

BAB II

LANDASAN TEORI

2.1 Tinjauan Pustaka

Berikut ini adalah beberapa tinjauan pustaka yang dilakukan oleh penulisan pada penelitian sebelumnya untuk menjadi pendukung penelitian yang sedang dilakukan:

Tabel 2. 1 Daftar Literatur

No	Nama Peneliti	Tahun	Judul
1	(Sikumbang, Muhammad Arya Rosyd, Habib, Roni, Pane, Syafrial Fachri)	2020	Sistem Informasi Absensi Pegawai Menggunakan Metode <i>RAD</i> dan Metode <i>LBS</i> Pada Koordinat Absensi.
2	(Rubiati, Nur Harahap, Sahara Widya)	2019	Aplikasi Absensi Siswa Menggunakan <i>Qr Code</i> Dengan Bahasa Pemrograman Php Di Smkit Zunurain Aqila Zahra Di Pelintung.
3	(Rahmalisa, Uci Irawan, Yuda Wahyuni, Refni)	2020	Aplikasi Absensi Guru Pada Sekolah Berbasis <i>Android</i> Dengan Keamanan <i>Qr Code</i> (Studi Kasus : Smp Negeri 4 Batang Gansal).
4	(Hutabri, Ellbert)	2022	Perancangan Dan Implementasi Aplikasi Absensi Berbasis <i>Qr Code</i> Dengan <i>Lock Gps</i> .
5	(Qois, Naviza Jumaryadi, Yuwan)	2021	Implementasi <i>Location Based Service</i> Pada Sistem Informasi Kehadiran Pegawai Berbasis <i>Android</i> .

2.1.1 Literatur 1

Absensi kehadiran berperan penting dalam sebuah perusahaan untuk meningkatkan kedisiplinan pegawai. Kedisiplinan karyawan menjadi tolak ukur utama dalam melihat kinerja karyawan berdasarkan kehadirannya di perusahaan. Seperti halnya absensi di Perusahaan Badan Pusat Statistik Kota Bandung yang menerapkan teknologi sistem absensi menggunakan teknologi sidik jari. Masalah yang sering terjadi di Badan Pusat Statistik Kota Bandung adalah pada saat pegawai ditugaskan untuk melakukan perjalanan dinas, seringkali pegawai tersebut tidak hadir terlebih dahulu ke kantor. Dikarenakan besarnya beban kerja yang diterima karyawan dari masing-masing divisi sehingga mempengaruhi terhadap kinerja pegawai dalam peraturan perusahaan.

Dari permasalahan tersebut, untuk meminimalisir kemungkinan terjadinya kendala dalam proses absensi, akan dibangun aplikasi sistem informasi absensi karyawan dengan menerapkan koordinat pada *platform android*. Dengan menggunakan aplikasi ini diharapkan bisa menambah pengetahuan dan wawasan pegawai dalam pengembangan ilmu dan pemanfaatan sistem informasi. Sehingga karyawan tidak perlu dipusingkan lagi untuk melakukan absensi terlebih dahulu ke kantor di penghujung hari saat bertugas dari kantor. Sehingga tidak mengganggu kinerja dari karyawan (Sikumbang dkk., 2020).

2.1.2 Literatur 2

Absensi ialah salah satu penunjang utama yang dapat mendukung serta memotivasi setiap kegiatan yang dilakukan. Seperti absensi pada SMKIT Zunurain

Aqila Zahra pengambilan daftar kehadiran dilakukan dua sampai dengan empat kali dalam proses belajar satu hari, dengan perhitungan jika dalam satu jam mata pelajaran siswa siswi tidak hadir atau tanpa keterangan (alpha) maka siswa siswi tersebut dianggap tidak hadir dalam satu hari kegiatan pembelajaran. Berdasarkan pengamatan yang dilakukan di SMKIT Zunurain Aqila Zahra, proses absensi yang dilakukan sekretaris kelas kurang efektif dalam pelaksanaannya, data absen yang ada rusak disebabkan oleh kondisi buku, faktor alam dan unsur ketidak sengajaan dalam memberi rekapitulasi hasil absensi kepada wali kelas. Ketidak sesuaian data kehadiran dapat merugikan siswa.

Tujuan penelitian ini adalah membuat sistem Absensi Siswa menggunakan *QR Code* dengan bahasa pemrograman PHP yang akan diterapkan pada SMKIT Zunurain Aqila Zahra Medang Kampai, yang mana dari segi ini penjelasan dari *QR Code*, kependekan dari *Quick Response Code*, adalah gambar dua dimensi yang memiliki kemampuan untuk menyimpan data. Kode *QR* biasanya digunakan untuk menyimpan data dalam bentuk teks, baik itu numerik, kode alfanumerik atau kode biner (Rubiati & Harahap, 2019).

2.1.3 Literatur 3

absensi guru di SMP Negeri 4 Batang Gansal yang melakukan tanda tangan di kertas serta menuliskan waktu kehadiran untuk setiap jam mengajar, yang dimana dengan cara ini dapat berdampak pada kecurangan karena tingkat disiplin yang tidak terkendali. Tata usaha juga sulit untuk membuat laporan serta memberikan informasi absensi kepada masing-masing guru mengenai pencatatan

absensi dan juga pencatatan waktu kerja karena laporan masih dibuat menggunakan *Microsoft Excel*, Dengan cara ini, dapat terjadi kesalahan ketik serta terhapusnya data. Dan alasan penulis melakukan penelitian di SMP Negeri 4 Batang Gansal karena ingin membangun serta mengembangkan ilmu teknologi di desa sendiri, dan mengajarkan ilmu yang didapat selama kuliah di STMIK Hang Tuah Pekanbaru agar bermanfaat.

Pemanfaatan sistem teknologi informasi yang semakin maju dapat mengatasi hal-hal yang tidak diinginkan di atas dengan hadirnya sistem absensi digital yang dapat memberikan pencapaian kinerja kepada sekolah (Rahmalisa dkk., 2020).

2.1.4 Literatur 4

Kehadiran karyawan di toko mempunyai dampak positif untuk kemajuan toko. Namun, Di era digital ini, tidak semua toko memanfaatkan teknologi khususnya terkait dengan absensi karyawan di toko Cahaya elektronik yang hanya bersifat konvensional serta menggunakan kertas. Dengan demikian, pemilik toko diharuskan untuk menyiapkan kertas absensi karyawan dan kertas untuk permohonan cuti. Konsekuensi dari proses absensi yang masih bersifat konvensional tersebut bisa membuat terjadinya penipuan yang tidak tak terduga, bahkan jika karyawan meminta cuti terkadang pemilik toko lupa menyimpan file formulir cuti.

Aplikasi absensi dirancang serta dibangun berdasarkan masalah ini, aplikasi kehadiran dirancang dengan menggunakan pemodelan *UML (Unified Modeling Language)* dengan metode *waterfall*. Sehingga menghasilkan Aplikasi absensi

yang efisien serta efektif yang dapat membantu pemilik toko untuk mendapatkan informasi yang akurat tentang kehadiran dan cuti. Dengan adanya aplikasi absensi ini juga mempengaruhi penghentian *COVID-19*, agar tidak terjadi kontak fisik yang dapat mengganggu kesehatan karyawan (Hutabri, 2022).

2.1.5 Literatur 5

Sistem pencatatan kehadiran yang saat ini berjalan dilakukan oleh karyawan melalui *group chat* dengan *update* tugas yang harus dilakukan. Karena karyawan *WFH* yang membuat laporan absensi lewat aplikasi *chat* dinilai kurang praktis. *HRD* wajib merekap secara manual bagi setiap karyawan untuk mengevaluasi prestasi kerja. Selain itu perusahaan mengalami kesulitan dalam mengetahui lokasi keberadaan karyawan, pengontrolan dan pengawasan terhadap keberadaan pegawai apakah telah memenuhi kewajibannya. Berdasarkan masalah tersebut, diperlukan suatu sistem aplikasi absensi yang dapat digunakan untuk membantu proses absensi bagi pegawai yang melakukan *Work From Home (WFH)*. *HRD* dapat memproses data absensi dengan mengetahui lokasi karyawan, atasan dapat memvalidasi absensi serta kinerja pegawai, serta sistem dapat mengolah data menjadi laporan yang dapat membantu perusahaan untuk mengetahui aktivitas dan kinerja karyawan. Dengan masalah Peneliti menganggap bahwa solusi untuk masalah ini adalah pembangunan sistem absensi untuk absensi karyawan.

Tujuan dari sistem yang dibangun adalah untuk membantu karyawan dalam melakukan absensi dengan *smartphone* yang merupakan alat absensi atau pengganti mesin absensi. Saat karyawan melakukan absensi di aplikasi, sistem dapat merekam

lokasi dan pelacakan lokasi *realtime* menggunakan layanan berbasis lokasi yang dapat mengubah *koordinat* menjadi sebuah alamat. Manfaat dari sistem ini adalah perusahaan mendapatkan laporan yang sesuai data yang diperlukan dan memberikan *visibilitas* kinerja karyawan yang melakukan *WFH* atau bekerja dari rumah agar perusahaan dapat mengukur kinerja karyawan.

2.2 Absensi

Menurut (Rubiati & Harahap, 2019) Absensi dapat dinyatakan sebagai pendataan kehadiran yang merupakan bagian dari pelaporan kegiatan yang ada di dalam suatu lembaga. Sedangkan menurut (Sugiarto dkk., 2019) Absensi disusun dan diatur sedemikian rupa agar mudah ditemukan dan digunakan saat dibutuhkan oleh pihak-pihak yang berkepentingan. Secara umum jenis-jenis absensi menurut metodenya penggunaannya dapat dikelompokkan menjadi dua, yaitu: 1 Absensi Manual, yaitu dengan cara menulis absensi dengan menggunakan pulpen berupa tanda tangan. 2 Kehadiran non-manual, yaitu dengan cara menggunakan alat yang terkomputerisasi, dapat mengenakan kartu RFID maupun fingerprint.

2.3 Location Based Service

Location Based Service (LBS) ialah istilah umum yang digunakan untuk menggambarkan teknologi yang digunakan untuk mencari lokasi perangkat yang kita gunakan. *LBS* merupakan layanan informasi yang bisa diakses menggunakan perangkat *mobile* dengan menggunakan *mobile network*, yang dilengkapi dengan kemampuan untuk memanfaatkan lokasi dari perangkat *mobile* tersebut. Ada dua elemen utama dalam *LBS*, yaitu:

- a. *Location Manager (API Maps)* memberikan *tools* atau alat untuk *LBS*, *Application Programming Interface (API Maps)* memberikan fasilitas untuk menampilkan, memanipulasi *maps* atau peta beserta fitur - fitur lainnya seperti tampilan satelit, *street* atau jalan, maupun keduanya. Paket ini terletak di *com.google.android.maps*.
- b. *Location Provider (API Location)* Memberikan teknologi pencarian lokasi yang dikenakan oleh perangkat. *API Location* menangani data *GPS (Global Positioning System)* dan data lokasi *real-time*. *API Location* ada di dalam paket *android* yaitu di dalam paket *android.location*. Selanjutnya Pengelola Lokasi, kita dapat memastikan lokasi kita saat ini, melacak perpindahan, juga kedekatan dengan lokasi tertentu dengan mendeteksi perpindahan.

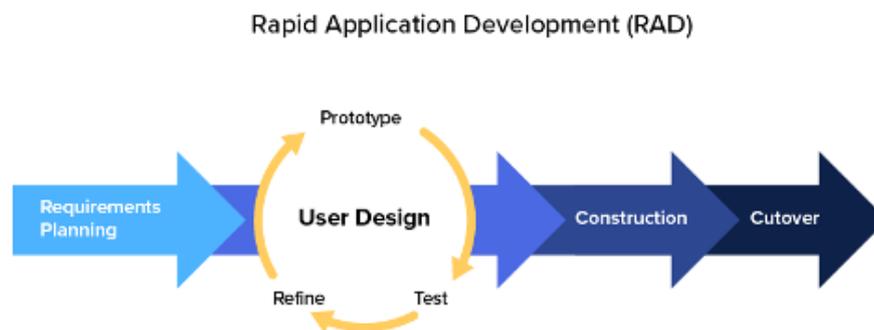
2.4 *Global Position System*

GPS merupakan singkatan dari *Global Positioning System* merupakan sistem untuk menentukan posisi serta navigasi secara global menggunakan satelit. Sistem yang pertama kali dikembangkan oleh Departemen Pertahanan Amerika Serikat ini digunakan untuk keperluan militer dan sipil (Susilo dkk., 2017). *GPS* ialah satu-satunya sistem satelit navigasi global untuk menentukan lokasi, kecepatan, arah, dan waktu yang beroperasi penuh di dunia saat ini. *GPS* mengenakan konstelasi 24 satelit yang mengorbit bumi, dimana penerima *GPS* menerima informasi dari tiga atau lebih satelit tersebut untuk menentukan posisi. Penerima *GPS* harus berada dalam *line-of sight (LoS)* dengan ketiga satelit untuk menentukan posisi, sehingga *GPS* hanya ideal untuk digunakan pada *outdoor positioning*.

2.5 Metode Pengembangan Sistem

2.5.1 Metode *Rapid Application Development*

Rapid Application Development (RAD) ialah golongan dari pengembangan model *System Development Life Cycle (SDLC)* model pengembangan perangkat lunak terhadap penekanan pada siklus pengembangan yang sangat singkat, yang bisa digunakan sebagai acuan untuk mengembangkan sistem informasi yang baik dari segi kecepatan dan ketepatan (Aswati dkk., 2017).



Gambar 2. 1 Tahapan Metode Rapid Application Development

Pada metode *Rapid Application Development* ini mempunyai empat tahapan pengembangan sistem, Keempat tahapan itu ialah:

1. Menentukan *Project Requirements*

Menentukan *project requirement* adalah proses pertama yang dikerjakan untuk menentukan kebutuhan apa yang dibutuhkan untuk membangun sebuah *project*, sesudah mendapatkan kebutuhan yang jelas, kemudian menentukan detailnya. Misalkan seperti tujuan, jadwal, dan anggaran yang diperlukan. Intinya,

tahap awal ini bermanfaat untuk memberikan gambaran tentang *project* yang akan dikerjakan nanti.

Pada tahap ini, pengumpulan informasi akan dilakukan dengan melakukan observasi dan wawancara dengan pihak SMA Gajah Mada Bandar Lampung untuk mengetahui apa saja kebutuhan serta bagaimana cara mengatasi kebutuhan dari *user* tersebut.

2. Membuat *prototype*

Tahap selanjutnya ialah membuat *prototype*, dimana *developer* secepatnya akan membuat *prototype* dari aplikasi yang diinginkan. Lengkap dengan berbagai fitur serta fungsi yang berbeda-beda. Tujuannya hanya untuk memberitahu apakah *prototype* yang dibangun sudah sesuai seperti keinginan serta kebutuhan *user*. Dan juga nantinya akan melibatkan *user* untuk *testing* dan memberikan *feedback*, proses ini memungkinkan untuk mempelajari *error* yang dapat timbul dikemudian hari, serta tahapan ini berguna untuk meminimalisir *error* dan *debugging*.

3. *Rapid Construction* dan Pengumpulan *Feedback*

Pada tahap ketiga ini dilakukan pengumpulan *feedback* yang diberikan oleh *user*. *Feedback* yang dimaksud disini meliputi fitur, fungsi, *visual*, serta *interface* dari program yang sedang dikembangkan. Lalu *prototipe* akan dikembangkan lagi sampai klien memberikan persetujuan untuk finalisasi produk. Dan seperti pembahasan sebelumnya, kedua tahapan ini akan diulang terus menerus, hingga hasilnya maksimal sesuai keinginan klien.

4. Implementasi dan finalisasi

Langkah terakhir ialah mengimplementasikan hasil *feedback* yang diberikan oleh *user* serta membuat *project* akhir. Dimana tugas utama *developer* ialah untuk menutupi kekurangan yang mungkin bisa terjadi pada saat proses pengembangan aplikasi nantinya, tugas ini penting karena mengoptimalkan *stabilitas* aplikasi, memperbaiki *interface*, sampai melakukan *maintenance* dan menyusun dokumentasi.

2.6 *Unified Modelling Language (UML)*

Menurut (Putra, 2018) *Unified Modeling Language (UML)* adalah bahasa visual untuk pemodelan serta komunikasi tentang sebuah sistem dengan menggunakan diagram serta teks pendukung.

1. *Activity Diagram*

Menurut (Nurman Hidayat & Kusuma Hati, 2021) *Activity Diagram* menggambarkan berbagai aliran aktivitas dalam sistem yang sedang dirancang, bagaimana setiap aliran dimulai, keputusan yang mungkin terjadi, serta bagaimana mereka berakhir.

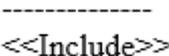
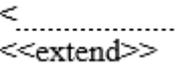
Tabel 2. 2 Activity Diagram

No	Simbol	Keterangan
1	Status Awal 	Status awal aktivitas sistem, diagram aktivitas memiliki status awal.
2	Aktivitas 	Aktivitas yang dilakukan oleh sistem, aktivitas biasanya dimulai dengan kata kerja.
3	Percabangan 	Percabangan asosiasi dimana jika ada lebih dari satu pilihan kegiatan.
4	Penggabungan 	Asosiasi penggabungan dimana lebih dari satu kegiatan digabungkan menjadi satu.
5	Status Akhir 	Status akhir yang dikerjakan sistem memiliki status akhir.

2. Use Case Diagram

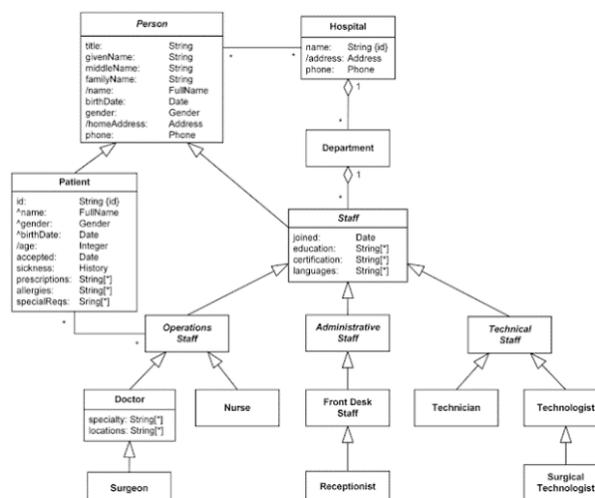
Menurut (Nurman Hidayat & Kusuma Hati, 2021) *Use case* diagram membuat fungsionalitas yang diharap dari suatu sistem. *Use Case* Diagram juga merupakan penggambaran grafis dari setengah atau semua *actor*, *use case*, dan interaksi diantara mereka yang memperkenalkan sebuah sistem. *Use case* diagram tidak menjelaskan secara rinci tentang *use case*, melainkan hanya menyampaikan gambaran singkat tentang hubungan antara *use case*, aktor, dan sistem.

Tabel 2. 3 Use Case Diagram

No	Simbol	Keterangan
1	Aktor/Actor 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibangun itu sendiri.
2		Asosiasi antara aktor dengan <i>use case</i> yang mengenakan panah terbuka agar menunjukkan saat aktor berinteraksi secara pasif dengan sistem.
3	 ----- <<Include>>	<i>Include</i> , ada di dalam <i>use case</i> lain (<i>required</i>) atau pemanggilan <i>use case</i> oleh <i>use case</i> lain, contohnya ialah pemanggilan fungsi.
4	 <----- <<extend>>	<i>Extend relationships</i> dimaksudkan agar menambahkan bagian ke <i>use case</i> yang ada serta untuk memodelkan sistem layanan opsional.

3. Class Diagram

Menurut (Nurman Hidayat & Kusuma Hati, 2021) *Class* diagram ialah salah satu diagram pertama dari *UML* untuk menggambarkan *class* atau *blueprint object* pada sebuah sistem. *Class* diagram sangat membantu dalam memvisualisasikan struktur kelas suatu sistem.



Gambar 2. 2 Contoh Class Diagram
(Sumber: <https://cutt.ly/0KuvDCq>)

2.7 Analisa *PIECES*

Menurut (Suharto, 2018) Untuk dapat mengidentifikasi suatu masalah, maka perlu untuk menganalisis kinerja, informasi, ekonomi, kontrol, efisiensi, serta layanan. Panduan ini dikenal sebagai analisis *PIECES* (*Performance, Information, Economy, Control, Efficiency, Service*). Analisis dilakukan pada sistem informasi berupa *hard copy* lama seperti brosur kapan band akan melaksanakan pentas. Dari analisis ini biasanya mendapatkan beberapa masalah dan pada akhirnya menemukan masalahnya utama. Berikut ini merupakan komponen-komponen dari analisis *PIECES*:

- a. Analisis Kinerja Sistem (*Performance*) ialah sebuah kemampuan sistem dalam menyelesaikan tugas cepat agar targetnya bisa segera tercapai. Kinerja diukur dengan jumlah produksi (*throughput*) serta waktu yang dipakai untuk menyesuaikan transfer pekerjaan (*response time*).
- b. Analisis Informasi (*information*) adalah hal penting karena dengan adanya informasi ini pihak manajemen (*marketing*) serta pengguna dapat melaksanakan langkah selanjutnya. Jika keterampilan sistem informasi baik, maka *user* akan mendapatkan informasi akurat, tepat waktu, serta relevan dengan apa yang diharapkan.
- c. Analisis Ekonomi (*Economy*) Pemanfaatan biaya telah digunakan dari penggunaan informasi. Peningkatan kebutuhan ekonomi berpengaruh terhadap kontrol biaya perusahaan serta peningkatan manfaat. Saat ini banyak perusahaan dan manajemen mulai menerapkan *paperless system* (minimalkan penggunaan kertas) untuk penghematan. Oleh karena itu dilihat dari

kegunaannya bahan serta biaya iklan di media cetak untuk media publikasi, sistem ini dinilai kurang ekonomis.

- d. Analisis Pengendalian (*Control*) Analisis ini digunakan untuk membandingkan sistem yang dianalisis berdasarkan akurasi waktu, kemudahan akses, serta akurasi data yang diproses.
- e. Analisis Efisiensi (*Efficiency*) Efisiensi berkaitan dengan bagaimana sumber digunakan secara optimal. Operasi pada perusahaan dikatakan efisien atau tidak biasanya didasarkan pada tugas serta tanggung jawab dalam melaksanakan kegiatan.
- f. Analisis Pelepasan (*Service*) Peningkatan layanan menunjukkan kategori yang berbeda. Proyek yang dipilih adalah peningkatan layanan yang lebih baik untuk manajemen (marketing), user serta bagian lain yang merupakan simbol kualitas dari sebuah sistem informasi.

2.8 *Android Studio*

Android studio ialah lingkungan pengembangan *Integrated Development Environment (IDE)* untuk pengembangan aplikasi *android*, berdasarkan *IntelliJ IDEA*. Selain sebagai *editor kode IntelliJ* serta alat Pengembang berdaya guna, *Android Studio* menawarkan lebih banyak fitur untuk meningkatkan produktivitas Anda saat membuat aplikasi *android*.

2.9 *Sublime Text 3*

Menurut (Pradana, 2021) *Sublime Text 3* adalah aplikasi *text editor* yang sangat berguna untuk menulis sejumlah *code* dan mampu membuka berbagai jenis

file. *Sublime Text* juga mendukung berbagai bahasa pemrograman seperti *HTML*, *C++*, *C*, *C#*, *CSS*, *Java*, *JS*, *ASP*, dan masih banyak lagi. *Sublime Text* merupakan *text editor* yang sering digunakan oleh para *programmer* khususnya *web developer* untuk menulis *coding*. *Sublime Text* telah menjadi sesuatu yang sangat penting bagi para *web developer* sebagai senjata *coding*. Aplikasi *Sublime Text* mudah digunakan karena warna tulisan *coding* yang sangat variatif dan interaktif tentunya sangat menarik dan juga bisa merubah warna tema.

2.10 Object Oriented Programming

Menurut (Perdana, 2021) *Object Oriented Programming (OOP)* ialah metode pemrograman berorientasi objek. Program yang ada merupakan kombinasi dari beberapa komponen kecil yang sudah ada sebelumnya. Hal ini dapat memudahkan pekerjaan seorang *programmer* dalam mengembangkan *program*. Objek-objek yang saling berhubungan dan tersusun menjadi satu kelompok disebut *class*. Nantinya, objek-objek tersebut akan saling berinteraksi untuk memecahkan masalah pemrograman yang kompleks atau rumit. Jika sebelumnya *developer* harus fokus pada *logic* yang akan dimanipulasi, dengan *OOP developer* bisa lebih fokus pada objek yang akan dimanipulasi. Pendekatan ini menawarkan cara mudah untuk menangani kompleksitas pemrograman. Tujuan utama *OOP* ialah untuk mengatasi kelemahan pendekatan pemrograman konvensional.

2.11 MySQL

MySQL adalah *Relational Database Management System (RDBMS)*, karena bisa mengatur data tentang bidang subjek yang berbeda ke dalam tabel. Pendekatan ini memudahkan untuk membawa data yang relevan saat diperlukan. Dengan

membentuk hubungan di antara tabel terpisah, bukannya menyimpan semua informasi dalam satu tabel besar, artinya dapat menghindari duplikasi banyak data, dapat menghemat ruang penyimpanan di komputer serta memaksimalkan kecepatan dan akurasi pekerjaan data. *MySQL* membantu mengelola *database* dengan menyediakan struktur yang efisien untuk menyimpan serta mengambil informasi (Sugiarto dkk., 2019).

2.12 Pengujian ISO 25010

Menurut (Salma, 2021) Pengujian ISO 25010 adalah bagian dari *Systems and Software Quality Requirements and Evaluation (SQuaRE)* yang merupakan versi lanjutan dari ISO 91261, yang telah direvisi secara teknis dengan menambahkan beberapa struktur serta bagian dari standar model kualitas. Tujuan penggunaan kualitas ialah untuk mengukur sejauh mana produk atau sistem dapat digunakan oleh pengguna untuk memenuhi kebutuhan dalam mencapai tujuan yang diinginkan dengan efisiensi, efektivitas, kepuasan dalam konteks penggunaan tertentu, serta bebas dari risiko. ISO 25010 Terdiri dari delapan karakteristik yaitu:

1. *Functional Suitability* adalah suatu sistem atau produk yang menyediakan fungsionalitas untuk memenuhi kebutuhan pada saat sistem atau produk tersebut dipakai dalam keadaan tertentu.
2. *Reliability* adalah sejauh mana suatu sistem atau produk dapat mempertahankan kinerjanya pada tingkat tertentu ketika digunakan dalam kondisi tertentu.

3. *Performance Efficiency* adalah tingkat dimana sistem atau produk memberikan kinerja yang baik dengan sejumlah sumber daya yang akan digunakan pada sistem atau produk.
4. *Usability* adalah sejauh mana pada sistem atau produk mudah dipahami, mudah digunakan, serta menarik untuk digunakan.
5. *Security* adalah sejauh mana sistem atau produk menyediakan layanan untuk melindungi akses, penggunaan, modifikasi, gangguan, atau pengungkapan.
6. *Compatibility* adalah kemampuan suatu komponen atau sistem untuk bertukar informasi.
7. *Maintainability* adalah tingkat dimana suatu sistem atau produk dapat dimodifikasi, yang meliputi perbaikan, pengembangan agar sesuai dengan lingkungan, modifikasi kriteria, serta spesifikasi fungsi.
8. Portabilitas adalah sejauh mana suatu sistem atau produk dapat dipindahkan darisatu ruang ke ruang lain.