

BAB II LANDASAN TEORI

2.1 Tinjauan Pustaka

Dalam penelitian ini akan digunakan lima tinjauan pustaka yang nantinya dapat mendukung penelitian, berikut ini merupakan tinjauan studi yang diambil yaitu:

Tabel 2.1 *Studi Literature*

1	Judul	Pemanfaatan Sistem Informasi Kearsipan Dalam Pengelolaan Arsip Inaktif di Kejaksaan Tinggi Jawa Tengah
	Penulis	Afrizal Prasetyo Nur Muhammad, Alamsyah
	Tanggal/Tahun	2018
	Tujuan Penelitian	Untuk mengetahui pemanfaatan sistem informasi kearsipan inaktif berdasarkan dua persepsi teori TAM, yaitu persepsi kemudahan penggunaan dan kebermanfaatan.
	Permasalahan	Sistem informasi arsip yang dimiliki belum terintegrasi antara sistem informasi yang berada di bagian arsip aktif dengan inaktifnya, sehingga informasi riwayat arsip menjadi kurang lengkap.
	Subjek Penelitian	Kejaksaan Tinggi Jawa Tengah
	Metode	<i>waterfall</i>
	Hasil Penelitian	Sistem Informasi Kearsipan Inaktif dimanfaatkan oleh petugas arsip untuk melakukan penomoran definitif, sebagai pengganti jadwal retensi arsip untuk menentukan masa simpan arsip, sarana temu kembali arsip, dan sebagai database arsip inaktif.

Tabel 2.1 *Studi Literature* (Lanjutan)

2	Judul	Pengembangan Sistem Informasi Manajemen Arsip Dinamis dalam Mendukung Tata Kelola Kearsipan berbasis Web menggunakan Metode SDLC
	Penulis	Riana Suprpto, Dedy Rahman Prehanto
	Tanggal/Tahun	2020
	Tujuan Penelitian	Merancang sebuah sistem yang dapat membantu proses pengelolaan arsip pada kantor desa karangrau banyumas
	Permasalahan	Dokumen surat yang sudah beredar baik surat masuk dan surat keluar selama ini masih bersifat aktif dan dapat diakses kapanpun. Namun pengelolaan arsip berdasarkan dokumen yang berjalan selingkung fakultas dan nonfakultas masih belum terdata atau terkumpul secara keseluruhan di bagian Tata Usaha kantor pusat Universitas Negeri Surabaya.
	Subjek Penelitian	Universitas Negeri Surabaya
	Metode	<i>SDLC (system Development Life Cycle)</i>
	Hasil Penelitian	Dikembangkan sebuah sistem pengelolaan kearsipan yang mendukung kelancaran aktivitas kearsipan selingkung Universitas Negeri Surabaya dimulai dari proses penciptaan surat yang dilakukan melalui sistem informasi persuratan elektronik yang dilakukan pada tiap-tiap unit baik fakultas, pasca sarjana, UPT maupun lembaga..
3	Judul	Sistem Informasi Pengelolaan Arsip Inaktif Berbasis Web Pada Badan Pengelola Keuangan dan Aset Daerah Kabupaten Tanah bambu.

Tabel 2.1 *Studi Literature* (Lanjutan)

	Penulis	Eko Handoyo, Dwi Agung Wibowo, Muhammad Edya Rosadi
	Tanggal/Tahun	2021
	Tujuan Penelitian	Membantu pekerjaan pengelola arsip dalam mengelola arsip inaktif serta dapat menampilkan laporan kinerja.
	Permasalahan	Arsip inaktif disimpan di gedung arsip hingga saat ini belum pernah diusulkan untuk dimusnahkan sehingga terjadi penumpukan berkas. Lokasi yang terpisah dari kantor utama mengakibatkan sulitnya pemantauan aktivitas kegiatan di gedung arsip.
	Subjek Penelitian	Badan Pengelolaan Keuangan dan Aset Daerah Kabupaten Tanah Bumbu
	Metode Penelitian	<i>Waterfall</i>
	Hasil Penelitian	Sistem informasi pengelolaan arsip inaktif untuk mendukung pengelolaan arsip inaktif di Badan Pengelolaan Keuangan dan Aset Daerah Kabupaten Tanah Bumbu.
4	Judul	Sistem Informasi Pengelolaan Arsip Inaktif Pada Politeknik Negeri Cilacap Berbasis Website
	Penulis	Raffi Riau Navallah
	Tanggal/ Tahun	2022
	Tujuan Penelitian	Merancang dan membangun sistem informasi pengelolaan arsip inaktif di Politeknik Negeri Cilacap berbasis website untuk memaksimalkan kinerja kearsipan agar lebih efektif, efisien dan menjamin integritas data yang memadai dalam bidang kearsipan.

Tabel 2.1 Studi Literature (Lanjutan)

	Permasalahan	Alur pengelolaan arsip inaktif di Politeknik Negeri Cilacap saat ini masih dilakukan secara konvensional sehingga menyebabkan waktu yang dalam dalam proses pencarian atau penyimpanan data, dan banyak resiko kesalahan dan ketidakteelitian.
	Subjek Penelitian	Politeknik Negeri Cilacap
	Metode Penelitian	<i>Waterfall</i>
	Hasil Penelitian	Sitem informasi pengelolaan arsip inaktif yang dapat mempermudah dan efisiensi waktu saat pemindahan arsip inaktif.
5	Judul	Analisis Perancangan Sistem Informasi Pengarsipan Surat Berbasis Web Pada Kantor Lurah Desa Dayah Tuha
	Penulis	Liza Rozana, Rahmat Musfekar
	Tanggal/Tahun	2020
	Tujuan Penelitian	Dengan adanya sistem ini diharapkan membantu kinerja, mempercepat dalam proses pengarsipan dan arsip yang dimiliki dapat tersimpan dengan baik dan aman
	Permasalahan	Pengarsipan dokumen masih menggunakan metode manual menggunakan buku arsip. Pengarsipan yang dilakukan secara manual mungkin masih bisa dilakukan jika data yang dimiliki jumlahnya masih sedikit, akan tetapi dokumen arsip akan terus bertambah seiring berjalannya waktu, yang membuat pekerjaan menjadi tidak efisien.

Tabel 2.1 Literatur Studi (Lanjutan)

Permasalahan	Selain membutuhkan banyak waktu, tenaga dan upaya dalam proses pengarsipannya, juga membutuhkan banyak tempat untuk penyimpanan. Selain itu dokumen dokumen yang hanya disimpan didalam lemari tanpa adanya data data cadangannya memungkinkan tercecer dan rawan rusak bahkan hilang jika terjadi suatu bencana seperti kebakaran atau banjir.
Subjek Penelitian	Kantor Lurah Desa Dayah Tuha
Metode Penelitian	<i>Research and Development (R&D)</i>
Hasil Penelitian	Merancang sebuah sistem informasi pengarsipan dokumen berbasis web yang sesuai dengan kebutuhan di kantor lurah desa dayah tuha

2.2 Sistem Informasi

Sistem informasi adalah kumpulan data yang tersimpan di basis data dengan model dan media teknologi informasi yang digunakan dalam pengambilan keputusan di suatu perusahaan, instansi, organisasi ataupun yang lainnya. Sistem adalah kumpulan prosedur yang saling berhubungan yang digabungkan untuk melakukan suatu tugas atau mencapai tujuan tertentu. Sedangkan Informasi biasanya didefinisikan sebagai proses mengubah data menjadi bentuk yang lebih bermanfaat dan menarik bagi orang yang menerimanya dan menggambarkan situasi nyata yang digunakan dalam proses pengambilan keputusan (Aswiputri, 2022).

Sistem informasi merupakan kumpulan orang, prosedur, *hardware* dan *software* yang berinteraksi satu sama lain untuk menyediakan suatu pelayanan informasi kepada pengguna. Tiga fungsi utama sistem informasi adalah sebagai berikut :

1. Menerima data (Input)
2. Mengubah data menjadi informasi (proses)

3. Memproduksi dan menyampaikan informasi dengan cepat sehingga pengguna dapat membuat keputusan.

Sebuah sistem informasi terdiri dari enam komponen utama yakni *hardware*, *software*, basis data, jaringan, prosedur, dan pengguna. Semua komponen dibarengkan sebagai berikut :

1. *Hardware*

Hardware adalah sekumpulan perangkat seperti *processor*, *monitor*, *keyboard*, dan *printer* yang menerima, memproses data, dan menampilkan hasil pengolahan.

2. *Software*

Software adalah kumpulan program komputer yang membuat hardware dapat memproses data.

3. Basis data

Basis data adalah himpunan kelompok data (arsip) yang saling berhubungan yang disusun sehingga dapat dimanfaatkan kembali dengan cepat dan mudah di masa depan.

4. Jaringan

Jaringan atau *network* adalah sistem penghubung yang mengizinkan terjadinya pembagian data dan sumber daya antara komputer-komputer yang berbeda.

5. Prosedur

Proses adalah rencana, peraturan, metode, dan prosedur yang digunakan untuk mengatur penggunaan sistem informasi.

6. Pengguna

Sistem informasi terdiri dari pengguna, yang termasuk orang yang menggunakannya atau menggunakan keluaran sistem informasi tersebut.

2.3 Arsip

Menurut Undang-Undang Republik Indonesia No.43 tahun 2009 pasal 1 ayat 3 tentang kearsipan, arsip merupakan “rekaman kegiatan atau peristiwa dalam berbagai bentuk dan media sesuai dengan perkembangan teknologi informasi dan komunikasi yang dibuat dan diterima oleh lembaga negara, pemerintahan daerah,

lembaga pendidikan, perusahaan, organisasi politik, organisasi kemasyarakatan, dan perseorangan dalam pelaksanaan kehidupan bermasyarakat, berbangsa dan bernegara”.

Arsip adalah bagian dari semua dokumen yang diterima atau dibuat oleh organisasi, instansi maupun perseorangan dan itu adalah kumpulan yang berisi tentang tindakan, keputusan, dan tindakan yang tendakan yang telah dilakukan oleh organisasi(Hayati, 2020) . Arsip dibagi menjadi 2 jenis yaitu :

a. Arsip Dinamis

Arsip dinamis yang digunakan secara langsung dalam kegiatan perkantoran sehari-hari, dibagi menjadi tiga kategori berdasarkan fungsi dan kegunaannya :

1. Arsip aktif, merupakan arsip yang masih banyak digunakan untuk kelangsungan kerja.
2. Arsip inaktif, merupakan arsip yang frekuensinya penggunaannya sudah menurun, atau jarang digunakan dalam proses pekerjaan.

b. Arsip Statis

Arsip statis merupakan arsip yang tidak digunakan secara langsung dalam kegiatan perkantoran sehari-hari dibagi menjadi tiga kategori yaitu :

1. Arsip aktif, merupakan arsip yang terus digunakan selama proses pekerjaan.
2. Arsip pasif, merupakan arsip yang jarang digunakan dalam proses penyelenggaraan kerja tetapi kadang-kadang masih diperlukan.
3. Arsip abadi, merupakan arsip yang disimpan untuk selamanya.

2.4 Arsip Inaktif

Arsip yang tidak lagi digunakan disebut arisp inaktif. Pada umumnya, ukuran penurunan frekuensi penggunaan arsip merujuk pada batas tertentu berapa banyak arsip yang digunakan dalam satu tahun(Arum, 2019). Satu tahun arsip dianggap aktif jika dirujuk lebih dari batas itu. Sebaliknya, arsip yang dirujuk kurang dari batas itu dianggap inaktif. Tiga langkah penting diperlukan untuk memastikan bahwa arsip inaktif dikelola dengan benar untuk kepentingan manajemen dan pengguna, yakni :

1. Jadwal retensi arsip (JRA)

JRA dibuat untuk memastikan bahwa arsip tetap tersimpan dalam jangka waktu tertentu.

2. Keputusan tentang media arsip

Apakah arsip akan disimpan dalam kertas, film, microfilm, magnetic, ataupun lainnya. Namun beberapa organisasi lebih suka menyimpan arsip dalam media kertas karena ini paling murah.

3. Menentukan fasilitas penyimpanan

Bagaimana memilih jenis fasilitas penyimpanan yang paling cocok untuk organisasi, instansi, maupun perserorangan. Beberapa biaya, rencana fasilitas dalam negeri, dan metode apa yang aman digunakan untuk memusnahkan arsip, adalah langkah yang harus diperhatikan saat memilih tempat untuk menyimpan arsip.

2.5 Website

Web adalah suatu media yang terdiri dari beberapa halaman yang saling berkaitan satu sama lain, dan berfungsi sebagai media untuk menampilkan suatu informasi, baik berbentuk gambar, video, teks, suara ataupun gabungan dari semuanya. Website bersifat multi platform yang artinya dapat dibuka dari segala perangkat atau device yang terhubung dengan jaringan internet.

Jadi dapat disimpulkan bahwa definisi web adalah suatu sistem yang memungkinkan pencarian informasi dan menampilkan teks, gambar, multimedia, dan elemen lainnya di jaringan internet(Suroso, 2021).

2.6 Database

Database adalah kumpulan data yang disimpan secara sistematis di dalam komputer yang dapat diolah atau diubah untuk menghasilkan informasi melalui penggunaan perangkat lunak, juga dikenal sebagai program aplikasi. Definisi database adalah gudang penyimpanan data yang akan diolah lebih lanjut dan meliputi spesifikasi berupa tipe data, struktur, dan batasan batasan data yang akan disimpan. Database sangat penting untuk mengorganisasi data, menghindari hubungan antar data yang tidak jelas, *update* yang rumit, dan duplikasi data(Nasution, 2023).

2.7 MySQL

MySQL adalah suatu RDBMS (*Relation Database Management System*), yaitu aplikasi sistem yang menjalankan fungsi pengolahan data. Keunggulan *MySQL* adalah praktis, handal, dan banyak forum yang membantu pengguna jika ada kendala. *MySQL* juga sering digabungkan dengan webserver, membuat instalansi lebih mudah. Sebagai kesimpulan dari teori diatas, dapat dikatakan bahwa *MySQL* adalah suatu aplikasi DBMS yang dapat menjalankan fungsi untuk mengolah suatu data (Nasution, 2023).

2.8 HTML

HTML juga dikenal sebagai *HyperText Markup Language* adalah program markup yang digunakan untuk membuat halaman web yang menampilkan berbagai informasi dalam penjelajahan web (Rachman, 2023).

2.9 PHP

PHP (*Hypertext Preprocessor*) adalah bahasa pemrograman yang umum digunakan untuk mengelola pembuatan dan pengembangan situs web. Ini dapat digunakan bersamaan dengan HTML (Noviana, 2022).

2.10 CSS

Cascading Style Sheet (CSS) merupakan aturan untuk mengatur beberapa bagian dari sebuah situs web agar lebih terorganisir. CSS bukan bahasa pemrograman. Sama halnya dengan *style* yang dapat diatur dalam aplikasi pengolahan kata seperti *microsoft word* yang memungkinkan berbagai *style*, seperti judul, subbab, *bodytext*, *footer*, gambar, dan lainnya untuk digabungkan dalam berbagai berkas atau file. Pada umumnya CSS digunakan untuk mengubah tampilan halaman web yang dibuat dengan bahasa HTML. CSS memiliki kemampuan untuk mengatur ukuran gambar, parameter seperti warna teks, warna tabel, *ukuran border*, warna *border*, spasi antar paragraf, spasi antar teks, *margin* kiri, kanan, atas, bawah, dan parameter lainnya (Saputra *et al.*, 2023).

2.11 Laravel

Laravel adalah sebuah *framework* PHP yang dapat digunakan untuk mengoptimalkan proses pembuatan website. Dengan menggunakan *framework* ini, hasil yang akan didapatkan sebuah website akan lebih dinamis. Dengan munculnya *framework* laravel, bahasa pemrograman PHP menjadi lebih kuat. Laravel memberikan fitur terbaru, selain itu laravel juga memiliki fitur bawaan yang lengkap termasuk fitur otentikasi. *Framework* laravel ini biasanya berkonsentrasi pada level *end-user*. Dalam hal ini, *framework* ini memiliki keunggulan tetapi tetap sederhana dalam hal penulisan kode dan tampilan. Meskipun demikian, *framework* laravel tetap bisa digunakan untuk membuat aplikasi berbasis *website* (Kevin *et al.*, 2023).

2.12 XAMPP

Xampp adalah perangkat lunak bebas yang berfungsi sebagai kumpulan program yang mendukung banyak sistem informasi dan berfungsi sebagai server yang berdiri sendiri (*localhost*). Ini terdiri dari program *programapace*, *HTTP server*, *database MySql*, dan penerjemah bahasa yang dibuat dengan *PHP* (Immanuel, 2022).

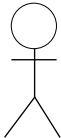


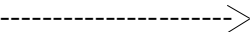
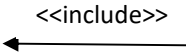
2.13 Perancangan UML (*Unified Modeling Language*)

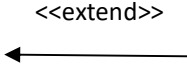
Unified Modelling Language (UML) merupakan suatu alat untuk memvisualisasikan dan mendokumentasikan hasil analisis dan desain yang berisi sintak yang digunakan untuk memodelkan sistem secara visual. Selain itu, termasuk kelompok konvensi pemodelan yang digunakan untuk mendefinisikan atau menggambarkan sistem software yang terkait dengan objek (Voutama, 2022). Tujuan utama UML diagram adalah untuk membantu tim pengembangan proyek berkomunikasi, mengevaluasi desain yang mungkin, dan memvalidasi desain arsitektur perangkat lunak atau pembuat program. Tiga kategori utama diagram *UML* adalah struktur, perilaku, dan interaksi. Masing-masing kategori memiliki diagram yang menjelaskan arsitektur sistem dan saling terintegrasi.

2.13.1 Use Case Diagram

Use case diagram merupakan titik awal yang baik dalam memahami dan menganalisis kebutuhan sistem pada saat perancangan. Penulis membuat *usecase* diagram untuk menunjukkan bagaimana interaksi dengan sistem pengelolaan arsip inaktif berbasis web. Simbol *usecase* diagram dapat diamati pada tabel 2.2.

Tabel 2.2 Simbol *Usecase Diagram*

Nama	Simbol	Keterangan
Aktor		aktor menunjukkan peran individu, sistem, atau alat.
Usecase		Interaksi antara aktor dan sistem serta abstraksi
Asosiasi		Abstraksi penghubung antara aktor dan usecase
Generalisasi		Menunjukkan spesialisasi aktor untuk berkontribusi pada kasus
Include		Menunjukkan bahwa fungsionalitas suatu use case secara keseluruhan identik dengan fungsionalitas suatu aplikasi lainnya



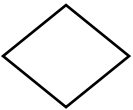


Nama	Simbol	Keterangan
Extend		Menunjukkan bahwa jika suatu kondisi terpenuhi, suatu use case dapat digunakan sebagai tambahan fungsional untuk use case lainnya

Sumber: (A.S., Rosa dan Shalahuddin, M., 2019)

2.13.2 Activity Diagram

Activity diagram adalah gambaran alur aktivitas sistem memberikan gambaran berakhir sistem. Disajikan pada tabel 2.3 Simbol-simbol yang ada pada *class diagram*.

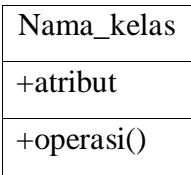



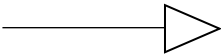
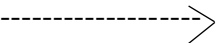
Tabel 2.3 Simbol *Activity Diagram*

Simbol	Deskripsi
Status awal 	Status awal aktivitas sistem ditunjukkan pada diagram aktivitas sistem
Aktivitas 	Aktivitas yang dilakukan dalam sistem biasanya dimulai dengan kata kerja
Penggabungan/ <i>decision</i> 	Asosiasi percabangan di mana Anda memiliki banyak pilihan aktivitas
Penggabungan/ <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu
Status Akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir

2.13.3 Class Diagram

Class diagram yang tersedia dalam UML adalah diagram kelas, yang menampilkan struktur, deskripsi class, atribut, metode, dan hubungan setiap objek. Namun, *class diagram* tidak menjelaskan apa yang terjadi jika kelas-kelasnya berhubungan, tetapi hanya menunjukkan hubungan yang terjadi. Simbol dalam kelas diagram disajikan dalam tabel 2.3.

Tabel 2.4 Simbol *Class Diagram*

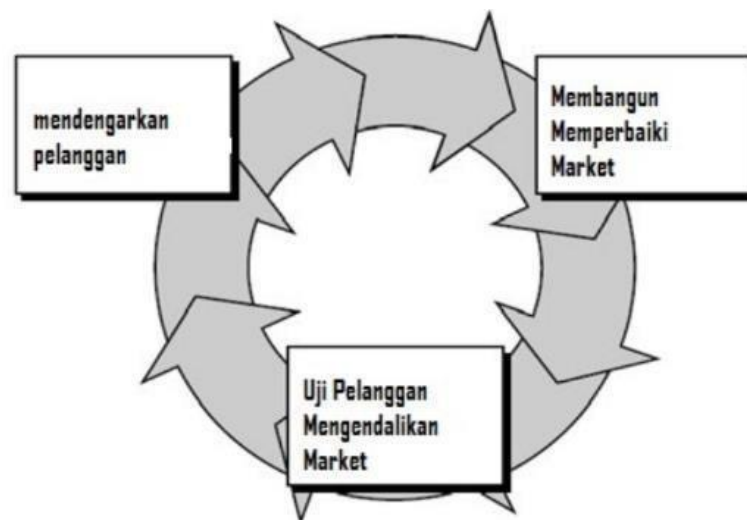
Simbol	Deskripsi
Kelas 	Kelas pada struktur sistem
Antarmuka 	Sama dengan konsep interfaces dalam pemrograman berorientasi objek.
Asosiasi/ <i>association</i> 	Relasi antarkelas dengan arti umum, asosiasi biasanya disertai dengan <i>multiplicity</i>
Asosiasi berarah/ <i>directed association</i> 	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas lain, asosiasi biasanya disertai dengan <i>multiplicity</i>
Generalisasi 	Relasi antarkelas dengan makna generalisasi spesialisasi (umum-khusus)
Kebergantungan/ <i>dependency</i> 	Ketergantungan antarkelas

2.14 Metode Pengembangan Sistem

Model *prototype* dapat digunakan untuk merancang sistem informasi. Model *prototype* memberikan pengembangan program dan objek penelitian kesempatan untuk berinteraksi satu sama lain selama proses perancangan sistem. Model *prototype* dapat digunakan untuk menyelesaikan masalah teknis yang tidak dipahami pelanggan dan untuk memberi tahu pengembang perangkat lunak tentang spesifikasi yang mereka butuhkan (Kevin *et al.*, 2023).

Model *prototype* dimulai dengan mengumpulkan informasi tentang apa yang diinginkan pelanggan tentang perangkat lunak yang akan dibuat. Selanjutnya, program *prototype* dibuat untuk memberi gambaran yang lebih baik tentang apa yang sebenarnya diinginkan. Program *prototype* biasanya masih dalam tahap pengembangan dan menawarkan tambahan melalui simulasi alur perangkat lunak untuk memberikan kesan perangkat lunak yang sudah jadi. Pelanggan atau user meninjau program *prototype* ini sampai pelanggan menemukan spesifikasi yang sesuai dengan keinginan pelanggan.

Berikut model *prototype* dapat dilihat pada gambar 2.1 berikut :



Gambar 2.2 Pengembangan *Prototype*

Sumber : (Roger S. Pressman Ph.D, 2010)

Dalam gambar diatas terdapat proses-proses dalam model *prototype* sebagai berikut :

1. Tahap pertama yaitu mendengarkan pelanggan dalam merancang sebuah sistem. Pada tahap ini akan menentukan informasi yang dibutuhkan oleh pelanggan agar tercipta sebuah aplikasi sehingga mengarah pada tujuan dibuatnya aplikasi tersebut.
2. Tahap kedua yaitu membangun dan memperbaiki *prototype*. Dalam tahap ini dilakukan perancangan dan pengkodean untuk sistem yang diusulkan yang mana tahapanya meliputi perancangan proses-proses yang akan terjadi di dalam sistem, perancangan diagram UML yang akan digunakan, perencanaan antar muka keluaran serta dilakukan tahap pengkodean terhadap rancangan yang telah ditentukan, kelengkapan *software* dan *hardware*.
3. Tahap ketiga yaitu pengujian *prototype*. Pada tahapan ini dilakukan pengujian terhadap sistem yang telah disusun dan melakukan pengenalan terhadap sistem yang telah diujikan serta evaluasi apakah sistem yang sudah jadi sudah sesuai dengan yang diharapkan.

2.15 Pengujian *Black Box Testing*

Black Box Testing merupakan pengujian yang menguji input dan output sesuai dengan spesifikasinya, sehingga tidak perlu memahami struktur internal *software*. Untuk memastikan bahwa *fungsi* aplikasi sudah memenuhi persyaratan, pengujian *Black Box* dilakukan untuk memverifikasi hasil eksekusi aplikasi berdasarkan data uji yang diberikan. *Black Box Testing* dilakukan untuk menemukan kesalahan perangkat lunak agar tidak terjadi lagi atau untuk memperbaikinya jika terjadi (Sakinah, 2024). Sebagaimana dinyatakan dalam metode pengujian *Black Box* terdiri dari berbagai pendekatan, seperti:

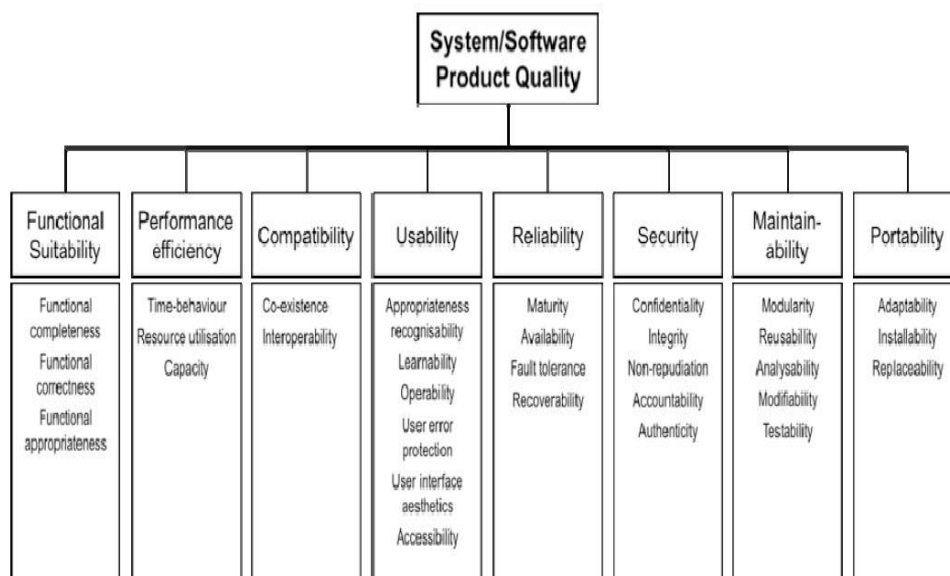
- a. Teknik *Equivalence Partitioning*: Teknik yang membagi input data menjadi beberapa partisi atau kelompok setara.
- b. Teknik *Boundary Value Analysis*: Teknik ini berfokus pada penemuan kesalahan di sekitar batas input minimum dan maksimum.
- c. Teknik *Fuzzing*: *Fuzzing* menemukan *bug* atau gangguan dalam perangkat lunak dengan menggunakan data cacat atau tidak *valid*.

- d. Teknik *Cause-Effect Graph*: Teknik ini menggunakan grafik untuk menunjukkan hubungan antara penyebab dan efek dalam perangkat lunak.
- e. Teknik *Orthogonal Array Testing* : Digunakan ketika domain input kecil tetapi kompleks, terutama untuk pengujian skala besar.
- f. Teknik *All Pair Testing*: Tujuan teknik ini untuk menghasilkan pasangan kasus uji yang mencakup setiap kombinasi diskrit yang dimungkinkan dari input parameter.

2.16 Pengujian ISO 25010

Standar ISO/IEC 25010 pertama kali diperkenalkan pada tahun 1991 melalui pertanyaan tentang definisi kualitas perangkat lunak. Dokumen standard ISO/IEC 25010 sangat panjang. Hal ini dikarenakan orang memiliki motivasi berbeda yang memungkinkan untuk tertarik pada kualitas perangkat lunak. Model ini merupakan bagian dari *software product quality requirements and evaluation (SquaRE)*, dimana model ini berkaitan dengan model kualitas perangkat lunak yang merupakan pengembangan dari model sebelumnya. Pada model ini terdapat beberapa sub-karakteristik tambahan dan beberapa sub-karakteristik yang dipindahkan ke karakteristik lain.

Karakteristik pada model ISO-25010 meliputi delapan karakteristik dapat dilihat pada gambar 2.1 berikut:



Gambar 2.1 Karakteristik ISO 25010

Sumber: (ISO/IEC 25010 dalam Kurniawati,2021)

1. *Functional Suitability*

Pengujian *functional suitability* merupakan tingkat dimana perangkat lunak dapat menyediakan fungsionalitas yang dibutuhkan ketika perangkat lunak digunakan pada kondisi yang spesifik.

2. *Performance Efficiency*

Pengujian *performance* ini dilakukan untuk mengukur karakteristik performa dari komponen aplikasi. Aspek ini merupakan aspek untuk mengukur keandalan sistem informasi yang digunakan pengembangan aplikasi sistem informasi.

3. *Compatibility*

Pengujian *compatibility* ini merupakan testing yang dilakukan dengan menjalankan produk, sistem atau komponen dengan perangkat keras dan/atau perangkat lunak dalam suatu sumber daya yang sama

4. *Usability*

Pengujian *usability* ini merupakan kemampuan perangkat lunak untuk dipahami, dipelajari, digunakan, dan menarik bagi pengguna, ketika digunakan dalam kondisi tertentu.

5. *Reliability*

Pengujian *reliability* ini merupakan kemampuan perangkat lunak untuk mempertahankan tingkat kinerja tertentu, ketika digunakan dalam kondisi tertentu.

6. *Security*

Pengujian *security* dilakukan untuk menguji keamanan aplikasi web terhadap serangan serangan atau *attack* kepada sistem.

7. *Maintainability*

Pengujian *maintainability* dilakukan untuk menguji efektifitas dan efisiensi perangkat lunak untuk dimodifikasi atau dikembangkan.

8. *Protability*

Pengujian *protability* ini merupakan kemampuan perangkat lunak untuk di transfer dari satu lingkungan lain.

Masing-masing karakteristik kualitas perangkat lunak model ISO 25010 dibagi menjadi beberapa sub-karakteristik kualitas. Tabel karakteristik kualitas perangkat lunak model ISO 25010 dapat dilihat pada tabel 2.5 berikut:

Tabel 2.5 Karakteristik ISO 25010

Karakteristik	Sub Karakteristik	Deskripsi
<i>Functional suitability</i>	<i>Functional Completeness</i>	Kemampuan perangkat lunak dalam menyediakan fungsi dapat mencakup semua tugas dan tujuan pengguna secara spesifik.
	<i>Functional Correctness</i>	Kemampuan perangkat lunak sejauh mana produk atau sistem mampu menyediakan hasil yang tepat sesuai dengan kebutuhan.
	<i>Functional Appropriateness</i>	Kemampuan perangkat lunak dalam fungsi yang disediakan mampu menyelesaikan tugas dan tujuan secara spesifik.
<i>Performance Efficiency</i>	<i>Time Behaviour</i>	Kemampuan perangkat lunak sejauh mana respon dan lama proses sebuah produk atau sistem saat menjalankan fungsinya sesuai dengan kriteria.
	<i>Resource Utilization</i>	Kemampuan perangkat lunak sejauh mana jumlah dan jenis sumber daya yang digunakan produk atau sistem saat menjalankan fungsinya sesuai dengan kriteria.
	<i>Capacity</i>	Kemampuan perangkat lunak sejauh mana batas maksimal sebuah produk atau sistem mampu memenuhi kriteria.
<i>Compatibility</i>	<i>Co-existence</i>	Kemampuan perangkat lunak sejauh mana produk atau sistem dapat menjalankan fungsi yang dibutuhkan secara efisien saat berbagi sumber daya dengan produk atau sistem lain tanpa memberi dampak terhadap produk atau sistem tersebut.

Karakteristik	Sub Karakteristik	Deskripsi
<i>Compatibility</i>	<i>Interoperability</i>	Kemampuan perangkat lunak sejauh mana dua atau lebih sistem, produk atau komponen dapat bertukar informasi dan menggunakan informasi tersebut.
<i>Usability</i>	<i>Appropriateness Recognizability</i>	Kemampuan perangkat lunak sejauh mana pengguna mengetahui sebuah produk atau sistem sesuai dengan kebutuhan mereka.
	<i>Learnability</i>	Kemampuan perangkat lunak sejauh mana produk atau sistem dapat digunakan pengguna tertentu untuk mencapai tujuan mempelajari sebuah produk atau sistem secara efektif, efisien dan bebas dari risiko serta memenuhi
<i>Usability</i>	<i>Operability</i>	Kemampuan perangkat lunak sejauh mana produk atau sistem mudah dioperasikan dan dikontrol.
	<i>User Error Protection</i>	Kemampuan perangkat lunak sejauh mana produk atau sistem melindungi pengguna dalam melakukan kesalahan.
	<i>User interface aesthetics</i>	Kemampuan sistem untuk mengukur seberapa efektif antarmuka pengguna dalam menciptakan pengalaman interaktif yang menyenangkan dan memuaskan bagi para pengguna.
	<i>Accessibility</i>	Kemampuan sistem untuk mengukur sejauh mana kemampuan dan karakteristik yang paling luas dari pengguna dapat dimanfaatkan dalam mencapai tujuan tertentu dalam suatu konteks penggunaan.

Karakteristik	Sub Karakteristik	Deskripsi
<i>Reliability</i>	<i>Maturity</i>	Kemampuan perangkat lunak sejauh mana sistem, produk atau komponen memenuhi kriteria reliabilitas dibawah kondisi normal.
	<i>Availability</i>	Kemampuan perangkat lunak sejauh mana sistem, produk, atau komponen.
	<i>Fault Tolerance</i>	Kemampuan perangkat lunak sejauh mana sistem, produk atau komponen.
	<i>Recoverability</i>	Kemampuan perangkat lunak ketika produk atau sistem mengalami kejadian atau kegagalan dapat mengembalikan data dan menjalankan kondisi sistem yang diharapkan.
<i>Security</i>	<i>Cofidentially</i>	Kemampuan perangkat lunak memiliki perlindungan terhadap data atau informasi dari pengguna, apakah sebagai ancaman atau kejanggalan.
	<i>Integrity</i>	Merupakan tingkat dimana kelengkapan dan ketepatan dari sejumlah asset telah dijaga.
	<i>Non-repudiation</i>	Kemampuan perangkat lunak dimana aksi atau tindakan yang dilakukan telah terbukti sehingga hal tersebut tidak dapat ditolak.
	<i>Accountability</i>	Kemampuan perangkat lunak dimana aksi dari sebuah entitas dapat ditelusuri keunikannya terhadap entitas.
	<i>Authenticity</i>	Sejauh mana identitas suatu subjek atau sumber daya dapat dibuktikan sebagai yang di klaim

Karakteristik	Sub Karakteristik	Deskripsi
<i>Maintainability</i>	<i>Modularity</i>	Kemampuan perangkat lunak dimana identitas dari sebuah subjek atau sumber dapat diterima.
	<i>Reusability</i>	Kemampuan perangkat lunak dimana asset dapat digunakan pada lebih dari satu sistem perangkat lunak atau pada pembangunan asset lainnya.
	<i>Analyzability</i>	Kemampuan perangkat lunak dimana perangkat lunak dapat dianalisis untuk mengetahui apa yang menyebabkan kegagalan pada perangkat lunak.
	<i>Modifiability</i>	Kemampuan perangkat lunak dimana perangkat lunak dapat menghindari efek yang tidak diharapkan dari modifikasi yang dilakukan terhadap perangkat lunak.
	<i>Testability</i>	Kemampuan perangkat lunak dimana perangkat lunak memungkinkan modifikasi perangkat lunak untuk dilakukan validasi.
<i>Portability</i>	<i>Adaptability</i>	Kemampuan perangkat lunak dapat beradaptasi dengan perubahan lingkungan atau sistem yang berbeda.
	<i>Installability</i>	Kemampuan perangkat lunak dapat digunakan dalam lingkungan atau sistem tertentu.
	<i>Replaceability</i>	Kemampuan perangkat lunak dapat menggantikan perangkat lunak lain apakah ada ketergantungan kepada perangkat lunak lain saat perangkat lunak tersebut digunakan

Adapun alasan penggunaan ISO 25010 karena ISO sudah berstandar *Internasional Organization for Standardization (ISO)* dan *International Electrotechnical Commission (IEC)*. Kualitas produk perangkat lunak ISO 25010 memiliki delapan karakteristik pendukung yang dapat digunakan sebagai acuan dalam menilai maupun memberikan masukan terhadap kualitas perangkat lunak yang akan dibangun dan akan menghasilkan nilai uji yang terukur. Dalam pengujian perangkat lunak ini penulis menggunakan suatu metode pengujian yang berfokus pada persyaratan fungsional dan kegunaan perangkat lunak dari sistem yang akan dibangun. Pengujian sistem ini akan diuji oleh pegawai pengarsipan dengan metode yang diambil yaitu metode pengujian ISO 25010 berdasarkan *functionality, reliability, security, usability, performance efficiency, dan operability*.

2.17 Skala Likert

Menurut Sugiono (2019) Skala Likert digunakan untuk mengukur sikap, persepsi, dan pendapat individu atau kelompok individu tentang fenomena sosial. Selain itu, skala Likert memiliki kemampuan untuk menawarkan solusi alternatif dari soal instrumen. Dengan gradasi dari sangat positif hingga sangat negatif, pertimbangan pemilihan pengukuran ini karena memudahkan responden untuk memilih jawaban. Data kualitatif diubah berdasarkan skor satu, dua, tiga, empat dan lima. Oleh karena itu, penelitian menggunakan skala likert untuk menilai sejauh mana responden setuju atau tidak setuju.

$$\text{Persentase Kepuasan}(100\%) = \frac{\text{skor yang diobservasi}}{\text{skor yang diharapkan}} \times 100\%$$

Berikut ini tabel skala likert dan bobot skor disajikan dalam tabel berikut.

Tabel 2.6 Skala Likert

No	Kategori	Skor
1	Sangat Setuju	5
2	Setuju	4
3	Kurang Setuju	3
4	Tidak Setuju	2
5	Sangat Tidak Setuju	1

Sumber : (Sugiyono, 2019)

Hasil persentase menunjukkan tingkat kepuasan pengguna dengan elemen-elemen yang diteliti. Arikunto (2019) terdapat lima skala yang digunakan untuk membagi kategori kepuasan. Skala ini memperhatikan rentang dari bilangan persentase. Nilai maksimal adalah 100% dan minimum 0%.

Tabel 2.7 Skor *Persentase*

No	Persentase (%)	Kategori Kepuasan
1	< 21%	Sangat Tidak Puas
2	21% - 40%	Tidak Puas
3	41% - 60%	Kurang Puas
4	61% - 80%	Puas
5	81% - 100%	Sangat Puas