitian. Lalu setelah itu penulis merancang sistem informasi IT *Helpdesk* menggunakan metode waterfall dimana setelah mendapatkan hasil observasi, penulis menganalisa kebutuhan pengguna terkait sistem yang akan dikembangkan. Kemudian selanjutnya melakukan perancangan desain sistem menggunakan ERD, implementasi dan testing. Hasilnya penulis membangun sebuah sistem informasi *Helpdesk Ticketing* yang dibangun berbasis web yang memberikan kemudahan untuk membantu proses dokumentasi laporan gangguan terkait teknologi informasi pada PT Mitra Tiga

● Berlian.