

## BAB II LANDASAN TEORI

### 2.1. Tinjauan Pustaka

Tinjauan pustaka dari penelitian sebelumnya digunakan dalam mendukung penelitian yang sedang dilakukan. Daftar literature penelitian sebelumnya terkait dengan penelitian yang akan dilakukan, disajikan dalam tabel 2.1:

**Tabel 2. 1** Tinjauan Pustaka

<b>No. 1</b>	Ridwan et al. (2019)
<b>Judul</b>	Perancangan Aplikasi Sewa Lapangan Berbasis Web Pada Aziz Futsal Kota Ternate
<b>Jurnal</b>	<i>IJIS - Indonesian Journal On Information System</i>
<b>Volume dan Halaman</b>	Vol. 4 No. 1
<b>Tahun</b>	2019
<b>Penulis</b>	Ridwan, Djunaidi Safi dan Hairil K. Siradjudin
<b>Identifikasi Masalah</b>	Dalam melakukan sewa lapangan dan pengolahan data sewa lapangan belum menggunakan aplikasi dimana proses pengolahan data sewa lapangan yang dilakukan dengan cara menulis pada buku registrasi penyewaan lapangan seperti nama penyewa, kode lapangan, durasi waktu sewa, jam mulai dan jam selesai, sehingga memperlambat kasir atau admin proses mengelolah data sewa lapangan.

Tabel 2.1 Tinjauan Pustaka (Lanjutan)

<b>Metode/Tools</b>	<p>1. Metode Analisis Sistem menggunakan Model <i>Driven Analysis</i> berorientasi objek dengan alat bantu UML dan Pengembangan Sistem menggunakan model <i>Waterfall</i> serta alat uji sistem menggunakan model <i>blackbox</i>.</p> <p>2. CSS, PHP, HTML digunakan sebagai bahasa pemrograman yang dipakai dalam <i>website</i>.</p>
<b>Hasil Penelitian</b>	<p>Aplikasi Sewa Lapangan pada Aziz Futsal kota Ternate berbasis web, Sistem ini akan maksimal ketika diakses secara <i>online</i>, Sistem Pengolahan data sewa berbasis web ini dibuat untuk mempermudah admin dan pelanggan dalam proses sewa lapangan pada sistem.</p>
<b>No. 2</b>	Setiawan dan Noris (2021)
<b>Judul</b>	Sistem Informasi Reservasi Penyewaan Penggunaan Gedung Lapangan Bulutangkis Berbasis Web Dengan Metode Waterfall
<b>Jurnal</b>	Prosiding Seminar Nasional Informatika dan Sistem Informasi
<b>Volume dan Halaman</b>	Vol. 5 No. 3
<b>Tahun</b>	2021
<b>Penulis</b>	Bagas Setiawan dan Shandi Noris

**Tabel 2.1** Tinjauan Pustaka (Lanjutan)

<p><b>Identifikasi Masalah</b></p>	<p>Salah satu kelemahan beberapa penyedia lapangan bulu tangkis adalah tidak mengikuti perkembangan teknologi. Ketika seorang calon pelanggan ingin memesan di gedung bulutangkis, ternyata gedung bulu tangkis tersebut sudah sepenuhnya disewa oleh pelanggan lain, sehingga calon pelanggan harus mencari gedung lain di lapangan bulutangkis. Tidak hanya calon pelanggan, kelemahan ini juga dirasakan oleh penyedia jasa pembangunan lapangan bulu tangkis, misalnya belum maksimal dalam melakukan promosi, karena masih menggunakan media promosi konvensional yaitu hanya menggunakan plang nama gedung. Selain itu, proses penyewaan gedung lapangan bulutangkis masih belum maksimal, dimana sering terjadi konflik waktu antar pelanggan akibat ketidakakuratan informasi yang diperoleh antar pegawai lapangan bulutangkis.</p>
<p><b>Metode/Tools</b></p>	<ol style="list-style-type: none"> <li>1. Model pengembangan menggunakan model <i>waterfall</i> menggunakan aplikasi UML.</li> <li>2. Menggunakan Bahasa pemrograman PHP seta melakukan pengujiannya melalui <i>blackbox</i>.</li> </ol>

**Tabel 2.1** Tinjauan Pustaka (Lanjutan)

<b>Hasil Penelitian</b>	Pembuatan aplikasi ini hanya berfokus ke reservasi penyewaan saja. Sistem informasi reservasi penyewaan gedung lapangan bulutangkis ini dibuat menggunakan bahasa pemrograman PHP, HTML, CSS, <i>Javascript</i> dan menggunakan MySQL sebagai <i>database</i> .
<b>No. 3</b>	Purwanto et al. (2021)
<b>Judul</b>	Sistem Informasi Penyewaan Lapangan Futsal Berbasis Web
<b>Jurnal</b>	Jurnal Informatika dan Rekayasa Perangkat Lunak
<b>Volume dan Halaman</b>	Vol. 3 No. 2
<b>Tahun</b>	2021
<b>Penulis</b>	Heri Purwanto, Fiqri Arya Nugraha, Mochamad Raffnie Prayogha dan Rafi Martua Syahputra
<b>Identifikasi Masalah</b>	Aplikasi reservasi lapangan futsal membantu untuk memesan bukan dengan catatan tetapi dengan aplikasi dan mudah digunakan, aplikasi pemesanan lapangan futsal dapat mengidentifikasi jadwal pemesanan yang dilakukan oleh pengguna dan pemesanan lapangan sesuai dengan waktu bermain.

**Tabel 2.1** Tinjauan Pustaka (Lanjutan)

<b>Metode/Tools</b>	<ol style="list-style-type: none"> <li>1. Metode pengembangan sistem yang penulis gunakan dalam perancangan sistem adalah <i>Waterfall</i> serta analisis perancangan menggunakan UML.</li> <li>2. Menggunakan Bahasa pemrograman PHP dan <i>database</i> menggunakan MySQL.</li> </ol>
<b>Hasil Penelitian</b>	Perangkat lunak sistem informasi ini bertujuan untuk membantu mengatasi permasalahan lapangan futsal secara manual. Diharapkan juga menggunakan teknologi saat ini untuk menjangkau banyak orang yang masih menyewa lapangan futsal agar dapat digunakan. Menghasilkan aplikasi berbasis web yang membantu penyewa mudah dalam melakukan penyewaan lapangan tanpa harus datang secara langsung.
<b>No. 4</b>	Iriadi et al. (2019)
<b>Judul</b>	Sistem Informasi Penyewaan Lapangan Futsal Berbasis Web Pada Futsal Station Bekasi
<b>Jurnal</b>	<i>Indonesian Journal on Networking and Security (IJNS)</i>
<b>Volume dan Halaman</b>	Vol. 8 No. 4
<b>Tahun</b>	2019

**Tabel 2.1** Tinjauan Pustaka (Lanjutan)

<b>Penulis</b>	Nandang Iriadi, Priatno, Ahmad Ishaq dan Winda Yulianti
<b>Identifikasi Masalah</b>	Futsal Station ini dalam proses pelayanan penyewaan lapangan futsal harus datang ketempat futsalnya untuk mem- <i>booking</i> lapangan futsal, pada penginputan data-data penyewa masih menggunakan kertas catatan. Bukti pembayaran saat ini masih menggunakan banyak kertas untuk membuat laporan pembayaran, sehingga jika ingin merekap data, pengelola kesulitan karena ada banyak kertas yang harus dikumpulkan.
<b>Metode/Tools</b>	<ol style="list-style-type: none"> <li>1. Dalam perancangan sistem infomasi penyewaan lapangan futsal berbasis web ini, penulis menggunakan salah satu model pengembangan sistem yaitu model SDLC atau <i>waterfall</i>.</li> <li>2. Bahasa pemrograman menggunakan PHP dan <i>database</i> yang dipakai MySQL dan di desain dengan <i>Entity Relationship Diagram</i> (ERD) dan <i>Logical Relational Structure</i> (LRS).</li> </ol>
<b>Hasil Penelitian</b>	<ol style="list-style-type: none"> <li>1. Dengan dibuatnya sistem berbasis web ini penyewa dapat mem-<i>booking</i> lapangan futsal secara <i>online</i>, dimana penyewa tidak perlu datang langsung ke futsal station untuk melakukan penyewaan.</li> <li>2. Mempermudah dalam hal penyajian informasi yang akurat.</li> </ol>

**Tabel 2.1** Tinjauan Pustaka (Lanjutan)

<b>No. 5</b>	Togu et al. (2021)
<b>Judul</b>	Sistem Informasi Penyewaan Lapangan Bulutangkis Berbasis Web Pada GOR Villa Mas Indah Bekasi Utara
<b>Jurnal</b>	<i>Journal of Students' Research in Computer Science</i>
<b>Volume dan Halaman</b>	Vol. 2 No. 1
<b>Tahun</b>	2021
<b>Penulis</b>	Tumbur Togu, Herlawati dan Adi Muhajirin
<b>Identifikasi Masalah</b>	<p>GOR Bulu Tangks Villa Mas Indah memiliki 4 boks untuk disewakan setiap hari yang buka dari pukul 10:00 hingga 23:00 dan proses penyewaan dilakukan secara manual, konsumen terlebih dahulu harus mendatangi GOR Bulutangkis Villa Mas Indah dan menanyakan jadwal yang tersedia, maka pemilik GOR melakukan pengecekan jadwal, jika tersedia pelanggan menyewa lapangan dan membayar tunai dengan harga yang telah ditetapkan oleh pemilik GOR, kemudian pelanggan menerima kwitansi. Pada Lapangan 4 dapat di hitung total 154 jam yang sudah disewa atau <i>Booking</i> dan terdapat 176 jam yang kosong atau tidak disewa dalam bulan November dan dapat di hitung total 203 jam yang sudah di sewa atau <i>Booking</i> dan</p>

Tabel 2.1 Tinjauan Pustaka (Lanjutan)

	<p>terdapat 127 jam yang kosong atau tidak disewa dalam bulan Desember. Berdasarkan hasil survei, peneliti memperoleh poin-poin penting tentang hasil sewa. Penyewaan lapangannya dan banyak konsumen telah mengalami perbedaan dalam rincian pembayaran selama pesanan secara tunai, sehingga pemilik GOR harus bertanggung jawab atas kesalahan tersebut.</p>
<b>Metode/Tools</b>	<ol style="list-style-type: none"> <li>1. Menggunakan metode <i>waterfall</i>.</li> <li>2. Menggunakan Bahasa pemodelan UML, perancangan menggunakan berbasis <i>java</i> dan MySQL dan pengujiannya menggunakan <i>blackbox</i>.</li> </ol>
<b>Hasil Penelitian</b>	<p>Dengan adanya sistem informasi penyewaan bulutangkis ini yang memudahkan pelanggan untuk menyewa lapangan sesuai waktu dan tanggal yang telah ditentukan, dengan adanya <i>website</i> penyewaan lapangan ini diharapkan dapat membantu pelanggan memberitahukan jadwal yang tersedia beserta informasi tentang bulutangkis serta mempercepat sistem pembayaran yang dilakukan pelanggan.</p>

## **2.2. Rancang Bangun**

Menurut Hidayat dan Piliang (2019) Rancang bangun adalah proses menerjemahkan suatu yang dikerjakan serta di analisa menggunakan berbagai metode yang beragam dan dapat dijelaskan melalui arsitektur serta berbagai komponen yang sangat sedikit material dan mengalami berbagai proses pengerjaannya.

## **2.3. Sistem Informasi**

Pengertian menurut Alzedan (2019) Sistem informasi dibagi menjadi dua kata, yaitu sistem dan informasi. Sistem berarti gabungan subsistem yang bertujuan untuk mencapai suatu tujaun. Informasi merupakan sesuatu yang mudah dimengerti oleh penerima. Sistem informasi berarti sistem yang mengeluarkan informasi. Sistem informasi telah ada pada zaman dahulu.

## **2.4. Penyewaan**

Menurut Fadilah (2018) Penyewaan merupakan proses, cara, pembuatan menyewa atau menyewakan. Yang dimaksud menggunakan sewa, yaitu balas jasa atas sewa ruangan pada keadaan kosong yang dapat ditagih dimuka (di awal penyewaan) atau dibelakang, sesuai menggunakan kontrak (perjanjian).

## **2.5. Website**

Menurut Hidayat dan Piliang (2019) *Website* merupakan beberapa halaman yang saling berinteraksi satu sama lain berupa file-file. Yang didalamnya

terdapat halaman homepage berada pada posisi atas, dengan halaman yang terkait dibagian bawahnya, terdapat *homepage child page* serta berisi *hyperlink* ke halaman lain dalam web.

## 2.6. Metode *PIECES*

Menurut Suharto (2018) Suatu masalah yang dapat diterjemahkan, maka dapat digambarkan sebagai kinerja, informasi, ekonomi, pengendalian, efisiensi dan pelayanan. Aturan ini biasanya disebut dengan analisis *PIECES* (*Performance, Information, Economic, Control, Efficiency, Service*). Dari hasil analisis ini akan diperoleh berbagai usulan untuk membantu sistem yang lebih baik. Hasil *PIECES* dapat dilihat pada tabel berikut:

**Tabel 2. 2 Analisis *PIECES***

ANALISIS	SISTEM LAMA	SISTEM USULAN
Kinerja	Proses pengolahan data laporan <i>booking</i> - lapangan futsal dan bulutangkis masih mencatat dikertas, sehingga ketika rekap akhir bulan terjadi kesalahan dalam penghitungan jumlah uang rekap akhir bulan.	Proses pengolahan data dapat memanfaatkan sistem (otomatis), dan informasi laporan rekap dapat secara langsung di lihat secara <i>website</i> tanpa harus mencatat ulang di kertas.

ANALISIS	SISTEM LAMA	SISTEM USULAN
Informasi	<p>Informasi jadwal penyewaan masih melihat buku serta disampaikan secara langsung atau bisa menggunakan aplikasi <i>Whatsapp</i>.</p>	<p>Informasi jadwal penyewaan untuk mengetahui lapangan yang masih kosong menggunakan <i>website</i> dan dapat dilihat diberanda awal <i>website</i> tanpa harus datang ke lapangan langsung, karena semua informasi akan di <i>update</i> di <i>website</i>.</p>
Ekonomi	<p>Pengolahan data laporan rekap penyewaan lapangan futsal dan bulutangkis masih berjalan kurang baik dikarenakan, setiap hari mencatat dikertas karena banyaknya penyewa setiap harinya. Serta menghabiskan kertas banyak dan</p>	<p>Pengolahan data laporan rekap penyewaan lapangan futsal dan bulutangkis akan di-<i>input</i> langsung secara <i>online</i> melalui <i>website</i> tanpa harus menggunakan kertas, jadi dapat menghemat biaya dalam</p>

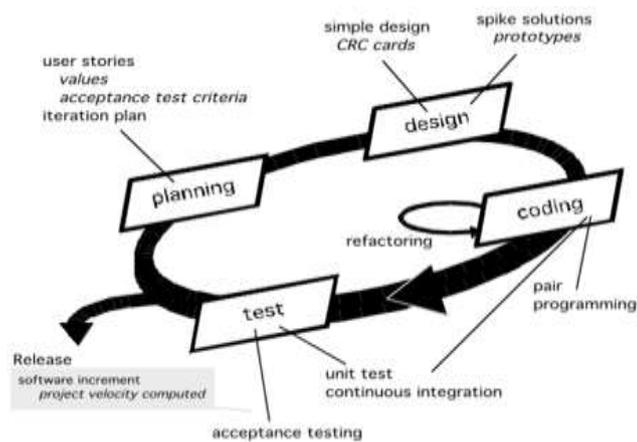
ANALISIS	SISTEM LAMA	SISTEM USULAN
	memerlukan biaya untuk mengeluarkan kertas setiap bulaannya untuk mencatat para penyewa.	menggunakan kertas setiap bulannya.
Kontrol	Belum adanya batasan data sehingga semua pegawai dapat melihat rekap laporan penyewaan lapangan futsal dan bulutangkis, sehingga untuk keamanan data masih belum terjamin, karena dapat mengalami kehilangan data yang disebabkan oleh pegawai lain.	Sistem baru dapat digunakan yang mempunyai hak akses masing-masing sehingga sehingga terjaga keamanan data serta tanggung jawab setiap bagiannya
Efisien	Pengolahan data serta penyewaan lapangan futsal dan bulutangkis kurang efisien karena masih datang langsung	Sistem yang diusulkan untuk mengolah data serta membuat penyewaan lapangan berbasis <i>website</i> akan

ANALISIS	SISTEM LAMA	SISTEM USULAN
	serta mencatat manual dan membuat waktu menjadi lama untuk merekapnya.	mempercepat proses kinerja dalam mem- <i>booking</i> lapangan futsal dan bulutangkis.
Pelayanan	Belum adanya fasilitas informasi penyewaan lapangan futsal dan bulutangkis secara <i>online</i> untuk mempermudah para penyewa serta mempermudah pimpinan dan pegawai.	Penambahan fasilitas informasi tentang penyewaan lapangan futsal dan bulutangkis menjadi suatu layanan yang dapat membantu memperoleh jadwal lapangan futsal dan bulutangkis tanpa harus datang langsung ketika akan menyewa sesuai dengan jam keinginannya secara <i>online</i> melalui <i>website</i> .

### 2.7. Extreme Programming

Menurut Chandra & Kosdiana, (2019) *Extreme Programming (XP)* adalah bagian dari *Agile Process* dan merupakan pengembangan dari pendekatan *Rapid Application Development (RAD)*, yang berfokus pada pengkodean

sebagai aktivitas inti di semua tahap siklus pengembangan perangkat lunak. Pendekatan ini mengusulkan proses pengembangan yang lebih minimalis, pengujian bertahap dan dokumentasi yang sederhana terhadap kebutuhan pengguna ("*agile*") dibandingkan dengan metode tradisional untuk menciptakan perangkat lunak yang lebih baik yang memberikan penekanan komunikasi yang *intens*. Pengembangan *extreme programming* dapat dilihat menggunakan suatu alur tahapan pengembangan yang dapat dilihat pada Gambar 2.1.



**Gambar 2.1** *Extreme Programming*

Berdasarkan tahapan tersebut merupakan suatu paradigma yang diinginkan mencakup didalam seperangkat aturan dan praktik-praktik dalam empat konteks kegiatan kerangka kerja yaitu:

1. *Planning* (Perencanaan)

Perencanaan yang bertujuan awal untuk mendengarkan kebutuhan secara fungsional dan non fungsional serta membuat penggambaran sistem secara *interface* sebagai keperluan *outputnya*.

2. *Design* (Perancangan)

Penggunaan rancangan bertujuan untuk membangun dan menggambarkan sistem yang akan dibuat guna mempermudah proses pengkodean.

### 3. *Coding* (Pengkodean)

Pengkodean bertujuan untuk membuat hasil perancangan dengan melakukan konstruksi atau pembuatan sistem secara seluruh menggunakan kode.

### 4. *Testing* (Pengujian)

Pengujian dilakukan untuk menguji hasil pengkodean dan selanjutnya dilakukan evaluasi dan implementasi.

#### **2.7.1. Kelebihan Metode *Extreme Programming***

Menurut Chandra & Kosdiana, (2019) Komunikasi dalam XP dibangun dengan melakukan pemrograman berpasangan (*pair programming*). *Developer* didampingi oleh pihak klien dalam melakukan *coding* dan *unit testing* sehingga klien bisa terlibat langsung dalam pemrograman sambil berkomunikasi dengan *developer*. Selain itu perkiraan beban tugas juga diperhitungkan.

1. Menekankan pada kesederhanaan dalam pengkodean. Komunikasi yang lebih banyak mempermudah, dan rancangan yang sederhana mengurangi penjelasan. (*Planning Phase*)
2. Menurunkan biaya pengembangan (*Implementation Phase*). Setiap *feedback* ditanggapi dengan melakukan tes, *unit test* atau *system integration* dan jangan menunda karena biaya akan membengkak (uang, tenaga, waktu).
3. Meningkatkan komunikasi dan sifat saling menghargai antar *developer*. (*Implementation Phase*). Banyak ide baru dan berani mencobanya, berani mengerjakan kembali dan setiap kali kesalahan ditemukan, langsung diperbaiki.

4. XP merupakan metodologi yang semi formal. Planning Developer harus selalu siap dengan perubahan karena perubahan akan selalu diterima, atau dengan kata lain eksibel. (*Maintenance Phase*)

### **2.7.2. Kelemahan Metode *Extreme Programming***

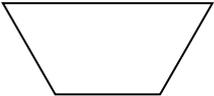
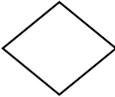
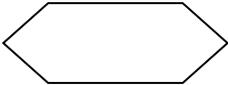
Menurut Chandra dan Kosdiana, (2019) Pengembangan *extreme programming* yang digunakan tentunya masih memiliki beberapa kelemahan atau kekurangan seperti berikut.

1. *Developer* harus selalu siap dengan perubahan karena perubahan akan selalu diterima.
2. Tidak bisa membuat kode yang detail di awal (*prinsip simplicity* dan juga anjuran untuk melakukan apa yang diperlukan hari itu juga).

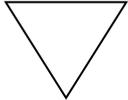
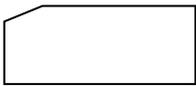
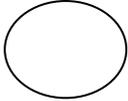
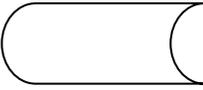
## **2.8. Bagan Alir Dokumen**

Bagan Alir Dokumen (*Dokument Flowchart*) (A. S. & Shalahudin, 2018) adalah Salah satu alat dan teknik yang digunakan dalam pengembangan sistem informasi, dokument *flowchart* merupakan bagan alir yang menunjukkan arus dari laporan dan formulir termasuk tembusannya. Simbol-simbol yang dipergunakan dalam pembuatan bagan alir dokumen dapat dilihat pada tabel 2.3.

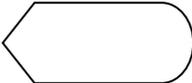
**Tabel 2. 3** Simbol-simbol dalam Bagan Alir Dokumen (A. S. & Shalahudin, 2018)

No.	Simbol	Keterangan
1.		<p><b>Simbol <i>Offline Connector</i></b>            Untuk menyatakan sambungan dari satu proses ke proses lainnya dalam halaman/lembar yang berbeda.</p>
2.		<p><b>Simbol Manual</b>            Untuk menyatakan suatu tindakan (proses) yang tidak dilakukan oleh komputer.</p>
3.		<p><b>Simbol <i>Decision/Logika</i></b>            Untuk menunjukkan suatu kondisi tertentu yang menghasilkan dua kemungkinan jawaban, ya/tidak.</p>
4.		<p><b>Simbol <i>Predefined Proses</i></b>            Untuk menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal.</p>

**Tabel 2. 3** Simbol-simbol dalam Bagan Alir Dokumen (A. S. & Shalahudin, 2018) (Lanjutan)

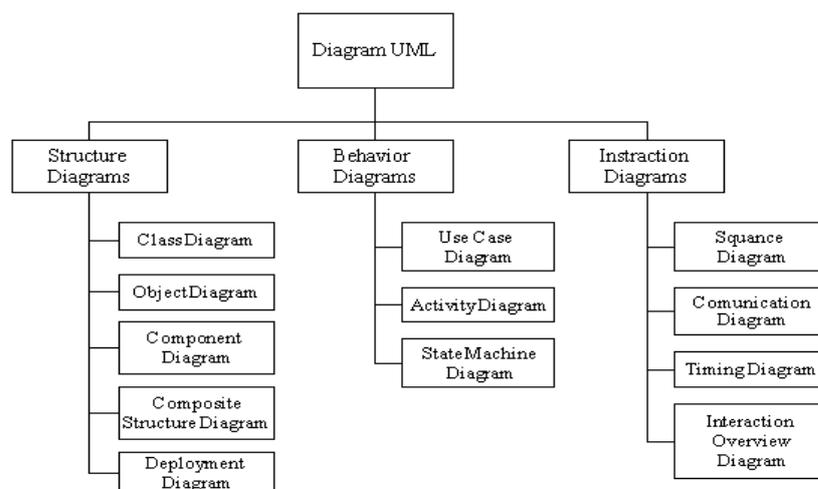
No.	Simbol	Keterangan
5.		<p><b>Simbol Terminal</b> Untuk menyatakan permulaan atau akhir suatu program.</p>
6.		<p><b>Simbol Keying Operation</b> Untuk menyatakan segala jenis operasi yang diproses dengan menggunakan suatu mesin yang mempunyai <i>keyboard</i>.</p>
7.		<p><b>Simbol Off-Line Storage</b> Untuk menunjukkan bahwa data dalam simbol ini akan disimpan ke suatu media tertentu.</p>
8.		<p><b>Simbol Manual Input</b> Untuk memasukkan data secara manual dengan menggunakan <i>online keyboard</i>.</p>
9.		<p><b>Simbol Input-Output</b> Untuk menyatakan proses <i>input</i> dan <i>output</i> tanpa tergantung dengan jenis peralatanya.</p>
10.		<p><b>Simbol Punched Card</b> Untuk menyatakan input berasal dari kartu atau <i>output</i> ditulis ke kartu</p>
11.		<p><b>Simbol Magnetic-tape Unit</b> Untuk menyatakan input berasal dari pita magnetic atau <i>output</i> disimpan ke pita magnetic.</p>
12.		<p><b>Simbol Disk storage</b> Untuk menyatakan input berasal dari disk atau <i>output</i> disimpan ke <i>disk</i>.</p>

**Tabel 2. 3** Simbol-simbol dalam Bagan Alir Dokumen (A. S. & Shalahudin, 2018) (Lanjutan)

No.	Simbol	Keterangan
13.		<b>Simbol Document</b> Untuk mencetak laporan ke printer.
14.		<b>Simbol Display</b> Untuk menyatakan peralatan <i>output</i> yang digunakan berupa layar (video, komputer).
15.		<b>Simbol Connector</b> Untuk menyatakan sambungan dari satu proses ke proses lainnya dalam halaman/lembar yang sama.

## 2.9. UML (*Unified Modelling Language*)

Menurut A. S. & Shalahudin (2018) UML (*Unified Modelling Language*) adalah bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. Berikut adalah bagan UML pada Gambar 2.2.

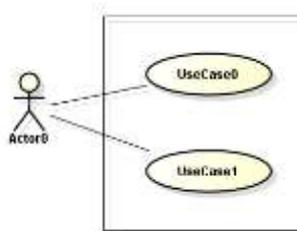


**Gambar 2. 2** Bagan UML

Berikut ini merupakan penjelasan tentang masing-masing diagram yang ada pada UML (*Unified Modelling Language*).

### 2.9.1. Use Case Diagram

Menurut A. S. & Shalahudin (2018) *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 adalah konsep *Use Case Diagram* pada Gambar 2.3.



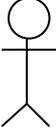
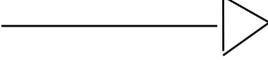
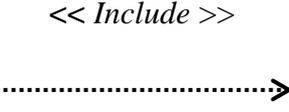
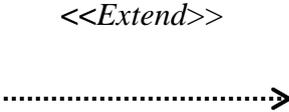
**Gambar 2.3** Use Case Diagram

Berikut simbol-simbol yang akan digunakan dalam menggambarkan *Use Case Diagram* dapat dilihat pada tabel 2.4 berikut ini:

**Tabel 2.4** Simbol *Use Case Diagram* (A. S. & Shalahudin, 2018)

No	Simbol	Deskripsi
1.		<i>Use case</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> .

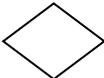
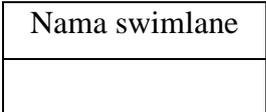
**Tabel 2. 4** Simbol *Use Case Diagram* (A. S. & Shalahudin, 2018) (Lanjutan)

No	Simbol	Deskripsi
2.		Aktor: seseorang/sesuatu yang berinteraksi dengan yang akan dibuat. diluar sistem informasi. Biasanya dinyatakan menggunakan kata benda
3.		Asosiasi ( <i>association</i> ): merupakan komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
4.		Generalisasi ( <i>generalization</i> ): merupakan hubungan (umum – khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum
5.		<i>Include</i> berarti <i>use case</i> yang ditambahkan akan dipanggil saat <i>use case</i> tambahan dijalankan.
6.		Ekstensi ( <i>extend</i> ) merupakan <i>use case</i> tambahan ke sebuah <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu.

### 2.9.2. Activity Diagram

Menurut A. S. & Shalahudin (2018) *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 2.5 berikut ini:

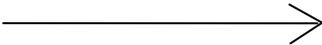
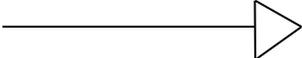
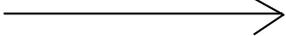
**Tabel 2. 5** Simbol *Activity Diagram* (A. S. & Shalahudin, 2018)

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.
5.		Swimlane Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas.
6.		Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.

### 2.9.3. *Class Diagram*

Menurut A. S. & Shalahudin (2018) *Class Diagram* adalah mengembangkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Berikut simbol-simbol yang akan digunakan dalam menggambarkan *Class Diagram* dapat dilihat pada tabel 2.6 berikut ini:

**Tabel 2. 6** Simbol *Class Diagram* (A. S. & Shalahudin, 2018)

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

### 2.10. Balsamiq Mockup

Menurut Firanda et al. (2021) *Balsamiq Mockup* merupakan *software* yang digunakan pada pembuatan desain atau *prototyping* pada pembuatan *user interface* sebuah aplikasi. Menggunakan *balsamiq mockup* dapat memudahkan dalam pembuatan *user interface*, karena *balsamiq mockup* telah menyediakan

*tools* yang dapat memudahkan dalam membuat desain *prototyping* aplikasi yang akan dirancang.

### **2.11.PHP**

Menurut Maliki et al. (2021) *PHP* artinya bahasa pemrograman berbasis web yang mempunyai kemampuan memproses data secara dinamis. *PHP* menjadi *server-side embedded script language* merupakan sintaks-sintaks serta perintah yang diberikan akan sepenuhnya dijalankan namun disertakan laman *HTML* biasa. Perangkat lunak yang dibangun oleh *PHP* umumnya akan menghasilkan *web browser*, namun prosesnya secara menyeluruh ketika dijalankan pada *server*.

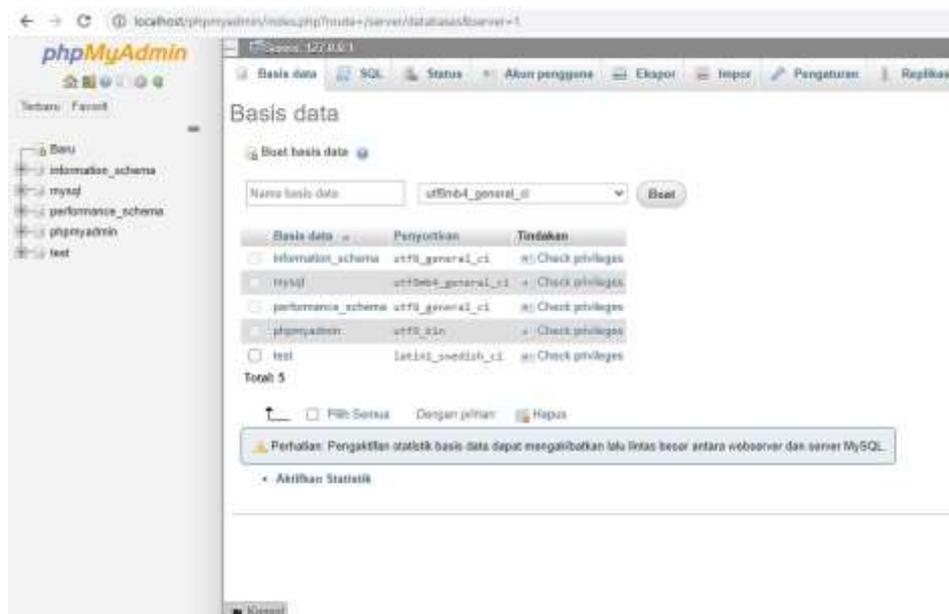
### **2.12.CodeIgniter**

Menurut Sallaby & Kanedi (2020) *CodeIgniter* merupakan sebuah *framework PHP* yang bersifat *open source* serta menggunakan metode *MVC* (*Model, View, Controller*) untuk memudahkan *developer* atau *programmer* untuk menghasilkan sebuah aplikasi yang berbasis web tanpa harus membuatnya dari awal.

### **2.13. MySql**

*MySQL* menurut Sabar et al. (2019) merupakan basis data yang bersifat *open source* sehingga banyak di gunakan untuk media. Walaupun gratis, *MySQL* tetap berkualitas dan sudah cukup memberikan *performance* yang memadai. Penggunaan *PHP MyAdmin* lebih mudah digunakan karena menggunakan

*interface* yang lebih mudah dipahami, berikut adalah tampilan awal *MySQL* pada Gambar 2.4.



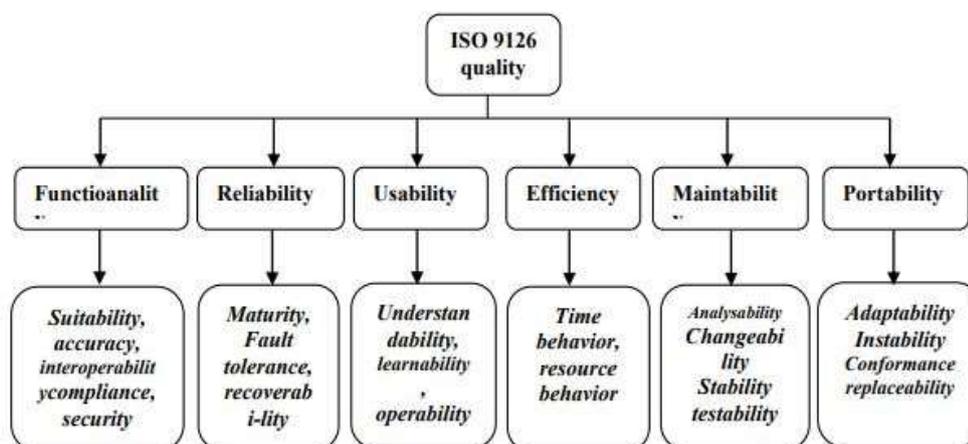
**Gambar 2. 4** *MySql*

Adapun kelebihan *MySQL* sebagai seperti *multi-user*, memiliki tipe data dan variabel, memiliki struktur keamanan yang baik, alat administrasi yang lengkap, dapat diintegrasikan dengan bahasa pemrograman lain, struktur tabel yang fleksibel, tetapi masih memiliki sedikit kekurangan seperti sulit diaplikasikan pada perusahaan yang besar dan tidak populer untuk aplikasi *game* dan *mobile*.

#### **2.14. ISO 9126**

Menurut Banjarnahor et al. (2018) ISO/IEC 9126 merupakan pengembangan dari *ISO 9001*, salah satu standar internasional dalam pengukuran serta pengevaluasian yang diterbitkan oleh Organisasi Internasional untuk Standarisasi (*International Organization for Standardization*) atau disingkat *ISO*. *ISO 9126* ialah model dengan kualitas yang lengkap serta mempunyai

analisa lebih baik dari contoh lainnya. Ada enam karakteristik dari *ISO 9126* yakni *functionality*, *reliability*, *usability*, *efficiency*, *maintainability*, dan *portability*. Karakteristik tersebut akan dianalisis dan dipilih untuk digunakan dalam menganalisa kualitas sistem yang diteliti. Dalam *ISO 9126* model evaluasi tidak didefinisikan secara spesifik sehingga peneliti dapat menentukan sendiri model evaluasinya sesuai dengan kebutuhan. Sehingga menjadi salah satu bagian kelebihan dari standar *ISO 9126*, berikut ini adalah karakteristik dan sub-karakteristik *ISO 9126* pada Gambar 2.5.



**Gambar 2. 5** Karakteristik dan Sub-karakteristik ISO 9126

Berikut ini penjelasan dari Gambar 2.5 tentang karakteristik dan sub-karakteristik dalam *ISO 9126*.

#### **2.14.1. Functionality**

Functionality adalah "kemampuan perangkat lunak untuk menyediakan fungsionalitas yang ditentukan pengguna atau dapat digunakan" (Supriyono et al., 2019). Subkarakteristik *functionality* meliputi *suitability*, *accuracy*, *interoperability*, dan *security*. Berikut ini penjelasan untuk masing-masing sub-karakteristik *functionality*, dapat di lihat pada tabel 2.7.

**Tabel 2. 7** Sub-Karakteristik *Functionality* (Supriyono et al., 2019)

<b>Karakteristik Functionality</b>	
<b>Sub-karakteristik</b>	<b>Penjelasan</b>
<i>Suitability</i>	Apakah software dapat melakukan tugas yang diperlukan?
<i>Accuracy</i>	Apakah hasil sesuai dengan yang diharapkan?
<i>Interoperability</i>	Dapatkah sistem berinteraksi dengan sistem yang lain?
<i>Security</i>	Apakah software menghalangi <i>unauthorized access</i> ?

#### **2.14.2. Reliability**

*Reliability* merupakan kemampuan suatu produk perangkat lunak (*software*) dalam mempertahankan tingkat kinerjanya pada kondisi tertentu yang telah ditetapkan pada periode waktu yang ditentukan (Supriyono et al., 2019). Sub-karakteristik *reliability* meliputi *maturity*, *fault tolerance*, dan *recoverability*. Penjelasan untuk masing-masing subkarakteristik *reliability* dapat dilihat pada tabel 2.8.

**Tabel 2. 8** Karakteristik *Reliability* (Supriyono et al., 2019)

<b>Karakteristik Reliability</b>	
<b>Sub-karakteristik</b>	<b>Penjelasan</b>
<i>Maturity</i>	Seberapa banyak kesalahan dalam <i>software</i> dapat dihapuskan dalam waktu tertentu ?
<i>Fault Tolerance</i>	Apakah <i>software</i> mampu menangani kesalahan ?
<i>recoverability</i>	Dapatkah <i>software</i> meneruskan kinerja dan mengembalikan data yang hilang setelah terjadi gangguan?

### 2.14.3. Usability

*Usability* adalah seperangkat perilaku yang mengukur kegunaan dan kepuasan yang dirasakan dari sebuah sistem untuk pengguna yang secara langsung maupun tidak langsung (Supriyono et al., 2019). Sub-karakteristik *Usability* meliputi *understandability*, *learnability*, *operability*, dan *attractiveness*. Penjelasan untuk masing-masing sub-karakteristik *usability* dapat dilihat pada tabel 2.9.

**Tabel 2. 9** Sub-Karakteristik *Usability* (Supriyono et al., 2019)

<b>Karakteristik Usability</b>	
<b>Sub-karakteristik</b>	<b>Penjelasan</b>
<i>Understandability</i>	Apakah pengguna memahami bagaimana menggunakan sistem dengan mudah?
<i>Learnability</i>	Dapatkah pengguna belajar menggunakan sistem dengan mudah?
<i>Operability</i>	Dapatkah pengguna menggunakan sistem tanpa upaya yang lebih?
<i>Attractiveness</i>	Apakah antarmuka terlihat baik?

#### 2.14.4. *Efficiency*

*Efficiency* adalah “kemampuan perangkat lunak untuk menyediakan fungsionalitas yang memadai dalam kaitannya dengan perangkat keras yang digunakan” (Supriyono et al., 2019). *Efficiency* adalah konsep kompleks yang menyajikan masalah konseptual dan praktis. Sub-karakteristik *efficiency* meliputi *time behavior* dan *resource utilization*. Penjelasan untuk masing-masing subkarakteristik *efficiency* dapat dilihat pada tabel 2.10.

**Tabel 2. 10** Sub-Karakteristik *Efficiency* (Supriyono et al., 2019)

<b>Karakteristik Efficiency</b>	
<b>Sub-karakteristik</b>	<b>Penjelasan</b>
<i>Time Behaviour</i>	Seberapa cepat respon sistem?
<i>Resource Unlisation</i>	Apakah sistem memanfaatkan sumber daya secara efisien?

### 2.14.5. Maintainability

*Maintainability* adalah atribut yang digunakan dalam upaya untuk membuat perubahan tertentu (Supriyono et al., 2019). Sub-karakteristik *maintainability* meliputi *analyzability*, *changeability*, *stability*, dan *testability*. Penjelasan untuk masing-masing sub-karakteristik *maintainability* dapat dilihat pada tabel 2.11.

**Tabel 2. 11** Sub-Karakteristik *Maintainability* (Supriyono et al., 2019)

Karakteristik Maintability	
Sub-karakteristik	Penjelasan
<i>Analyzability</i>	Dapatkah kesalahan didiagnosa dengan mudah?
<i>Changeability</i>	Dapatkah <i>software</i> dimodifikasi dengan mudah?
<i>Stability</i>	Dapatkah <i>software</i> tetap berfungsi jika ada perubahan?
<i>Testability</i>	Dapatkah <i>software</i> dapat diuji dengan mudah?

### 2.14.6. Portability

*Portability* adalah kemampuan produk perangkat lunak untuk diangkut dari satu lokasi ke lokasi lainnya (Supriyono et al., 2019). Produk perangkat lunak dalam *e-book* harus dengan mudah ditransfer ke sumber lain (seperti teknologi canggih) sesuai kebutuhan. Sub-karakteristik *portability* meliputi *adaptability*, *installability*, *co-existence*, dan *replaceability*. Penjelasan untuk masing-masing sub-karakteristik *portability* dapat dilihat pada tabel 2.12.

**Tabel 2. 12** Sub-Karakteristik *Portability* (Supriyono et al., 2019)

<b>Karakteristik Protability</b>	
<b>Sub-karakteristik</b>	<b>Penjelasan</b>
<i>Adaptability</i>	Dapatkah <i>software</i> dipindahkan ke lingkungan lain?
<i>Installability</i>	Dapatkah <i>software</i> di-install dengan mudah?
<i>Coexistence</i>	Apakah <i>software</i> dapat bekerja dengan sistem software lain?
<i>Replaceability</i>	Dapatkah <i>software</i> dengan mudah menggantikan <i>software</i> lain?

### 2.15. Skala Pengukuran

Menurut Sugiyono (2018), skala pengukuran merupakan kesepakatan yang digunakan sebagai acuan untuk menentukan panjang pendeknya interval yang ada dalam alat ukur, sehingga alat ukur tersebut bila digunakan dalam pengukuran akan menghasilkan data kuantitatif.

#### 2.15.1. Skala Likert

Menurut Sugiyono (2018), *skala likert testing* adalah *Skala Likert* digunakan untuk mengukur sikap, pendapat, dan persepsi seseorang atau sekelompok orang tentang fenomena sosial. Dalam penelitian fenomena sosial ini telah ditetapkan secara spesifik oleh peneliti, yang selanjutnya disebut sebagai variabel penelitian.

Dengan *skala likert* maka variabel yang akan diukur dijabarkan menjadi indikator variabel. Kemudian indikator variabel tersebut dijadikan sebagai titik tolak untuk menyusun item-item instrumen yang dapat berupa pernyataan atau

pertanyaan. Pengujian menggunakan 5 kategori jawaban dengan bobot yang berbeda untuk setiap jawabannya seperti pada tabel 2.13.

**Tabel 2. 13** Jenjang Dalam *Skala Likert* (Sugiyono, 2018)

<b>Pernyataan positif</b>	<b>Nilai</b>	<b>Pernyataan Negatif</b>	<b>Nilai</b>
Sangat setuju	5	Sangat setuju	1
Setuju	4	Setuju	2
Ragu-ragu (Netral)	3	Ragu-ragu (Netral)	3
Tidak Setuju	2	Tidak Setuju	4
Sangat tidak setuju	1	Sangat tidak setuju	5

Rumus perhitungan skala Likert adalah sebagai berikut:

$$\% \text{ Skor Aktual} = \frac{\text{Skor Aktual}}{\text{Skor Ideal}} \times 100\%$$

Keterangan :

1. Skor aktual adalah hasil jawaban seluruh responden atas kuisoer yang telah diajukan.
2. Skor ideal adalah nilai tertinggi atau semua responden diasumsukan memilih jawaban dengan skor tertinggi.

Kemudian hasil perhitungan yang didapatkan dari angket, selanjutnya dibandingkan dengan rentang kriteria interpretasi skor untuk menyatakan hasil yang didapatkan dengan rentang pada tabel 2.14.

**Tabel 2. 14** Rentang Kriteria Interpretasi (Sugiyono, 2018)

<b>No</b>	<b>Rentang Kriteria</b>	<b>Kriteria</b>
1	0% - 20%	Sangat Tidak Baik
2	21% - 40%	Tidak Baik
3	41% - 60%	Kurang Baik
4	61% - 80%	Baik
5	81% - 100%	Sangat Baik