

BAB II LANDASAN TEORI

2.1 Tinjauan Pustaka

Tinjauan pustaka merupakan sumber referensi terhadap penelitian terdahulu seperti berikut:

Tabel 1.1 Tinjauan Pustaka

No.	Nama dan Tahun	Judul	Metode	Hasil Dari Penelitian
1	(Kurniati, 2021)	Penerapan Metode Prototype Pada Perancangan Sistem Pengarsipan Dokumen Kantor Kecamatan Lais	Metode Prototype	Sistem pengarsipan dokumen mampu memberikan kemudahan kepada petugas dan admin kantor Kecamatan Lais dalam mengelola dokumen arsip dengan baik serta lebih mudah dalam proses pencarian dan keamanan data lebih terjamin
2	(Solikin and Putra, 2018)	Aplikasi E-Document Pada Kantor Kepala Desa Tugu Jaya Berbasis Website	Metode waterfall	Penelitian ini menghasilkan suatu aplikasi e-document pada kantor kepala desa Tugu Jaya Kabupaten Ogan Komering Ilir berbasis web yang bertujuan mempermudah penyimpanan, meningkatkan keamanan dokumen atau agar dokumen tidak rusak, mempermudah pencari dokumen bila diperlukan sewaktu-waktu
3	(Suroyo and Amin, 2017)	Aplikasi Sistem Manajemen Dokumen Elektronik Berorientasi Standar Borang BAN PT	Metode Unified Process (UP)	Penelitian ini telah menghasilkan sistem aplikasi Sistem Manajemen Dokumen Elektronik berorientasi standar borang akreditasi BAN PT

No.	Nama dan Tahun	Judul	Metode	Hasil Dari Penelitian
4	(Kartomiharjo <i>et al.</i> , 2019)	Analisis dan Perancangan Sistem Informasi Pelaporan Sampah (Studi Kasus pada Dinas Lingkungan Hidup dan Kebersihan Kabupaten Sidoarjo)	Pendekatan OOAD (Object Oriented Analysis and Design)	Setelah masyarakat melakukan pengaduan, Dinas Lingkungan Hidup dan Kebersihan dapat memeriksa pelaporan dan masyarakat dapat melihat status dari laporan yang mereka laporkan. Hal tersebut telah menyelesaikan masalah yang terdapat pada proses bisnis pelaporan sampah saat ini yang memiliki masalah pelaporan dilakukan dengan cara manual dan masyarakat juga tidak dapat mengetahui status dari laporan yang telah dilaporkan sehingga sulit untuk melakukan pengawasan terhadap laporan yang diberikan.
5	(Priabudi <i>et al.</i> , 2021)	Sistem Pelayanan Dokumen dan Pengaduan Lingkungan Hidup berbasis Web (Studi Kasus : Dinas Lingkungan Hidup Kota Malang)	Model waterfall	Di dalam website DLH kota Malang terdapat halaman yang hanya memuat mengenai informasi pelayanan tentang tata cara penyampaian aduan. Dengan adanya halaman tersebut, DLH kota Malang ingin memanfaatkan website tersebut untuk dapat berfungsi sebagai media pengaduan untuk menunjang kualitas dari pelayanan pengaduan
6	(Nasution <i>et al.</i> , 2021)	Urgency of Environmental Management System Implementation	Metode penelitian kualitatif	Sistem manajemen lingkungan yang terstandarisasi diperlukan oleh perusahaan untuk

No.	Nama dan Tahun	Judul	Metode	Hasil Dari Penelitian
		on Oil Palm Plantation Management Policies in North Sumatera		mengembangkan dan menerapkan aspek lingkungan
7	(Nkenyereye and Jang, 2017)	Design of Environmental Monitoring system for Auxiliary Data Center using Lower Hardware Cost	Metode penelitian kualitatif	Teknologi Informasi fisik dan fasilitas yang terkait dengan lingkungan pada pemantauan suhu, kelembaban, daya, banjir, asap, aliran udara, dan masuknya ruangan adalah cara paling proaktif untuk mengurangi biaya yang tidak perlu untuk penggantian perangkat keras yang mahal atau waktu henti yang tidak direncanakan.

Berdasarkan kelima jurnal tersebut peneliti mengacu pada penelitian oleh Dhany Sakti Priabudi, Denny Sagita Rusdianto dan Eriq Muhammad Adams Jonemaro (2021) yang membahas mengenai sistem pelayanan dokumen dan pengaduan lingkungan hidup dengan tujuan berupa pengolahan dokumen dan pelaporan pada bidang lingkungan hidup. Adapun perbedaan pada penelitian terletak pada bagian jenis pelaporan yaitu pemantauan lingkungan hidup dan metode yang digunakan berupa metode *prototype*.

2.2 Pelaporan Data

Menurut Maulidiansyah, *et al.*, (2017) pelaporan atau dapat disebut laporan merupakan bentuk penyampaian informasi baik secara lisan atau tulisan yang disampaikan dalam bentuk berita, pengumuman hingga pertanggungjawaban yang disesuaikan dengan kebutuhan pengguna.

Menurut Kartomiharjo *et al.*, (2019) dapat diartikan sebagai bentuk penyajian informasi yang dikemas atau ditampilkan dalam bentuk susunan data sebagai bentuk sumber yang akan digunakan untuk tujuan penggunaannya.

Sehingga pelaporan dapat disimpulkan dari definisi tersebut sebagai bentuk penyajian berupa informasi maupun data yang diberikan kepada pengguna sebagai bahan atau sumber keputusan.

2.3 *Electronic*

Menurut Munsarif *et al.*, (2022) *Electronic* merupakan teknologi yang digunakan dalam konteks internet, dalam hal media maka elektronik yang mengacu pada teknologi internet dapat diakses melalui website atau media online lainnya.

Menurut Oetomo and Maharginono (2020) *electronic* merupakan aktifitas penggunaan teknologi informasi dan komunikasi pengolahan digital dalam melakukan transaksi bisnis untuk menciptakan, mengidentifikasi dan dapat saling menghubungkan antar pengguna.

Sehingga *electronic* merupakan pemanfaatan teknologi informasi dalam jaringan internet dengan penerapan digitalisasi untuk pengolahan data.

2.4 *Document*

Menurut Sugiyono (2018) *Document* atau dokumen merupakan berupa informasi yang dapat berupa sumber data atau kumpulan informasi yang disajikan dalam bentuk file atau berkas.

Menurut Solikin and Putra (2018) Dokumen merupakan surat tertulis atau tercetak yang dapat dipakai sebagai bukti keterangan maupun rekaman suara atau gambar yang dapat disajikan kepada pengguna.

Sehingga *document* merupakan berkas atau file yang dapat berupa tercetak dengan kumpulan informasi yang dapat disajikan kepada pengguna.

2.5 E-Document

Menurut Ramdani (2018) *Electronic Document* merupakan informasi elektronik yang dibuat, diteruskan, dikirimkan, diterima atau disimpan dalam bentuk analog, digital, elektromagnetik, optikal atau sejenisnya yang dapat dilihat, ditayangkan, didengar melalui komputer atau sistem elektronik, termasuk tidak terbatas pada tulisan, suara, gambar, peta, rancangan, foto, huruf, tanda, angka, kode akses, simbol, preformasi yang memiliki makna atau arti dapat dipahami oleh pengguna.

Menurut Solikin and Putra (2018) *e-document* adalah suatu konten elektronik seperti program atau file komputer yang membutuhkan teknologi elektronik untuk dapat digunakan, dibaca atau dilihat.

Sehingga *electronic document* merupakan bentuk pemanfaatan teknologi yang dapat dilihat dan digunakan kembali.

2.6 Prototype

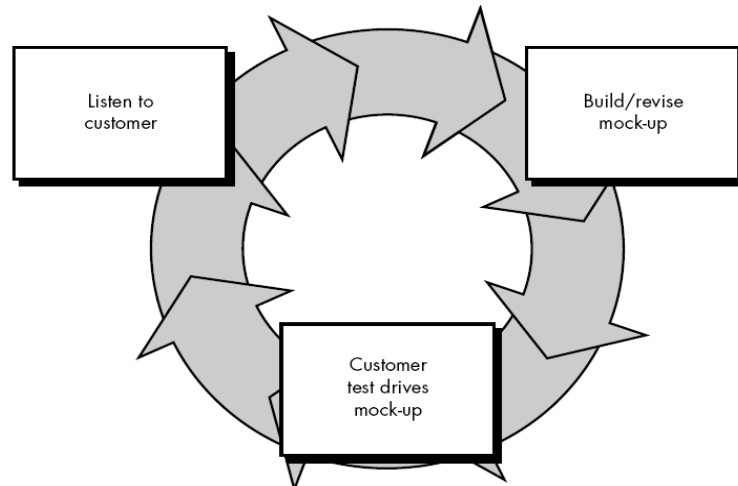
Menurut Rosa and Shalahuddin (2019) model prototipe cocok digunakan untuk menggali spesifikasi kebutuhan pelanggan secara lebih detail tetapi beresiko tinggi terhadap membengkaknya biaya dan waktu proyek". Model

pengembangan perangkat lunak yang digunakan ini menggunakan model prototype.

Model prototype adalah “suatu proses yang memungkinkan developer membuat sebuah model software, metode ini baik digunakan apabila client tidak bisa memberikan informasi yang maksimal mengenai kebutuhan yang diinginkannya”. Mock-up yang dimaksud pada tahapan prototipe di atas, adalah sesuatu yang digunakan sebagai model desain yang digunakan untuk mengajar, demonstrasi, evaluasi desain, promosi atau keperluan lain.

Mock-up disebut sebagai prototipe apabila perangkat lunak tersebut menyediakan atau mampu mendemonstrasikan sebagian besar fungsi sistem perangkat lunak dan memungkinkan pengujian desain sistem perangkat lunak. Maka dari itu dapat disimpulkan bahwa model prototipe merupakan model dari metode pengembangan perangkat lunak yang digunakan untuk menjabarkan 15 kebutuhan pelanggan secara lebih detail karena pelanggan sering kali kesulitan menyampaikan kebutuhannya secara detail tanpa melihat gambaran yang jelas tetapi beresiko tinggi terhadap pembengkakan biaya dan waktu proyek.

Menurut Purnomo (2017) metode prototype digunakan untuk tujuan mengumpulkan informasi dari pengguna sehingga pengguna dapat berinteraksi dengan model prototypenya sehingga hasil sistem yang dibangun sesuai dengan keinginan penggunanya.



Gambar 1.1 Metode Prototype
Sumber: (Rosa and Shalahuddin, 2019)

Model prototipe ini memiliki beberapa tahapan dalam pelaksanaannya.

Tahapan dari model prototipe, diuraikan sebagai berikut:

1. Mendengarkan pelanggan Model prototipe dimulai dari mengumpulkan kebutuhan pelanggan terhadap perangkat lunak yang akan dibuat.
2. Membangun atau memperbaiki mock-up Setelah mendengarkan kebutuhan pelanggan, maka dibuatlah program prototipe agar pelanggan lebih terbayang dengan apa yang sebenarnya diinginkan. Program prototipe biasanya merupakan program yang belum jadi dan menyediakan tampilan dengan simulasi alur perangkat lunak sehingga tampak seperti perangkat lunak yang sudah jadi.
3. Pelanggan melihat atau menguji mock-up Program prototipe ini dievaluasi oleh pelanggan atau user sampai ditemukan spesifikasi yang sesuai dengan keinginan perusahaan.

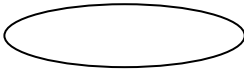
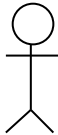

2.7 Unified Modelling Language (UML)

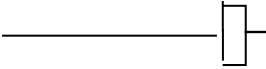
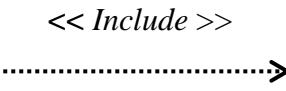
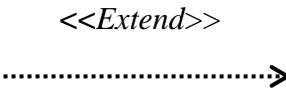
Menurut Rosa and Shalahuddin (2019) UML (*unified Modelling Language*) adalah bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. Berikut ini merupakan penjelasan tentang masing-masing diagram yang ada pada UML (*Unified Modelling Language*).

2.7.1 Use Case Diagram

Menurut Rosa and Shalahuddin (2019) *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. Berikut simbol-simbol yang akan digunakan dalam menggambarkan *Use Case Diagram* dapat dilihat pada Tabel 2.2.

Tabel 1.2 Simbol *Use Case Diagram*

No	Simbol	Deskripsi
1.		<i>Usecase</i> Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal <i>frase</i> nama <i>use case</i> .
2.		Aktor Aktor seseorang/sesuatu yang berinteraksi dengan yang akan dibuat. diluar sistem informasi. Biasanya dinyatakan menggunakan kata benda
3.		Asosiasi/association merupakan komunikasi antara aktor dan use case yang berpartisipasi pada use case memiliki interaksi dengan aktor.


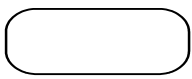
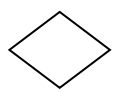

No	Simbol	Deskripsi
4.		Generalisasi (<i>generalization</i>) merupakan hubungan (umum – khusus) antara dua buah use case dimana fungsi yang satu adalah fungsi yang lebih umum
5.		Include berarti use case yang ditambahkan akan dipanggil saat use case tambahan dijalankan.
6.		Ekstensi (<i>extend</i>) merupakan use case tambahan ke sebuah use case yang ditambahkan dapat berdiri sendiri walau tanpa use case tambahan itu.

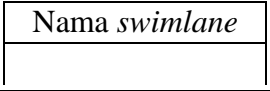

Sumber: (Rosa and Shalahuddin, 2019)

2.7.2 Activity Diagram

Menurut Rosa and Shalahuddin (2019) *activity* diagram menggambarkan aliran kerja atau aktivitas dari sebuah sistem atau proses bisnis atau menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem. Berikut simbol-simbol yang akan digunakan dalam menggambarkan *activity diagram* dapat dilihat pada Tabel 2.3 :

Tabel 1.3 Simbol *Activity* Diagram

No.	Simbol	Keterangan
1.		Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
2.		Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3.		Percabangan (<i>Decision</i>) merupakan asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
4.		Penggabungan (<i>Join</i>) merupakan asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.

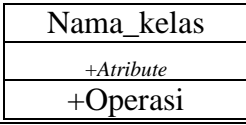


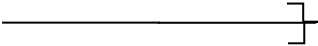
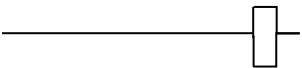
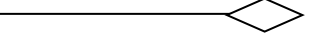
No.	Simbol	Keterangan
5.		Swimlane Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas.
6.		Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.

Sumber: (Rosa and Shalahuddin, 2019)

2.7.3 Class Diagram

Menurut Rosa and Shalahuddin (2019) *Class diagram* mengembangkan struktur sistem dari segi pendefinisian. Berikut simbol-simbol yang akan digunakan dalam menggambarkan *Class Diagram* dapat dilihat pada Tabel 2.4.

Tabel 1.4 Simbol *Class Diagram*

No.	Simbol	Deskripsi
1.		Kelas pada struktur sistem.
2.	Antar Muka/ <i>Interface</i> 	Sama dengan konsep interface dalam pemrograman berorientasi objek.
3.	<i>Asosiasi / Association</i> 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan symbol
4.	<i>Asosiasi Berarah / Directed Association</i> 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan symbol.
5.	Generalisasi 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus)
6.	<i>Agregasi / aggregation</i> 	Relasi antar kelas dengan maksna semua bagian (<i>whole-part</i>)

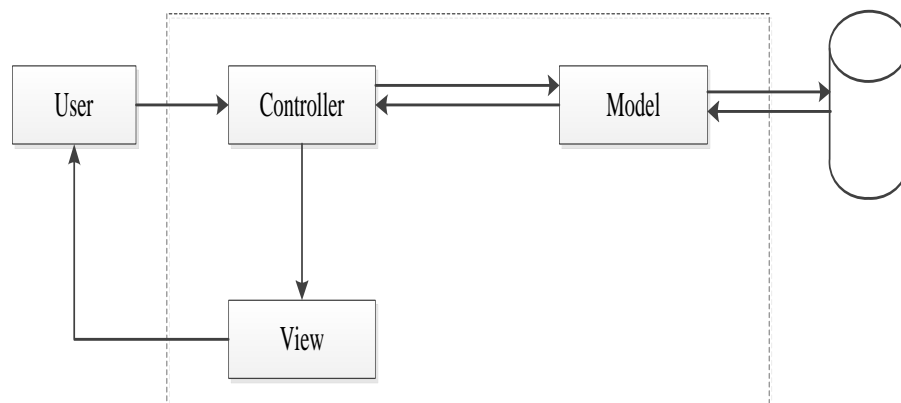
Sumber: (Rosa and Shalahuddin, 2019)

2.8 CodeIgniter

Menurut Raharjo (2018), *CodeIgniter* adalah *Framework* untuk bahasa pemrograman PHP, yang dibuat Rick Ellis pada tahun 2006. *CodeIgniter* memiliki banyak fitur yang membantu para pengembang PHP untuk dapat membuat aplikasi secara mudah dan cepat serta memiliki sifat yang fleksibel dapat mengembangkan dalam perangkat *web*, *dekstop* maupun *mobile*.

Menurut Heru (2018) *codeigniter* adalah sebuah aplikasi gratis yang berupa kerangka kerja untuk membangun website menggunakan bahasa pemrograman PHP.

CodeIgniter memiliki konsep atau pola *Model-View-Controller* (MVC) sehingga kode-kode dapat di sederhanakan.



Gambar 1.2 Arsitektur MVC

Sumber : (Raharjo, 2018),

2.8.1 PHP (*Hypertext Preprocessor*)

Menurut Heru (2018) PHP (*Hypertext Preprocessor*) merupakan bahasa pemrograman untuk membuat website atau situs dinamis dan mengenai rangkaian bahasa pemrograman antara *client side scripting* dan *server side scripting*.

Menurut Oetomo and Maharginono (2020) merupakan bahasa pemrograman yang digunakan secara luas untuk penanganan, pembuatan dan pengembangan sebuah situs web dan biasanya bersamaan dengan HTML.

Sehingga PHP merupakan bahasa pemrograman yang digunakan oleh pengembang untuk membuat sistem *website* dengan kumpulan bahasa HTML dan *script* lainnya.

2.8.2 MySql

Menurut Setyawan dan Pratiwi (2019) MySQL adalah sebuah *database management system* (manajemen basis data) menggunakan perintah dasar SQL (*Structured Query Language*) yang cukup terkenal. *Database management system* (DBMS) MySQL multi pengguna dan bersifat gratis. Mysql digunakan sebagai wadah dalam mengelola data yang dapat disimpan digunakan kembali dengan cara yang lebih efisien

2.9 Metode Pengujian Sistem

Metode pengujian sistem merupakan metode yang digunakan untuk melakukan testing pada sistem yang dibangun sehingga di peroleh hasil berupa sistem yang sesuai fungsinya.

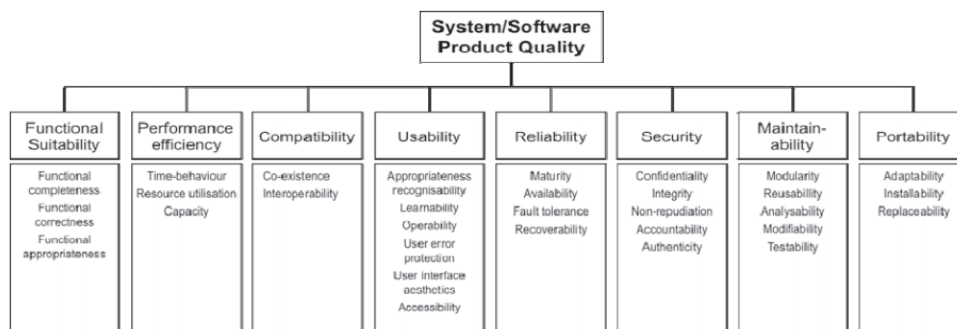
2.9.1 ISO 25010

Menurut *International Organisation for Standardisation* (2011), ISO/IEC 25010 adalah Model kualitas sistem dan perangkat lunak yang menggantikan ISO/IEC 9126 tentang *software engineering* sebagai bentuk konsep pengujian terhadap kelayakan suatu sistem yang telah dibangun.

Product quality ini juga digunakan untuk tiga model kualitas yang berbeda untuk produk perangkat lunak antara lain:

1. Kualitas dalam model penggunaan,
2. Model kualitas produk, dan
3. Data model kualitas

Model kualitas produk terdiri dari delapan karakteristik yang berhubungan dengan sifat statis perangkat lunak dan sifat dinamis dari sistem komputer. Model ini berlaku untuk sistem komputer dan produk perangkat lunak. Karakteristik yang didefinisikan oleh kedua model tersebut relevan untuk semua produk perangkat lunak dan sistem komputer. Karakteristik dan subkarakteristik memberikan terminologi yang konsisten untuk menentukan, mengukur dan mengevaluasi kualitas sistem dan perangkat lunak. Mereka juga menyediakan seperangkat karakteristik kualitas yang sesuai dengan persyaratan kualitas yang dapat dibandingkan untuk kelengkapan.



Gambar 1.3 ISO 25010

Sumber : (*International Organisation for Standardisation, 2011*)

Berdasarkan tahapan-tahapan ISO 25010 tersebut maka peneliti menggunakan pengujian terhadap kualitas perangkat lunak berupa aplikasi *web* dapat dinilai dari empat aspek yaitu:

1. *Functional Suitability*

Sejauh mana suatu produk atau sistem menyediakan fungsi yang memenuhi kebutuhan yang dinyatakan dan tersirat ketika digunakan di bawah kondisi tertentu. Sub pengujian tersebut sebagai berikut:

a. *Functional Completeness* (Kelengkapan Fungsional)

Sejauh mana rangkaian fungsi mencakup semua tugas dan tujuan pengguna yang ditentukan

b. *Functional Correctness* (Kebenaran Fungsional)

Sejauh mana suatu produk atau sistem memberikan hasil yang benar dengan tingkat presisi yang dibutuhkan.

Alat pengujian yang digunakan yaitu kuisioner yang memiliki 2 jawaban pertanyaan seperti sukses atau gagal.

2. *Usability*

Sejauh mana suatu produk atau sistem dapat digunakan oleh pengguna tertentu untuk mencapai tujuan tertentu dengan efektivitas, efisiensi, dan kepuasan dalam konteks penggunaan tertentu. Sub pengujian tersebut sebagai berikut :

a. *Appropriateness Recognizability*

Sejauh mana pengguna dapat mengenali apakah suatu produk atau sistem sesuai dengan kebutuhan mereka.

b. *Learnability*

Sejauh mana suatu produk atau sistem dapat digunakan oleh pengguna tertentu untuk mencapai tujuan pembelajaran yang ditentukan untuk

menggunakan produk atau sistem dengan efektifitas, efisiensi, bebas dari resiko dan kepuasan dengan tujuan tertentu konteks penggunaan.

Alat pengujian yang digunakan yaitu kuisioner yang memiliki 5 jawaban pertanyaan seperti Sangat Setuju, Setuju, Ragu-ragu, Tidak Setuju dan Sangat Tidak Setuju.

3. *Portability*

Tingkat efektivitas dan efisiensi dimana sistem, produk atau komponen dapat ditransfer dari satu perangkat keras, perangkat lunak atau lingkungan operasional atau penggunaan lainnya ke yang lain. Berikut sub pengujian tersebut:

a. *Adaptability*

Sejauh mana suatu produk atau sistem dapat secara efektif dan efisien diadaptasi untuk perangkat keras, perangkat lunak, atau lingkungan operasional atau penggunaan yang berbeda atau berkembang.

b. *Coexistence*

Sejauh mana sistem dapat bekerja pada perangkat lunak yang ada.

Alat yang digunakan untuk pengujian tersebut berupa *cross browsing compability testing* seperti *chrome*, *firefox* dan *opera*.

4. *Performance efeciency.*

kinerja relatif terhadap jumlah sumber daya yang digunakan dalam kondisi yang dinyatakan, berikut sub pengujian tersebut:

a. *Time Behaviour*

Sejauh mana respon dan waktu pemrosesan dan tingkat throughput dari suatu produk atau sistem, ketika menjalankan fungsinya, memenuhi persyaratan

b. *Resource Utilization*

Sejauh mana jumlah dan jenis sumber daya yang digunakan oleh suatu produk atau sistem, ketika menjalankan fungsinya, memenuhi persyaratan

Alat pengujian yang digunakan yaitu dengan melakukan pengecekan performa sistem pada media web test secara online sehingga mampu menampilkan detail performa sistem, adapun link web test yaitu <https://www.webpagetest.org/>.

5. *Compatibility*

Sejauh mana sebuah produk, sistem atau komponen dapat bertukar informasi dengan produk, sistem atau komponen dan/atau menjalankan fungsi lain yang diperlukan secara bersamaan ketika berbagi perangkat keras dan *environment* perangkat lunak yang sama. Karakteristik ini dibagi menjadi 2 karakteristik yaitu.

- a. *Co-existence*, sejauh mana produk atau sistem dapat menjalankan fungsi yang dibutuhkan secara efisien sementara berbagi sumber daya dengan produk atau sistem yang lain tanpa merugikan produk atau sistem tersebut.
- b. *Interoperability*, sejauh mana dua atau lebih produk, sistem atau komponendapat bertukar informasi dan menggunakan informasi tersebut.

6. *Reliability*

Sejauh mana sebuah sistem, produk atau komponen dapat menjalankan fungsi tertentu dalam kondisi tertentu selama jangka waktu yang ditentukan.

Karakteristik ini terbagi menjadi beberapa subkarakteristik yaitu.

- a. *Maturity*, sejauh mana produk atau sistem mampu memenuhi kebutuhan secara handal di bawah keadaan normal.
- b. *Availability*, sejauh mana produk atau sistem siap beroperasi dan dapat diakses saat perlu digunakan.
- c. *Fault tolerance*, sejauh mana produk atau sistem tetap berjalan sebagaimana yang dimaksud meskipun terjadi kesalahan pada perangkat keras atau perangkat lunak.
- d. *Recoverability*, sejauh mana produk atau sistem mampu dapat memulihkan data yang terkena dampak secara langsung dan menata ulang kondisi sistem seperti yang diinginkan ketika terjadi gangguan.

7. *Security*

Sejauh mana sebuah produk atau sistem melindungi informasi dan data sehingga seseorang atau sistem lain dapat mengakses data sesuai dengan jenis dan level otorisasi yang dimiliki. Karakteristik ini terbagi menjadi beberapa karakteristik yaitu.

- a. *Confidentiality*, sejauh mana produk atau perangkat lunak memastikan data hanya bisa diakses oleh mereka yang berwenang untuk memiliki akses.
- b. *Integrity*, sejauh mana produk atau perangkat lunak mampu mencegah akses yang tidak sah untuk memodifikasi data.

- c. *Non-repudiation*, sejauh mana peristiwa atau tindakan dapat dibuktikan telah terjadi, sehingga tidak ada penolakan terhadap peristiwa atau tindakan tersebut.
- d. *Accountability*, sejauh mana tindakan dari suatu entitas dapat ditelusuri secara unik untuk entitas.
- e. *Authenticity*, sejauh mana identitas subjek atau sumber daya dapat terbukti menjadi salah satu yang diklaim.

8. *Maintainability*

Sejauh mana keefektifan dan efisiensi dari sebuah produk atau sistem dapat dirawat. Karakteristik ini terbagi menjadi beberapa sub karakteristik yaitu.

- a. *Modularity*, sejauh mana sistem terdiri dari komponen terpisah sehingga perubahan atau modifikasi pada salah satu komponen tersebut memiliki dampak yang kecil terhadap komponen yang lain.
- b. *Reusability*, sejauh mana aset dapat digunakan lebih oleh satu sistem atau digunakan untuk membangun aset lain.
- c. *Analyzability*, tingkat efektivitas dan efisiensi untuk mengkaji dampak perubahan pada satu atau lebih bagian-bagian produk atau sistem, untuk mendiagnosis kekurangan atau penyebab kegagalan produk, untuk mengidentifikasi bagian yang akan diubah.
- d. *Modifiability*, sejauh mana produk atau sistem dapat dimodifikasi secara efektif dan efisien tanpa menurunkan kualitas produk yang ada.

Testability, tingkat efektivitas dan efisiensi untuk membentuk kriteria uji dari produk, sistem atau komponen dan uji dapat dilakukan untuk menentukan apakah kriteria tersebut telah terpenuhi.