

BAB II

TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1 Kajian Pustaka

Tabel 2. 1 Kajian Pustaka

No	Judul	Perbandingan	Perbedaan	Kritik	Sintesis	Ringkasan
1	Rancang Bangun Aplikasi <i>UTS In Me</i> Berbasis Android Menggunakan Flutter Dengan Metode Rapid Application Development[7]	1. Keduanya menggunakan Flutter sebagai kerangka pengembangan utama untuk aplikasi mobile. 2. Mengimplemenasikan prinsip-prinsip RAD dalam pengembangan aplikasi untuk meningkatkan kecepatan dan responsivitas.	.Metode yang berbeda yang merupakan metode campuran (mixed). Metode pengumpulan datanya ada studi pustaka, wawancara, dan kuesioner	Mengkritik aplikasi yang hanya berbasis android	Dengan menggabungkan temuan dari penelitian A yang menekankan kecepatan pengembangan menggunakan Flutter dan penelitian B yang menyoroti efisiensi operasional, sintesis dapat menunjukkan bagaimana kecepatan pengembangan berkontribusi pada efisiensi operasional	Dibangun untuk mempermudah mengakses informasi, waktu dan lebih jelasnya menjadi lebih efisien.
2	Rancang Bangun Aplikasi Monitoring	Jurnal ini bertujuan sama untuk memonitor sebuah	Metode yang menggunakan Observasi. Dan	Kurangnya menu dan hanya berbasis android. Tidak ada	Sintesis dapat mengidentifikasi pola umum seperti	Pada pengembangan ini memudahkan

No	Judul	Perbandingan	Perbedaan	Kritik	Sintesis	Ringkasan
	Santri Berbasis Android[8]	objek dan memporleh informasi lebih efisien	perancangan system yang UML (Unified Modelling language)	profile lengkap untuk santri seperti foto santri.	kecepatan pengembangan dan fleksibilitas sebagai elemen kunci yang berkontribusi pada keberhasilan implementasi Flutter dengan pendekatan UML	memporleh informasi yang secara detail.
3	Evaluasi Rancang Bangun Aplikasi Pembelajaran Sejarah Proklamasi Berbasis Android[9]	Rancang bangun berbasis android dan mempermudah pekerjaan atau kegiatan	Kombinasi penelitian sekuensial eksploratif—yaitu, penelitian yang menggunakan dua pendekatan kualitatif dan kuantitatif secara berurutan—digunakan dalam banyak topik yang membahas pembelajaran dan prosedur.	Desain yang kurang modern dan banyak campuran assets. Desain Menu yang kurang menarik yang seharusnya dibuat lebih minimalis dan modern.	Sintesis dapat menciptakan gambaran menyeluruh tentang bagaimana penggunaan metodenya dalam pengembangan aplikasi mobile telah meningkatkan efisiensi operasional.	Penyampaian informasi yang menjadi lebih efisien untuk pembelajaran Sejarah berbasis android.
4	Rancang Bangun Aplikasi Pemesanan Jasa Perbaikan Komputer, Laptop, dan Smartphone Berbasis Android[10]	Mempersingkat waktu dan tenaga. Membuat pengguna terhindar dari rugi waktu dan tenaga yang berlebih.	Berbeda tema dan pada jurnal ini rancangan yang online. Metodenya yang menggunakan Waterfall dan perancangan menggunakan UML (Unified Modeling Language)	Tampilan interface yang kurang menarik. Desain User Interface harus ditingkatkan lagi agar user dapat mengerti dan memahami lebih jelas	Sintesis dapat menciptakan bagaimana pengembangan aplikasi mobile telah meningkatkan efisiensi operasional di berbagai sektor bisnis.	Kemudahan pada user dalam menggunakan aplikasi ini untuk pemesanan jasa perbaikan online

No	Judul	Perbandingan	Perbedaan	Kritik	Sintesis	Ringkasan
5.	Penerapan Metode Rapid Application Development (RAD) Dalam Pengembangan Sistem Informasi Data Toko[5].	Pendekatan Rapid Application Development digunakan dalam penelitian ini. Tidak membutuhkan waktu lama untuk sampai pada tahap pengembangan sistem.	Penelitian ini berfokus pada fungsi pada sistem yang dibuat. Dan memiliki beberapa hak akses yaitu Admin, Karyawan dan pemilik.	Penelitian ini yang berfokus pada fungsi sehingga desain tampilan sistem yang dibuat kurang menarik dan hambar.	Sintesis dapat menciptakan antara penelitian untuk membuat sistem yang memiliki beberapa hak akses.	Dengan penelitian ini saran pengembangan desain yang dibuat lebih menarik. Dan dengan adanya hak akses tersebut, mengurangi resiko kehilangan data di toko.
6	Rancang Bangun Aplikasi Pemesanan Menu Makanan Restoran Berbasis Android[11]	memanfaatkan kerangka Cardova dan teknik Rapid Application Development (RAD) untuk merancang aplikasi yang menyederhanakan tugas dan meningkatkan efisiensi	Jurnal ini yang hanya memfokuskan pada analisis kebutuhan spesifik pada pihak restoran	Terdapat menu "hubungi kami" yang tidak efektif karena jika memesan produk restoran langsung di tempat.	Sintesis dapat menciptakan bagaimana pengembangan aplikasi mobile telah meningkatkan efisiensi operasional di berbagai sektor bisnis.	Aplikasi pemesanan menu restoran dapat dimasukkan ke dalam sistem pemesanan melalui layanan web; dalam skenario ini, aplikasi web-admin berfungsi sebagai server dan aplikasi Android sebagai klien untuk melakukan pemesanan.
7	Pengembangan Sistem Informasi Akademik Berbasis Mobile Menggunakan Flutter Di Universitas Narotama Surabaya[12]	Rancang bangun aplikasi untuk mobile yang lebih fleksibel. Menggunakan framework Flutter	Jurnal ini menggunakan metode waterfall dan berfokus pada aspek desain antarmuka pengguna yang optimal	Saran pengembangan kedepannya bertujuan agar aplikasi sistem informasi akademik Universitas Narotama dapat menggabungkan	Dengan menggabungkan temuan dari sumber-sumber yang berbeda, sintesis dapat menyajikan pemahaman yang lebih mendalam	Saran untuk pengembangan kedepan agar semua fitur yang ada pada sistem informasi akademik untuk mahasiswa berbasis website dapat juga disematkan pada

No	Judul	Perbandingan	Perbedaan	Kritik	Sintesis	Ringkasan
				seluruh kemampuan yang terdapat pada sistem berbasis website untuk mahasiswa.	tentang bagaimana penggunaan Flutter dan UML bersama-sama dapat berkontribusi efisiensi operasional	aplikasi sistem informasi akademik Universitas Narotama untuk mahasiswa ini.
8	Development of e-butler: Introduction of robot system in hospitality with mobile application[13]	Rancang bangun aplikasi untuk mobile yang lebih fleksibel. Menggunakan framework Flutter	Jurnal Internasional ini pengembangan BeeButler System Application dan Robot yang memungkinkan menerima fasilitas dengan bantuan robot	Pada jurnal ini perancangannya hanya berfokus pada fungsionalitas yang akan digunakan, tampilannya yang kurang menarik dan terlalu hambar.	Sintesis laporan manajemen toko retail dengan Flutter, bersama jurnal pengembangan robot menggunakan BeeButler dan Flutter, menunjukkan kemajuan Flutter dalam aplikasi beragam. Laporan fokus pada manajemen toko, sementara jurnal membuka perspektif di bidang pengembangan robot. Gabungan keduanya menonjolkan fleksibilitas Flutter dalam menciptakan solusi lintas sektor.	Aplikasi BeeButler membantu pengguna memesan amenities hotel dengan fitur yang memungkinkan pemesanan tanpa perlu ke front desk, termasuk fitur Scheduled Order untuk menjadwalkan pengantaran amenities. Perbaikan di masa depan dapat melibatkan peningkatan desain UI, penggunaan database realtime, dan implementasi fitur notifikasi untuk meningkatkan pengalaman pengguna selama siklus pemesanan.

No	Judul	Perbandingan	Perbedaan	Kritik	Sintesis	Ringkasan
9	Mobile application for vehicle operation management[14]	Perancangan aplikasi mobile yang berfokus pada fungsionalitas untuk pengoperasian kendaraan menggunakan framework flutter	Perbedaan subjek, metode dan material kepada kendaraan. Dan mengembangkan untuk platform ios juga. Metode yang digunakan Model,View, Controller (MVC) dan untuk back-end menggunakan bahasa program Java	Pada jurnal ini perancangannya hanya berfokus pada fungsionalitas yang akan digunakan, tampilannya yang kurang menarik dan terlalu hambar.	Sintesis yang dapat mengembangkan platform untuk ios. Dan penambahan databases yang bisa menggunakan code program Java yang membuat lebih terjaga keamanan privasi.	Alat seluler yang dikembangkan memfasilitasi pengemudi dalam mengontrol operasi kendaraan dengan memberikan informasi analitis tentang biaya operasional, menentukan lokasi SPBU terdekat, dan meramalkan konsumsi bahan bakar.
10	DLDiagnosis: A mobile and web application for diseases classification using Deep Learning[15]	Rancang bangun aplikasi untuk mobile yang lebih fleksibel. Menggunakan framework Flutter	Jurnal ini menggunakan beberapa software architecture, yaitu RabbitMQ yang merupakan implementasi dari Advanced Message Queuing Protocol (AMQP). Keamanan databases yang menggunakan JSON di platform JHipster	Pada jurnal ini belum ada pengujian pada aplikasi yang dibuat. Dan berfokus pada fungsionalitas sehingga tampilan yang kurang menarik	Dengan menggabungkan temuan dari sumber-sumber yang berbeda, sintesis dapat menyajikan pemahaman yang lebih mendalam tentang bagaimana penggunaan RabbitMQ, AMQP dan JSON bersama-sama dapat berkontribusi efisiensi operasional	Pengembangan perangkat lunak berpotensi membantu para spesialis dan tenaga kesehatan ketika membuat keputusan mengenai tindakan yang akan diambil dengan mengambil hubungan dengan penyakit yang didiagnosis dan sesuai dengan tahapnya.

2.2 Landasan Teori

2.2.1 Toko Retail

Retail, menurut Dirjen Bea dan Cukai adalah perseorangan yang memperoleh barang dalam jumlah besar lalu menjualnya kembali kepada pelanggan perorangan. Pedagang Retail adalah entitas korporasi yang membeli komoditas dan menjualnya kembali kepada pengusaha (yang bukan pengguna akhir). Supermarket, department store, perusahaan ritel, dealer mobil, dll. adalah beberapa contoh pengecer [16].

Toko retail sebagai entitas bisnis memiliki manajemen persediaan menjadi fokus utama, dengan pendekatan strategis terhadap pengelolaan stok barang dagangan, pengawasan tingkat persediaan, dan pengaturan pembelian guna menjaga ketersediaan produk yang optimal. Selain itu, penataan tampilan toko memegang peranan penting dalam menciptakan pengalaman pelanggan yang positif, di mana desain tampilan yang efektif dapat meningkatkan daya tarik dan mempermudah pelanggan menemukan produk yang dicari[17].

Toko Retail mempunyai ciri-ciri yang sama dengan model penjualan, yaitu dilakukan secara swalayan secara eceran, langsung ke konsumen akhir; yaitu, pelanggan mengeluarkan barang dari rak dan membayar di kasir. Barang kebutuhan sehari-hari dan kebutuhan rumah tangga lainnya diperjualbelikan.[18].

2.2.2 Android

Android adalah sistem operasi berbasis Linux yang dikembangkan oleh Google untuk perangkat mobile. Dikenal dengan kebebasan, Android menawarkan kebebasan kepada pengembang untuk merancang aplikasi dengan beragam fitur. Android merupakan sistem operasi yang biasa digunakan pada gadget yang memiliki navigasi layar sentuh penuh, seperti tablet dan smartphone. Google Inc. telah mengambil alih kendali Android setelah membelinya dari Android Inc. pada tahun 2005.[19].

Pengembangan aplikasi Android menggunakan bahasa pemrograman Java atau Kotlin. Java adalah bahasa pemrograman yang umum digunakan dalam pengembangan Android, sedangkan Kotlin adalah bahasa yang diperkenalkan oleh Google sebagai bahasa resmi untuk pengembangan aplikasi Android, menawarkan sintaksis yang lebih ringkas dan modern[20].

Penggunaan sistem operasi Android pada perangkat mobile membawa kita pada pemahaman bahwa sistem ini dapat menghasilkan distribusi informasi dan pemenuhan kebutuhan secara real-time dengan akurasi. Selain itu, sistem ini dapat menyajikan sistem informasi yang bermanfaat, kompak, interaktif, dan mudah diakses oleh pengguna di seluruh dunia melalui satu perangkat seluler.[21].

2.2.3 Flutter

Dengan bantuan Flutter toolkit Antarmuka Pengguna Google (UI), pengembang dapat membuat prototipe sistem yang menarik secara visual yang dapat dikompilasi dari satu basis kode untuk digunakan dalam aplikasi berbasis web atau ponsel cerdas[22]. Pemrograman dalam Flutter menggunakan bahasa Dart, yang memberikan sintaksis yang jelas dan konsisten. Struktur dasar UI dalam Flutter dibangun dengan menggunakan widget, yaitu elemen dasar yang membentuk antarmuka pengguna. Fitur Hot Reload dalam Flutter memungkinkan pengembang melihat perubahan langsung pada aplikasi tanpa perlu memuat ulang seluruh aplikasi, meningkatkan efisiensi dalam pengembangan[23].

Keunggulan utama Flutter terletak pada kemampuannya untuk memberikan solusi cross-platform, memungkinkan pengembangan aplikasi yang dapat dijalankan di berbagai platform, termasuk Android, iOS, dan web, dari satu basis kode. Flutter menyediakan widget yang dapat diintegrasikan dengan desain Material Design untuk aplikasi Android dan desain Cupertino untuk aplikasi iOS, memastikan pengalaman pengguna yang konsisten[24]. Selain itu, komunitas Flutter yang aktif dan ekosistem

yang berkembang pesat memberikan sumber daya dan dukungan yang penting bagi pengembang.

Flutter bertujuan untuk menghilangkan kebutuhan mempelajari dua bahasa pemrograman yang berbeda dengan menyederhanakan dan mempercepat proses pembuatan aplikasi seluler yang berfungsi di iOS dan Android. Karena produk akhir dapat berjalan di berbagai platform termasuk desktop, web, iOS, dan Android seluler kerangka kerja ini saat ini berkembang dengan cepat [25].

2.2.4 Bahasa Pemrograman Dart

Google menciptakan bahasa pemrograman Dart untuk tujuan umum, yang dapat digunakan untuk membuat berbagai platform, seperti web, server seluler, dan Internet of Things. Selain itu, Flutter menggunakan bahasa ini sebagai bahasa standarnya[26]. Framework Flutter yang merupakan platform untuk mengembangkan aplikasi seluler menggunakan Dart. Kerangka kerja Flutter dibuat secara tegas untuk menciptakan antarmuka aplikasi seluler (UI) yang lebih menarik secara visual dan ekspresif.[27].

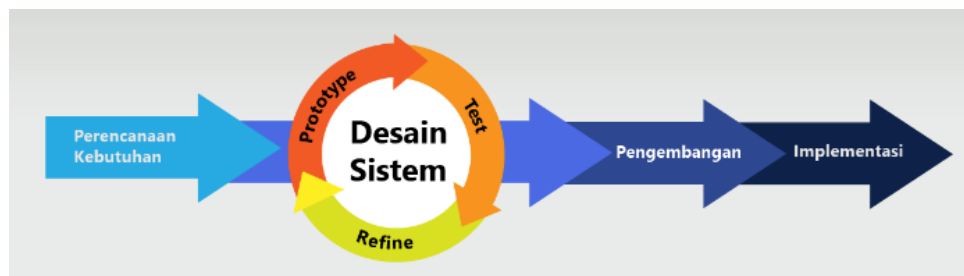
Dart adalah bahasa pemrograman berbasis kelas, single-thread, dan diketik secara opsional dengan tata bahasa yang terkenal. Itu dapat dijalankan secara bersamaan berkat mekanisme konkurensi yang disebut isolasi. Pengguna dapat mengeksekusi kode Dart melalui baris perintah selain mengeksekusinya di browser web dan mengubahnya menjadi JavaScript. ditempatkan di mesin virtual yang dibuat untuk Dart, memungkinkan pengguna menulis kode untuk komponen klien dan server aplikasi mereka dalam bahasa yang sama. Sintaks bahasa ini sangat mirip dengan JavaScript, C#, dan Java. Tujuan utama Dart adalah membuat bahasanya tampak alami. Ini adalah skrip Dart singkat dengan hanya satu fungsi, utama[28].

2.2.5 Metode Rapid Application Development

Metode Rapid Application Development (RAD) merupakan pendekatan pengembangan perangkat lunak yang memfokuskan pada pengembangan yang cepat dan responsif terhadap perubahan kebutuhan pengguna. Dalam konteks pengembangan aplikasi mobile untuk manajemen toko retail, penerapan RAD memiliki landasan teori yang signifikan.

RAD mengutamakan iterasi dan partisipasi aktif pengguna dalam seluruh siklus pengembangan. Pendekatan ini memungkinkan pengembang untuk menghasilkan prototipe aplikasi dengan cepat, memfasilitasi umpan balik pengguna yang langsung. Proses iteratif memungkinkan penyesuaian yang cepat terhadap perubahan kebutuhan dan penambahan fitur baru[29].

Terdapat tiga fase dalam RAD yang melibatkan penganalisis dan pengguna dalam tahap penilaian, perancangan, dan penerapan. Adapun ketiga fase tersebut adalah requirements planning (perencanaan kebutuhan), RAD design workshop (workshop desain RAD), dan implementation (implementasi). Sesuai dengan metodologi RAD tahap-tahap pengembangan sistem dari tiap-tiap fase pengembangan aplikasi.



Gambar 2. 1 Tahapan Metode Rapid Application Development[30]

1. Requirements Planning Dalam fase ini, Pengguna dan analis berkumpul untuk mendiskusikan tujuan aplikasi atau sistem dan kebutuhan informasi yang dihasilkan dari tujuan tersebut. Tujuan dari orientasi fase ini adalah untuk menyelesaikan masalah bisnis. Meskipun beberapa sistem di atas mungkin diarahkan oleh teknologi dan sistem informasi, tujuan perusahaan akan selalu didahulukan.
2. Design Sistem Fase ini adalah fase yang mirip dengan workshop untuk desain dan penyempurnaan. Pemrogram dan analis dapat

berkolaborasi untuk membuat dan menyampaikan representasi visual desain dan alur kerja kepada pengguna. Tergantung seberapa besar aplikasi yang harus dikembangkan, lokakarya desain ini mungkin memerlukan waktu beberapa hari untuk diselesaikan. Pengguna bereaksi terhadap prototipe saat ini selama lokakarya desain RAD, dan penganalisis memodifikasi modul yang diusulkan sebagai respons terhadap umpan balik pengguna. Menurut Kendall, jika pengembangnya adalah pengguna atau pengembang berpengalaman, karya inovatif ini dapat merangsang pengembangan ke tingkat yang lebih tinggi.

3. Pada tahap pengembangan ini hingga versi final, aplikasi versi beta dibangun menggunakan desain sistem yang telah disepakati. Pada titik ini, selain mempertimbangkan umpan balik pengguna atau klien secara teratur, pemrogram juga harus melakukan tugas pengembangan dan integrasi dengan komponen lain. Pemrogram akan kembali ke tahap perancangan sistem jika prosesnya tidak berhasil memenuhi kebutuhan, kemudian dilanjutkan ke tahap berikutnya.
4. **Implementation** Pada fase implementasi ini, selama workshop analisis berkolaborasi erat dengan pengguna untuk menciptakan komponen komersial dan non-teknis organisasi. Setelah kesepakatan mengenai faktor-faktor ini dan pengembangan serta penyempurnaan sistem, sistem atau komponen sistem baru diuji sebelum diperkenalkan ke perusahaan[31].

2.2.6 *Unified Modeling Language (UML)*

Standar industri untuk memvisualisasikan, merancang, dan mendokumentasikan sistem perangkat lunak adalah Unified Modeling Language(UML). Standar untuk pemodelan sistem ditawarkan oleh UML. Sistem pengembangan perangkat lunak berorientasi objek (OO) biasanya divisualisasikan, didefinisikan, dibuat, dan didokumentasikan menggunakan




UML, yang umumnya disebut sebagai bahasa grafis atau berbasis gambar. Karena keberhasilannya dalam mensimulasikan sistem yang besar dan rumit, pemodelan UML sangat populer [32].

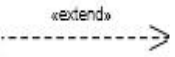

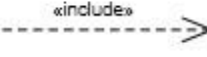
Unified Modeling Language (UML) memiliki langkah - langkah yaitu, membuat daftar bisnis dari proses terdahulu hingga selesai, melakukan pemetaan use case, membuat istilah yang mencakup semua proses, menjelaskan seluruh objek-objek yang ada, membuat rancangan user interface, membuat class diagram dan terakhir sudah dapat melakukan pembangunan sistem. Unified Modeling Language(UML) memiliki beberapa jenis, antara lain:

2.2.6.1 Use Case Diagram

Model perilaku sistem informasi masa depan adalah diagram use case. Interaksi antara satu atau lebih aktor dan sistem informasi yang akan datang dijelaskan dalam use case. Kasus penggunaan digunakan untuk menentukan fitur sistem informasi mana yang tersedia dan siapa yang diizinkan untuk menggunakannya[33]. Use case diagram memiliki beberapa simbol utama seperti yang ada pada tabel 2.2

Tabel 2. 2 Simbol Use Case Diagram




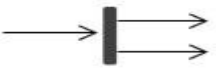
Nama	Simbol	Keterangan
Aktor		Orang proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari actor adalah gambar orang, biasanya dinyatakan menggunakan kata benda di awal frase nama actor.
Use Case		Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau actor biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama use case.
Associaton		Komunikasi antara actor dan use case yang berpartisipasi pada use case atau use case memiliki interaksi dengan actor.


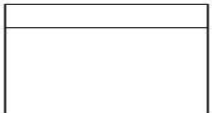
Extend		Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan dapat berdiri sendiri walau tanpa use case tambahan memiliki nama depan yang sama dengan use case yang ditambahkan.
Generalisasi		Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah use case dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
Include		Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan memerlukan use case ini untuk menjalankan fungsional atau sebagai syarat dijalankan use case ini

2.2.6.2 Activity Diagram

Alur kerja dari use case dapat direpresentasikan secara grafis menggunakan diagram aktivitas (Activity Diagram). Langkah-langkah proses, titik keputusan alur kerja yang bertanggung jawab untuk melakukan setiap tindakan, dan objek alur kerja semuanya digambarkan dalam diagram ini[34]. Activity diagram memiliki beberapa simbol utama seperti yang ada pada tabel 2.3

Tabel 2. 3 Simbol Activity Diagram

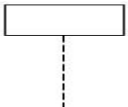



Simbol	Nama	Keterangan
	Status awal	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
	Aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
	Percabangan	Asosiasi percabangan dimana lebih dari satu aktivitas digabungkan menjadi satu.
	Penggabungan	Asosiasi penggabungan dimana lebih dari satu aktivitas lebih dari satu.

	Status Akhir	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status satu.
	Swimlane	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

2.2.6.3 Sequence Diagram

Sequence diagram adalah sebuah diagram yang menggambarkan kolaborasi dari objek-objek yang saling berinteraksi antar elemen dari suatu class[35]. Berikut merupakan komponen dalam sequence diagram :

Tabel 2. 4 Simbol Sequence Diagram

Simbol	Nama	Keterangan
	<i>Lifeline</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
	<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi tentang aktivitas yang terjadi
	<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi tentang aktivitas yang terjadi
	<i>Actor</i>	Menjelaskan tentang peran yang melakukan serangkaian aksi suatu proses

2.2.1 Black-box Testing

Metode Blackbox Testing merupakan Penguji dapat membuat serangkaian kondisi masukan dan menjalankan pengujian terhadap spesifikasi fungsional program untuk membuat pengujian yang berfokus pada spesifikasi fungsional perangkat lunak[36]. Pada metode Black Box Testing, aplikasi yang dikembangkan diuji dengan mencoba memasukkan data dalam segala bentuk. Untuk memastikan apakah perangkat lunak beroperasi sesuai dengan kebutuhan bisnis, diperlukan pengujian. Dua metode pengujian yang digunakan pada metode Blackbox Testing adalah

Boundary Value Analysis dan Equivalency Partitioning. Pendekatan Equivalency Partitioning diterapkan dalam penelitian ini. Equivalency Partitioning adalah teknik yang digunakan dalam pengujian black box. Untuk melakukan hal ini, perlu untuk memutuskan kasus uji, kriteria, partisi, data uji, kasus uji, serta pengujian dan penilaian[6].

Partisi, analisis nilai batas, grafik sebab-akibat, pengujian susunan ortogonal, pengujian transisi keadaan, dan fuzzing adalah beberapa contoh teknik pengujian kotak hitam. Selain itu, ada kelebihan dan kekurangan menggunakan pengujian black box. Menemukan kesenjangan dalam spesifikasi kebutuhan yang disediakan untuk pengembangan perangkat lunak adalah salah satu manfaatnya. Selain itu, pengujian tidak dapat diselesaikan karena pemahaman penguji yang tidak memadai terhadap perangkat lunak yang sedang dievaluasi, yang merupakan kelemahan pengujian kotak hitam.[37].

2.2.2 Metode System Usability Scale

Dengan sepuluh item pertanyaan, System Usability Scale (SUS) merupakan kuesioner yang dapat digunakan untuk menilai kegunaan sistem komputer dari sudut pandang subjektif pengguna SUS. Pengujian kegunaan dilakukan untuk mengetahui seberapa nyaman sesuatu bagi pengguna. Metode System Usability Scale (SUS) merupakan pendekatan pengukuran yang digunakan. SUS adalah tes di mana pengguna menilai perangkat lunak menggunakan skala Likert.[37].

Gunakan skala Likert 5 poin pada tabel 2.4 untuk menentukan SUS. Berdasarkan evaluasi subjektifnya, responden diminta mengurutkan 10 item pernyataan SUS. Salah satu alat untuk mengukur tingkat kepuasan pelanggan suatu produk adalah kuesioner SUS. Ada banyak aturan dalam menghitung skor di SUS. Untuk pertanyaan bernomor ganjil skor yang dijawab pada kuesioner dikurangi 1 (persamaan 1). Untuk pertanyaan genap, dikurangi 5 (persamaan 2). Kemudian semua skor dijumlahkan lalu dikalikan 2,5 (persamaan 3). Range nilai kuesioner adalah 0- 100. Skor rata-

rata SUS adalah 68, skor SUS di atas 68 berarti puas[38]. Dan instrumen pengujian yang terdapat pada System Usability Scale ditunjukkan dalam tabel 2.5

Tabel 2. 5 Instrumen Pengujian SUS

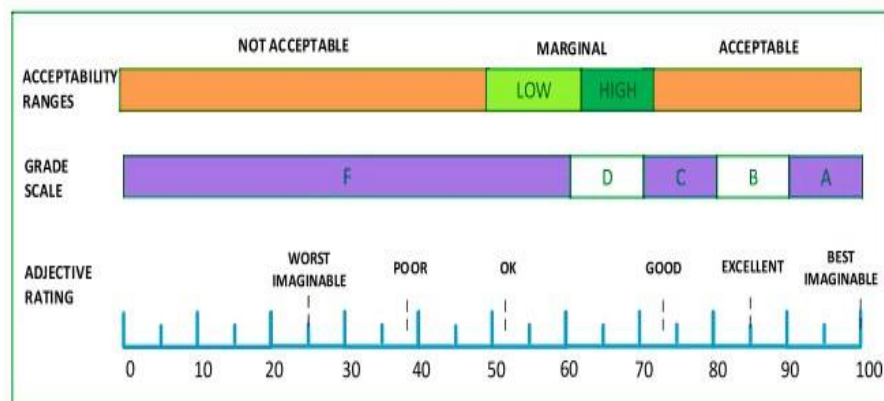
No	Pernyataan	Skala Likert
1	Saya akan sering menggunakan aplikasi ini	1 – 5
2	Saya rasa aplikasi ini tidak perlu dibuat serumit ini	1 – 5
3	Saya rasa aplikasi ini mudah untuk digunakan oleh siapapun	1 – 5
4	Saya rasa membutuhkan bantuan tenaga teknis dalam mengoperasikan aplikasi ini	1 – 5
5	Saya menemukan berbagai macam fitur yang dapat terintegrasi dengan baik	1 – 5
6	Saya rasa terdapat ketidaksesuaian dalam aplikasi ini	1 – 5
7	Saya rasa akan ada banyak orang yang dapat mempelajari aplikasi ini dengan sangat cepat	1 – 5
8	Saya menemukan aplikasi yang sangat rumit	1 – 5
9	Saya merasa yakin dalam menggunakan aplikasi ini	1 – 5
10	Saya perlu memahami aplikasi ini sebelum menggunakannya	1 – 5

SUS (System Usability Scale) memiliki aturan dalam perhitungan hasil, seperti yang ditunjukkan dalam Tabel 2.6

Tabel 2. 6 Aturan Perhitungan Hasil Pengujian SUS

No	Aturan	Rumus
1.	Setiap pernyataan bernomor ganjil yaitu 1,3,5,7,9 skala jawaban pertanyaan dikurangi 1	Skor SUS ganjil = $\sum Px-1$
2.	Setiap pernyataan bernomor genap yaitu 2,4,6,8,10 maka genap = $\sum 5-Pn$ 5 dikurangkan skala jawaban	Skor SUS
3.	Hasil penilaian skala 1 - 5 (5 merupakan jawaban terbaik)	-
4.	Melakukan penjumlahan jawaban kemudian dikali	$(\sum \text{skor ganjil} - \sum \text{skor genap}) \times \frac{2,5}{100}$ dengan 2,5 agar mendapatkan rentang nilai 0-100
4.	Menentukan nilai rerata jawaban instrumen pengujian $(\sum x)/n$ semua responden	$\bar{x} =$

Dalam SUS (*System Usability Scale*) untuk menentukan hasil perhitungan juga menggunakan tiga sudut pandang yakni *acceptability*, *grade-scale*, dan *adjective rating* seperti yang ditunjukkan dalam Gambar 2.2



Gambar 2. 2 Penentuan Hasil Perhitungan SUS[39].