

BAB II LANDASAN TEORI

2.1. Tinjauan Pustaka

Beberapa penelitian yang berkaitan dengan penerapan sistem informasi pemesanan lapangan berdasarkan jurnal penelitian terlihat pada Tabel 2.1:

Tabel 2.1 Tinjauan Pustaka

NO	Penulis (Tahun)	Judul	Masalah	Metode	Hasil
1.	(Anwar <i>et al.</i> , 2020)	Aplikasi Marketplace Penyewaan Lapangan Olahraga Dari Berbagai Cabang Dengan Metode Agile Development	Pengelolaan lapangan masih harus melakukan cara manual untuk operasional dan promosi lapangan. Di sisi penyewa lapangan olahraga juga masih harus mendatangi lapangan untuk mengetahui ketersediaan jadwal, fasilitas, dan harga yang ditawarkan. Kemudian	<i>Agile</i>	Hasil dari penelitian dapat disimpulkan bahwa menggunakan konsep marketplace dapat mempertemukan calon penyewa lapangan dengan banyak pengelola lapangan serta mudahnya mencari lapangan olahraga sesuai dengan keinginan penyewa dengan memilih kategori jenis lapangan, harga dan lokasi terdekat dan pengelola lapangan olahraga dapat mempromosikan lapangan olahraga dengan mendaftarkan lapangannya.

Tabel 2.1 Tinjauan Pustaka

2.	(Sidiarta, Ardyanti ² and Putra, 2018)	Rancang Bangun Sistem Informasi Marketplace Penyewaan Lapangan Futsal Berbasis Web	permasalahan yang dihadapi diantaranya pemesanan masih dilakukan secara manual	<i>Waterfall</i>	Penyedia dapat mengelola detail informasi yang disediakan, data lapangan, harga sewa, info turnamen, info nonton bareng, info kualitas lapangan dan info fasilitas lapangan. Penyewa dapat memilih jadwal lapangan yang ingin dipesan, melakukan pemesanan lapangan, konfirmasi pembayaran, dan cari lawan sparring.
3	(Nurullah, Arif and Vandini, 2021)	Perancangan Sistem Informasi Penyewaan Lapangan Bulutangkis Dan Futsal Pada Gor Bhanthong Jakarta	Dari survei yang dilakukan, Proses pengelolaan bisnis GOR Bhanthong yang dijalankan seperti melakukan penginputan data, melakukan pencarian data, dan pembuatan laporan masih dilakukan secara manual yaitu dengan tulis tangan sehingga belum efektif dan efisien.	<i>Waterfall</i>	Aplikasi ini dikelola oleh operator yaitu penjaga/pengurus lapangan yang bertugas untuk mencatat penyewaan lapangan, yang dimana operator dapat menggunakannya sebagai tempat penginputan data seperti input data booking, input data pembayaran lapangan, input data member, serta pembuatan laporan.

Tabel 2.1 Tinjauan Pustaka

4	(Merdekawati, Rahayu and Yulianti, 2019)	Sistem informasi penyewaan lapangan futsal berbasis web pada futsal station bekasi	Masih menggunakan cara manual.	<i>Waterfall</i>	Aplikasi ini dapat digunakan oleh 3 pengguna, yaitu admin, petugas dan penyewa. Admin dapat melakukan olah data lapangan, harga sewa, penyewa, booking, konfirmasi pembayaran dan halaman statis. Penyewa dapat melakukan booking via web, melihat informasi jadwal lapangan dan konfirmasi pembayaran
5	(Jamaris <i>et al.</i> , 2022)	Sistem Marketplace Pencarian Lapangan Futsal Menggunakan Metode Haversine Berbasis Android	Sistem informasi yang di terapkan dalam perusahaan masih cenderung memiliki banyak kekurangan dan kelemahan dengan menggunakan buku besar	<i>Waterfall</i>	Aplikasi marketplace pencarian lapangan futsal menggunakan metode Haversine ini dapat digunakan calon penyewa yang pada penelitian ini khususnya kota Pekanbaru dengan dua Kecamatan yaitu Tampan dan Marpoyan Dama

2.2. Aplikasi

Aplikasi adalah penggunaan atau penerapan suatu konsep yang menjadi suatu pokok pembahasan. Aplikasi dapat diartikan juga sebagai program komputer yang

dibuat untuk menolong manusia dalam melaksanakan tugas tertentu, Aplikasi software yang dirancang untuk suatu tugas khusus dapat dibedakan menjadi dua jenis (Tohari, 2017), yaitu :

- a. Aplikasi software spesialis, program dengan dokumentasi tergabung yang dirancang untuk menjalankan tugas tertentu.
- b. Aplikasi software paket, suatu program dengan dokumentasi tergabung yang dirancang untuk jenis masalah tertentu.

2.3. *E-marketplace*

E-marketplace adalah sebuah sistem informasi antar organisasi dimana pembeli dan penjual di pasar mengkomunikasikan informasi tentang harga, produk dan mampu menyelesaikan transaksi melalui saluran komunikasi elektronik. Suatu *e-marketplace* merepresentasikan suatu struktur sosial, konsep ekonomi pasar, dan penggunaan teknologi. *E-marketplace* dapat memberikan peluang untuk melakukan bisnis dan melaksanakan transaksi melalui saluran elektronik, biasanya pada *platform* yang berbasis *internet* (Hans, Andjarwirawan and Rostianingsih, 2020)

E-marketplace adalah wadah komunitas bisnis interaktif secara elektronik yang menyediakan pasar dimana perusahaan dapat ambil andil dalam B2B *e-commerce* dan atau kegiatan *e-business* lain. Ada dua jenis *e-marketplace* :

1. *E-marketplace horizontal*

E-marketplace horizontal dikategorikan berdasarkan fungsi atau produk umum yang ditawarkan perusahaan. Dapat diartikan pasar yang digunakan

untuk industri umum. Seperti pasar penjualan smartphone, pc, baju. Biaya transaksi yang dikeluarkan lebih rendah.

2. *E-marketplace vertical*

E-marketplace vertikal dapat diartikan pasar yang digunakan untuk industri yang memenuhi kebutuhan khusus pada masing-masing industri. Seperti pasar penjualan beton dan baja. Dalam pengaplikasian *e-marketplace* dibutuhkan strategi untuk mengoptimalkan *e-marketplace*.

2.4. Pemesanan

Pemesanan yaitu pemesanan fasilitas yang diantaranya akomodasi, meal, seat pada pertunjukan, pesawat terbang, kereta api, bus, hiburan, *night club*, *discoutegue* dan sebagainya (Suartana, 2017). Pemesanan adalah proses, perbuatan, cara memesan suatu tempat, barang, dan lain-lain (McLoad, 2017).

Berdasarkan pengertian para ahli diatas dapat disimpulkan bahwa pemesanan adalah suatu aktifitas yang dilakukan oleh konsumen sebelum membeli, menyewa atau melakukan *service* untuk mewujudkan kepuasan konsumen maka perusahaan harus mempunyai sebuah sistem *booking* yang baik.

Pemesanan dibagi menjadi dua jenis yaitu pemesanan *online* dan pemesanan *offline*:

1) Pemesanan *Online*.

Kemajuan teknologi saat ini mengakibatkan sistem pemesanan juga mengalami perkembangan kearah sistem pemesanan *online*. Pemesanan *online* bisa diakses oleh siapapun dan dimanapun mereka berada yang memiliki akses *internet*.

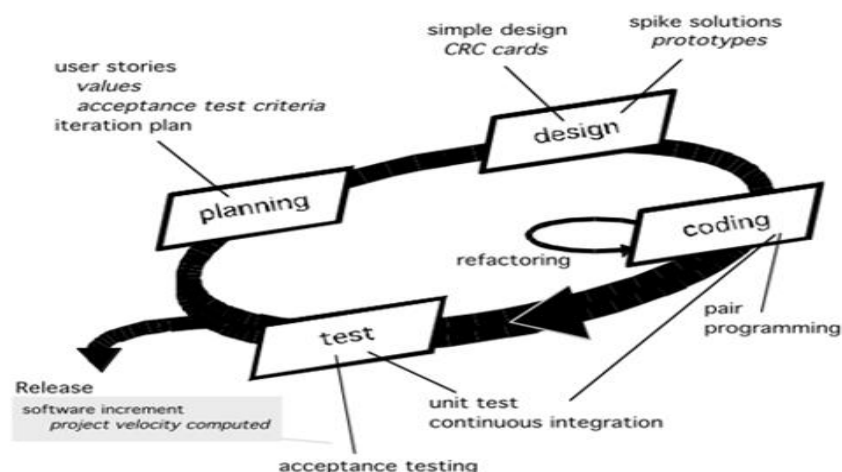
2) Pemesanan *Offline*

Sistem pemesanan yang menggunakan pengiriman *booking* langsung ke tempat dengan media pemesanan seperti *telepon, fax, e-mail, dan walk in*.

2.5. Metode Pengembangan Sistem

Extreme Programming (XP) merupakan sebuah proses rekayasa perangkat lunak yang cenderung menggunakan pendekatan berorientasi objek dan sasaran dari metode ini adalah tim yang dibentuk dalam skala kecil sampai medium serta metode ini juga sesuai jika tim dihadapkan dengan *requirement* yang tidak jelas maupun terjadi perubahan-perubahan *requirement* yang sangat cepat (Supriyatna,2018).

Extreme Programming (XP) dikenal dengan metode atau “*technical how to*” bagaimana suatu tim teknis mengembangkan perangkat lunak secara efisien melalui berbagai prinsip dan teknik praktis pengembangan perangkat lunak. XP menjadi dasar bagaimana tim bekerja sehari-hari (Lubis, 2016). Tahapan *Extreme Programming* dapat dilihat pada **Gambar 2.1**



Gambar 2. 1 Ilustrasi Proses *Extreme Prgamming*

Sumber: Lubis (2016)

1. *Planning* (Perencanaan)

Kegiatan Perencanaan dimulai dengan mendengarkan suatu kegiatan yang bertujuan mengumpulkan kebutuhan-kebutuhan untuk memahami konteks bisnis dan *fungsi*alitas.

2. *Design* (Perancangan)

Perancangan yang simple, menarik, dan sederhana selalu memberikan hasil yang lebih disukai daripada gambaran-gambaran yang lebih kompleks. Perancangan XP memberikan panduan implementasi untuk suatu cerita ketika ditulis, tidak kurang, tidak lebih.

3. *Coding* (Pengkodean)

Pengkodean ini dilanjutkan setelah cerita yang telah dikembangkan dan rancangan yang telah dilakukan selesai.

4. *Pengujian* (Pengujian)

Unit pengujian yang harus dibuat dan kemudian dijalankan menggunakan kerangka kerja yang memungkinkan sehingga dapat dijalankan dengan mudah dan dapat dijalankan berulang kali.

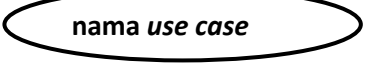



2.6. Bahasa Pemodelan Pengembangan Sistem (UML)

Bahasa Pemodelan Pengembangan Sistem (*Unified Modeling Language*) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement* (kebutuhan), membuat analisis & desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek (Rosa & Shalahuddin, 2018).



2.6.1. Use Case Diagram

Use case diagram atau diagram *use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat (Rosa & Shalahuddin, 2018). *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat menjelaskan simbol-simbol yang ada pada diagram *use case* dapat dilihat pada gambar 2.2 di bawah ini:

Tabel 2.2 Simbol Diagram Use Case

Simbol	Deskripsi
<p><i>Use Case</i></p> 	<p>Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>use case</i></p>
<p>Aktor/<i>actor</i></p> 	<p>Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda di awal frase nama <i>actor</i></p>
<p>Asosiasi/<i>association</i></p> 	<p>Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan <i>actor</i></p>
<p>Ekstensi/<i>extend</i></p> <p><<<i>extend</i>>></p> 	<p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu mirip dengan prinsip <i>inheritance</i> pada pemrograman</p>

Tabel 2.2 Simbol Diagram *Use Case* (Lanjutan)

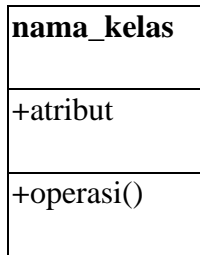
Simbol	Deskripsi
	berorientasi objek biasanya <i>use case</i> tambahan memiliki nama depan
<p>Generalisasi/<i>generalization</i></p> 	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
<p>Menggunakan/<i>Include/uses</i></p> <p><<<i>include</i>>></p> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini

Sumber : (Rosa & Shalahuddin, 2018)




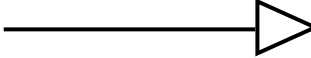

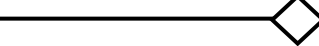
2.6.2. Class Diagram

Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi (Rosa and Shalahudin, 2018). menjelaskan simbol-simbol yang ada pada diagram kelas pada tabel *class diagram* 2.3.

Tabel 2.3 Simbol *Class Diagram*

Simbol	Deskripsi
<p>Kelas</p> 	Kelas pada struktur sistem

Tabel 2. 3 Simbol *Class Diagram* (Lanjutan)

Simbol	Deskripsi
Antarmuka/ <i>Interface</i>  nama_interface	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek
Asosiasi/ <i>asociation</i> 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
Asosiasi berarah/ <i>directed association</i> 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya disertai dengan <i>multiplicity</i>
Generalisasi 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus)
Kebergantungan/ <i>dependecy</i> 	Relasi antar kelas dengan makna kebergantungan antar kelas
Agregasi/ <i>agregation</i> 	Relasi antar kelas dengan makna semua bagian (<i>whole-part</i>)





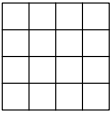


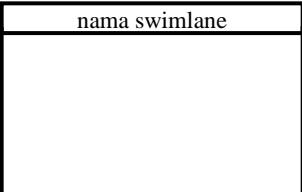
Sumber : (Rosa & Shalahuddin, 2018)

2.6.3. Activity Diagram

Activity diagram atau Diagram aktivitas menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis (Rosa & Shalahuddin,

2018), menjelaskan Simbol-simbol yang ada pada *activity diagram* dapat dilihat pada tabel 2.4 di bawah ini :

Tabel 2.4 Simbol *Activity Diagram*

Simbol	Deskripsi
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/ <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
Penggabungan/ <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu
Tabel 	Suatu file komputer dari mana data bisa dibaca atau direkam selama kejadian bisnis
Dokumen 	Menunjukkan dokumen sumber atau laporan
Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
<i>Swimlane</i> 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

Sumber : (Rosa & Shalahuddin, 2018)

2.7. Analisis PIECES

Analisis pieces juga sangat banyak diterapkan untuk penelitian pada suatu perusahaan atau organisasi. Selain mudah dan dapat dipahami analisis pieces juga bersifat ringan tidak membutuhkan data yang banyak. Berikut ini kriteria yang wajib ada pada analisis pieces menurut (Sanusi, 2012) :

1. Analisis Kinerja (*Performance*), kinerja sistem saat ini masih jauh dari yang diharapkan. Pekerja Toko harus melakukan pengecekan data secara berulang agar tidak terjadi kesalahan dalam pengolahan data, hal ini menunjukkan kinerja yang belum optimal dan membutuhkan waktu yang cukup lama dalam pengolahan data penjualan. Akhirnya bisa memperburuk kinerja karyawan. Jika kinerja karyawan menurun, maka performance perusahaan pun akan menurun.
2. Analisis Informasi (*Information*), sistem yang berjalan belum dapat memberikan informasi yang akurat yang terbebas dari kesalahan, dan informasi yang dihasilkan sering terjadi kesalahan. Selain itu pelanggan mengalami kesulitan untuk memperoleh informasi mengenai produk baru. Hal ini juga terjadi karena karyawan lebih sedikit dan pelanggan lebih banyak sehingga *input* dan *output* nya jarang diperhatikan sehingga bisa merusak informasi yang tepat. Pelanggan dari kalangan bisnis juga banyak dan belum ter-handle secara sistem.
3. Analisis Ekonomi (*Economy*), proses pengelolaan yang dilakukan saat ini masih dilakukan dengan cara pencatatan dokumen. Dalam pembuatan laporan masih banyak pengolahan data yang berulang dan mengakibatkan pemborosan biaya.

4. Analisis Kendali (*Control*), pengelolaan sistem yang berjalan sudah dirasa cukup baik dalam pengendalian, hal ini ditunjukkan dengan adanya dokumen yang dibuat lengkap dalam prosesnya.
5. Analisis Efisiensi (*Efficiency*), penggunaan waktu yang belum maksimal dikarenakan sering terjadinya pengecekan data secara berulang agar tidak terjadi kesalahan dalam pengolahan datanya, sehingga mempengaruhi proses pembuatan laporan.
6. Analisis Layanan (*Services*) dalam hal pelayanan masih dirasa belum baik, perlu ditingkatkan dengan cara meningkatkan keakuratan serta kecepatan dalam proses pelayanan sehingga sangatlah diperlukan sistem yang berbasis database dan real-time.

2.8. Aplikasi Implementasi

2.8.1. Xampp

XAMPP adalah suatu program yang digunakan sebagai server untuk mengeksekusi fungsi yang ada dalam halaman website (Supono and Putratama, 2016)

XAMPP merupakan software yang digunakan sebagai server untuk mengeksekusi fungsi website tanpa jaringan internet. XAMPP terdiri dari apache web server, MySQL, PHP, Perl, FTP server dan phpMyAdmin (Palit, Rindengan and Lumenta, 2019). di dalam folder utama xampp, terdapat beberapa folder penting yang perlu diketahui. Untuk lebih memahami setiap fungsinya, Anda dapat melihat penjelasannya pada tabel 2.5 dibawah ini:

Tabel 2.5 Folder Penting Xampp

Folder	Keterangan
<i>Apache Folder</i>	Folder utama dari Apache Webserver Htdocs
<i>Htdocs</i>	Folder utama untuk menyimpan data-data latihan web, baik PHP maupun HTML biasa.
<i>Manual</i>	Berisi subfolder yang di dalamnya terdapat manual program dan database, termasuk manual PHP dan MySQL.
<i>MySQL</i>	Folder utama untuk database MySQL Server.
<i>PHP</i>	Folder utama untuk program PHP.

Sumber : (Supono and Putratama, 2016)

2.8.2. Visual Code Studio

Visual Studio Code adalah aplikasi code editor buatan Microsoft yang dapat dijalankan di semua perangkat desktop secara gratis. Kelengkapan fitur dan ekstensi membuat code editor ini menjadi pilihan utama para pengembang. Visual Studio Code bahkan mendukung hampir semua sistem operasi seperti Windows, Mac OS, Linux, dan lain sebagainya (Destiningrum and Adrian, 2017).

Visual Studio Code adalah aplikasi *editor* teks gratis di kembangkan oleh *Microsoft* yang dapat digunakan di semua bahasa pemrograman yang ada tanpa perlu berganti aplikasi *editor*, serta dapat dijalankan di berbagai *platform Operating System (OS)* seperti *windows*, *linux*, dan *mac OS*. *Visual Studio Code* memudahkan para *Programmer* saat berganti bahasa pemrograman tanpa perlu berganti aplikasi *editor* serta memahami dan konfigurasi *tools* kembali di aplikasi *editor* barunya. *Visual Studio Code* juga memberikan kebebasan kepada penggunaanya dalam tema, *debugger*, *extension*, dan lainnya (Rudianto, 2018).

2.8.1 MySQL

MySQL adalah *software* atau program Database Server. Sedangkan SQL adalah bahasa pemrogramannya, bahasa permintaan (*query*) dalam database server termasuk dalam MySQL itu sendiri. SQL juga dipakai dalam *software database* server lain, seperti SQL Server, Oracle, PostgreSQL dan lainnya (Nugroho, 2015).

MySQL adalah database server open source yang cukup populer keberadaannya. Dengan berbagai keunggulan yang dimiliki, membuat software database ini banyak digunakan oleh praktisi untuk membangun suatu project. Adanya fasilitas API (*Application Programming Interface*) yang dimiliki oleh Mysql, memungkinkan bermacam-macam aplikasi Komputer yang ditulis dengan berbagai bahasa pemrograman dapat mengakses basis data MySQL (Palit, Rindengan and Lumenta, 2019). Adapun perintah yang terdapat pada MySQL adalah sebagai berikut:

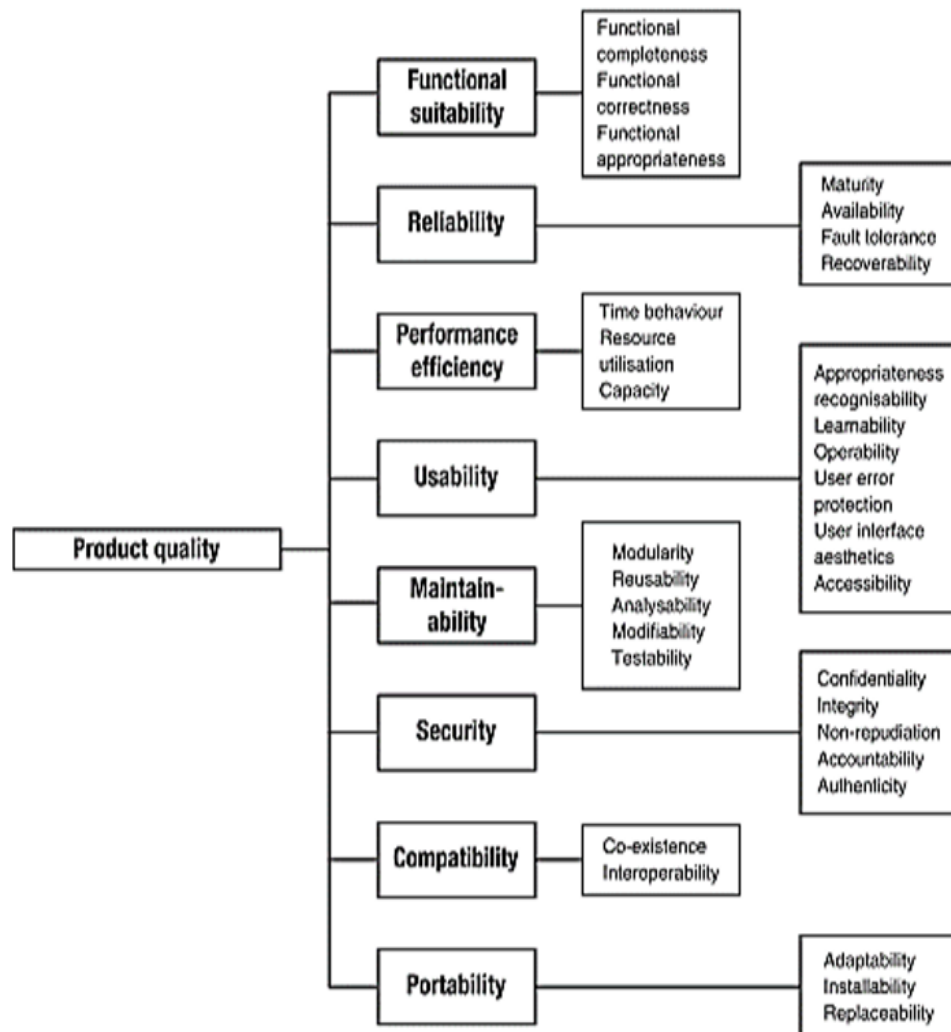
Tabel 2.6 Perintah Dasar MySQL

Perintah	Keterangan
<i>Show databases</i>	Perintah ini digunakan untuk menampilkan atau melihat daftar database yang sudah ada (sudah dibuat)
<i>Use</i>	Perintah ini digunakan untuk masuk atau mengakses database yang sudah ada
<i>Show tables</i>	Perintah ini digunakan untuk melihat atau menampilkan semua tabel yang ada di dalam database aktif (yang sudah dibuka, sudah di use)
<i>Desc/describe</i>	Perintah ini digunakan untuk melihat struktur tabel
<i>Quit</i>	Perintah ini digunakan untuk keluar dari MySQL Server.

Sumber : (Palit, Rindengan and Lumenta, 2019)

2.9. Pengujian Sistem ISO 25010

ISO/IEC 25010 merupakan standar yang digunakan oleh dunia internasional untuk melakukan evaluasi atau penguku-ran kualitas dari perangkat lunak (Wattiheluw, Rochimah and Faticah, 2019). Secara keseluruhan ISO/IEC 25010 memiliki 8 karakteristik untuk mengukur kualitas perangkat lunak secara menyeluruh seperti ditampilkan pada Gambar 2.3.



Gambar 2. 2 Model kualitas produk ISO/IEC 25010

Sumber : (Wattiheluw, Rochimah and Faticah, 2019)

Adapun beberapa definisi karakteristik ISO/IEC 25010 adalah sebagai berikut:

1. *Functional suitability* adalah produk aplikasi yang memberikan fungsional untuk memenuhi kebutuhan saat menggunakan produk dalam keadaan tertentu.
2. *Reliability* adalah tingkat dimana produk aplikasi dapat mempertahankan kinerja pada level tertentu ketika digunakan dalam keadaan tertentu.
3. *Performance efficiency* adalah tingkat dimana produk aplikasi menyediakan performa yang baik dengan jumlah *resource* yang digunakan.
4. *Usability* adalah dimana produk aplikasi mudah dimengerti, dipakai dan menarik untuk digunakan.
5. *Security* adalah tingkat produk aplikasi menyediakan layanan untuk melindungi akses, penggunaan, modifikasi, pengrusakan, atau pengungkapan yang berbahaya.
6. *Compatibility* adalah kemampuan dari suatu komponen aplikasi atau lebih untuk bertukar informasi.
7. *Maintainability* adalah tingkat dimana produk aplikasi dapat dimodifikasi. Modifikasi yang dilakukan dapat meliputi perbaikan, pengembangan atau adaptasi perangkat lunak untuk menyesuaikan dengan lingkungan, serta modifikasi pada kriteria dan spesifikasi fungsi.
8. *Portability* adalah tingkat dimana produk aplikasi dapat dipindahkan dari satu ruang ke ruang lain.