

## BAB II LANDASAN TEORI

### 2.1. Tinjauan Pustaka

Untuk mendukung penelitian ini diperlukan tinjauan pustaka yang diambil dari beberapa jurnal penelitian yang berkaitan dengan judul penelitian ini dan pokok bahasan, berikut ini merupakan penelitian terdahulu dapat dilihat pada tabel 2.1.

**Tabel 2.1** Tinjauan Pustaka

No Literatur	Penulis	Tahun	Judul
Literatur 01	Rahmansyah & Nurrahmi	2019	Perancangan Sistem Informasi Penjadwalan Kegiatan Kepegawaian Berbasis Web
Literatur 02	Fadila & Djaelani	2019	Perancangan Sistem Informasi Manajemen Kegiatan Pegawai Pada Kantor Urusan Agama (KUA)
Literatur 03	Siregar & Ani	2019	Perancangan Sistem Reporting Kerja Karyawan Berbasis Web Pada Ruangan Network Operation Center Bagian Front Office (Studi Kasus: Network Operation Center XI Axiata)
Literatur 04	Puryani et al	2018	Perancangan Sistem Kerja Untuk Meningkatkan Produktivitas Dengan Pendekatan Sistem Siosioteknik
Literatur 05	Raharja et al	2018	Penjadwalan Agenda Pelaksanaan Tridharma Perguruan Tinggi Secara Online Menggunakan Google Calendar
Literatur 06	Mara Destiningrum dan Adrian	2017	Sistem Informasi Penjadwalan Dokter Berbasis Web Dengan Menggunakan Framework Codeigniter ( Studi Kasus : Rumah Sakit Yukum Medical Centre )
Literatur 07	Hendradi dan Anggraini	2016	Perancangan Sistem Informasi Manajemen Agenda Kegiatan Pertemuan Usni Berbasis Web (Studi Kasus : Ppk Usni)

### **2.1.1. Literatur 01**

BKKBN masih menerapkan sistem manual dan belum adanya penyimpanan data di dalam suatu sistem database yaitu penjadwalan kegiatan sehingga kegiatan yang seharusnya selesai tepat waktu menjadi terhambat dan rentan terhadap kesalahan. Untuk mengembangkan sistem, digunakan metode waterfall yang mencakup tahap Requirement Analisis, System Design, Implementation, Integration dan Testing, dan Operation dan Maintenance. Aplikasi yang selanjutnya dikembangkan dengan PHP dan MySQL. Dengan adanya sistem informasi penjadwalan kegiatan kepegawaian dapat memudahkan staf dan pejabat dilingkungan Badan Kependudukan dan Keluarga Berencana Nasional untuk melaksanakan tugas-tugas keseharian dan memudahkan dalam hal penjadwalan dan monitoring kegiatan kepegawaian.

### **2.1.2. Literatur 02**

KUA merupakan salah satu instansi pemerintah yang bergerak di bidang keagamaan, baik dalam hal pelayanan pernikahan, pelayanan wakaf, pelayanan haji, pelayanan pendaftaran talak atau rujuk, serta berbagai layanan dalam bidang keagamaan yang lainnya. Berdasarkan penelitian yang kami buat saat ini sistem pengolah data kegiatan pegawai telah menggunakan komputer namun hanya bisa mengolah beberapa kegiatan pegawai saja, seperti pelayanan pernikahan dan pelayanan wakaf, sedangkan dalam pembuatan laporan kegiatan sehari-hari belum melibatkan penggunaan teknologi komputer sehingga dalam pencatatan laporan menjadi tidak sama seperti keadaan di lingkungan kerja. Aplikasi Sistem

Informasi Manajemen Kegiatan Pegawai ini dirancang untuk menyelesaikan permasalahan yang ada sehingga diharapkan bisa mengolah data semua kegiatan dan membuat laporan secara efisien sehingga diharapkan dapat mengurangi potensi kesalahan dalam pengolahan data.

### **2.1.3. Literatur 03**

Sistem reporting ini nantinya akan membantu pimpinan perusahaan dalam memantau pekerjaan karyawan. Sistem reporting yang dirancang dimaksudkan untuk mempermudah, mempercepat dan mengurangi tingkat kesalahan dalam pengolahan data pekerjaan karyawan, data jenis pekerjaan dan juga data kinerja karyawan yang ada di PT. Huawei Service bagian Network Operation Center (NOC). Adapun permasalahan pada ruangan NOC adalah sistem reporting kerja karyawan yang tidak efisien dengan pengiriman email kepada pimpinan, pimpinan perusahaan tidak membutuhkan banyak waktu untuk pengecekan email. Sistem reporting karyawan yang dirancang nantinya akan digunakan mengubah sistem reporting yang ada di NOC yang semula melalui pengiriman email menjadi berbasis web. Dimana website ini akan memiliki fitur untuk mengarsipkan seluruh data pekerjaan karyawan berupa report pekerjaan yang sudah selesai dan juga report pekerjaan yang masih dalam proses. Dalam perancangan sistem reporting karyawan berbasis web ini metode pengumpulan data menggunakan teknik wawancara dan observasi. Metode yang digunakan adalah dengan model pengembangan Waterfall melalui tahapan analysis dan design.

### **2.1.4. Literatur 04**

Proses produksi yang terdapat pada UKM ini masih sangat sederhana sehingga tingkat produktivitasnya pun juga rendah. Oleh sebab itu, pemilik ingin menaikkan kemampuan produksinya dengan mengganti proses pengiratan dengan teknologi yang baru yang memiliki kemampuan produksi yang lebih tinggi. Dengan demikian, perlu dilakukan pemilihan alternatif teknologi yang baru sesuai keinginan pemilik UKM. Selanjutnya dilakukan penerapan ergonomi makro berdasarkan pendekatan sistem sosioteknik untuk merancang sistem kerja yang baru dengan mempertimbangkan keinginan teknologi dari pemilik UKM. Pada penelitian ini, pemilihan alternatif teknologi menggunakan metode Analytical Hierarchy Process (AHP). Pemilihan alternatif yang tepat nantinya akan berdampak pada desain sistem kerja yang baru. Penelitian ini bertujuan untuk membuat desain ulang sistem kerja yang diinginkan untuk meningkatkan produktivitas dengan penerapan ergonomi makro yang menggunakan pendekatan sistem sosioteknik dari pemilihan alternatif yang terpilih.

#### **2.1.5. Literatur 05**

Saat ini banyak sekali perubahan-perubahan yang ditunjang oleh berbagai teknologi baru. Banyak pula penunjang untuk mempermudah kegiatan pendidikan. Salah satunya yaitu kalender. Tidak hanya digunakan untuk melihat tanggal bulan dan tahun, kalender ternyata memiliki berbagai fungsi lainnya. Yaitu salah satunya berfungsi sebagai pengingat atau reminder. Selain itu bisa digunakan untuk membuat note atau catatan pada hari tersebut. Dari berbagai macam aktivitas atau kegiatan yang kita akan dan sedang dilakukan dapat di buat kan catatan dan pengingat dari setiap kegiatan tersebut. Setiap jadwal atau reminder yang sudah dibuat dapat juga menampilkan notifikasi pengingatnya.

Penggunaan google calendar pun dapat digunakan oleh berbagai kalangan mulai dari murid Sekolah sampai Orang Tua sekali pun baik personal maupun kelompok atau organisasi. Penggunaanya juga terbilang mudah karena user interface nya yang userfriendly.

#### **2.1.6. Literatur 06**

Rumah Sakit Yukum Medical Centre merupakan perusahaan yang bergerak pada bidang kesehatan yang berdiri di Yukum Jaya Kabupaten Lampung Tengah. Sistem penjadwalan di Rumah Sakit Yukum Medical Centre masih menggunakan kertas apabila pasien ingin mengetahui informasi jadwal dokter dapat menanyakan kebagian informasi. Faktor kesehatan merupakan salah satu faktor yang penting dalam kehidupan masyarakat. Terutama untuk mendapatkan layanan kesehatan ketika sakit atau membutuhkan pelayanan kesehatan seperti hanya sekedar konsultasi. Hasil pengujian yang telah dilakukan dapat disimpulkan bahwa dengan adanya sistem Informasi Penjadwalan Dokter Berbasis Web dengan menggunakan Framework Codeigniter dapat memberikan informasi jadwal praktik dokter dengan cepat dan akurat dan menjadikan proses penjadwalannya lebih efisien

#### **2.1.7. Literatur 07**

Selama ini banyak kegiatan pertemuan yang dilakukan Fakultas yang sering kali tidak diketahui oleh para mahasiswa, akibat nya, kegiatan pertemuan tidak banyak dihadiri para peserta karena kurangnya informasi kegiatan pertemuan dan pelaksanaan kegiatan. Oleh karena itu, dibuatlah suatu Aplikasi Sistem Informasi Manajemen Agenda Kegiatan Pertemuan USNI berbasis web yang bertujuan untuk memudahkan pengguna mengetahui informasi kegiatan pertemuan yang

akan dilaksanakan. Sistem ini tidak bisa berjalan dengan sendirinya tanpa adanya administrator, maka diperlukan administrator untuk mengelola informasi-informasi yang akan ditampilkan. Metode yang digunakan penulis dalam penelitian ini adalah model Waterfall. Penelitian ini diharapkan dapat membantu user untuk mendapatkan informasi mengenai berbagai macam kegiatan yang ada di USNI.

Berdasarkan beberapa penelitian yang telah dilakukan terdapat beberapa perbedaan dengan peneliti yaitu :

- 1) Metode pengembangan sistem yang digunakan yaitu *extreme programming*
- 2) Metode pengujian yang digunakan yaitu ISO 25010 dengan aspek *functionality* dan *reability*
- 3) Sistem yang dibangun dapat diakses oleh admin dan anggota
- 4) Sistem dapat melakukan pengupload-an dan pendownload-an berkas
- 5) Sistem dapat mencetak hasil kegiatan tugas kerja secara periode.

## **2.2. Landasan Teori**

Dalam mendukung penelitian ini digunakan beberapa tinjauan studi yang sesuai dan berkaitan dengan pokok bahasan sebagai berikut :

### **2.2.1. Perancangan**

Perancangan adalah penggambaran, perencanaan, dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah kedalam suatu kesatuan yang utuh dan berfungsi. Dengan demikian pengertian rancang bangun merupakan kegiatan menerjemahkan hasil analisa ke dalam bentuk paket perangkat lunak

kemudian menciptakan sistem tersebut atau memperbaiki sistem yang sudah ada (Pressman, 2015)

### **2.2.2. Sistem Informasi**

Sistem dapat didefinisikan sebagai kumpulan dari elemen-elemen berupa data, jaringan kerja dari prosedur-prosedur yang saling berhubungan, sumber daya manusia, teknologi baik hardware dan software yang saling berinteraksi sebagai kesatuan untuk mencapai tujuan atau sasaran tertentu yang sama (Maniah dan Hamidin, 2017).

sistem adalah sekumpulan unsur atau elemen dan komponen-komponen yang saling berkaitan dan saling mempengaruhi dalam melakukan kegiatan bersama untuk mencapai suatu tujuan tertentu (Fernanda dan Fernando, 2017)

Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya (Wati et al., 2019)

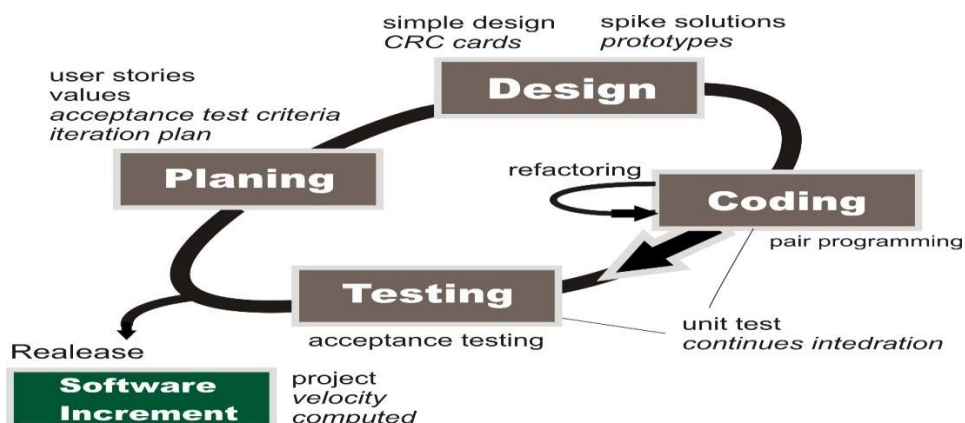
Informasi merupakan kumpulan data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerima. Tanpa suatu informasi, suatu sistem tidak akan berjalan dengan lancar dan akhirnya bisa mati. Suatu organisasi tanpa adanya suatu informasi maka organisasi tersebut tidak bisa berjalan dan tidak bisa beroperasi (Kristanto, 2018).

Sistem informasi ialah suatu sistem yang menyajikan informasi yang bertujuan untuk mengambil suatu keputusan dan menjalankan kegiatan operasional suatu organisasi dan memberikan suatu keunggulan kompetitif (Awaludin et al., 2019)

Sistem informasi adalah suatu sistem yang dibuat oleh manusia yang terdiri dari komponen-komponen dalam organisasi untuk mencapai suatu tujuan yaitu menyajikan informasi (Adrian & Pramono, 2017)

### 2.2.3. Metode Pengembangan Sistem *Extreme Programming*

*Extreme Programming* (XP) adalah metodologi pengembangan perangkat lunak yang ditujukan untuk meningkatkan kualitas perangkat lunak dan tanggap terhadap perubahan kebutuhan pelanggan. Jenis pengembangan perangkat lunak semacam ini dimaksudkan untuk meningkatkan produktivitas dan memperkenalkan pos pemeriksaan dimana persyaratan pelanggan baru dapat diadopsi. Tahapan-tahapan dari *Extreme Programming* terdiri dari *planning* seperti memahami kriteria pengguna dan perencanaan pengembangan, *designing* seperti perancangan *prototype* dan tampilan, *coding* termasuk pengintegrasian, dan yang terakhir adalah *testing* (Pressman, 2012)



**Gambar 2. 1** Model *Extreme Programming* (XP)  
Sumber : (Lestari et al., 2020)

Proses *Extreme Programming* (XP) menurut (Pressman, 2012)

1. *Planning* : Tahap *planning* dimulai dengan membuat *user stories* yang menggambarkan output, fitur, dan fungsi - fungsi dari *software*



yang akan dibuat. *User stories* tersebut kemudian diberikan bobot seperti prioritas dan dikelompokkan untuk selanjutnya dilakukan proses *delivery* secara incremental.

2. *Design* : *Design* di *Extreme Programming* mengikuti prinsip *Keep It Simple* (KIS). Untuk *design* yang sulit, *Extreme Programming* akan menggunakan *Spike Solution* dimana pembuatan *design* dibuat langsung ke tujuannya. *Extreme Programming* juga mendukung adanya *refactoring* dimana *software system* diubah sedemikian rupa dengan cara mengubah struktur kode dan menyederhanakannya namun hasil dari kode tidak berubah.
3. *Coding* : Proses *coding* pada *Extreme Programming* diawali dengan membangun serangkaian unit test. Setelah itu pengembangan akan berfokus untuk mengimplementasikannya. Dalam *Extreme Programming* diperkenalkan istilah *Pair Programming* dimana proses penulisan program dilakukan secara berpasangan. Dua orang *Programmer* saling bekerjasama di satu komputer untuk menulis program. Dengan melakukan ini akan didapat *real-time problem solving* dan *real-time quality assurance*.
4. *Testing* : Tahap ini dilakukan pengujian kode pada unit test. Dalam *Extreme Programming*, diperkenalkan *Extreme Programming acceptance test* atau biasa disebut *customer test*. Tes ini dilakukan oleh customer yang berfokus kepada fitur dan fungsi sistem secara keseluruhan. *Acceptance test* ini berasal dari *user stories* yang telah diimplementasikan.

#### 2.2.4. Karakteristik Pemodelan Berorientasi Objek

Berikut ini adalah beberapa karakteristik dasar yang harus dipahami tentang metodologi berorientasi objek:

1. Kelas (*Class*)

Kelas adalah kumpulan objek-objek dengan karakteristik yang sama. Sebuah kelas akan mempunyai sifat (atribut), kelakuan (operasi/metode), hubungan (relationship) dan arti.

2. Objek (*Object*)

Objek adalah abstraksi dan sesuatu yang mewakili dunia nyata seperti benda, manusia, satuan organisasi, tempat, kejadian, struktur, status, atau hal-hal lain yang bersifat abstrak.

3. Metode (*Method*)

Operasi atau metode atau method pada sebuah kelas hampir sama dengan fungsi atau prosedur pada metodologi struktural. Sebuah kelas boleh memiliki lebih dari satu metode atau operasi.

4. Atribut (*attribute*)

Atribut dari sebuah kelas adalah variabel global yang dimiliki sebuah kelas. Atribut dapat berupa nilai atau elemen-elemen data yang dimiliki oleh objek dalam kelas objek. Atribut dimiliki secara individual oleh sebuah objek, misalnya berat, jenis, nama, dan sebagainya.

5. Abstraksi (*abstraction*)

Prinsip untuk mempersentasikan dunia nyata yang kompleks menjadi satu bentuk model sederhana dengan mengabaikan aspek-aspek lain yang tidak sesuai dengan permasalahan.

6. Enkapsulasi (*encapsulation*)

Pembungkusan atribut data dan layanan (operasi-operasi) yang dipunyai objek untuk menyembunyikan implementasi dan objek sehingga objek lain tidak mengetahui cara kerjanya.

7. Pewarisan (*inheritance*)

Mekanisme yang memungkinkan satu objek mewarisi sebagian atau seuruh definisi dan objek lain sebagai bagian dan dirinya.

8. Antarmuka (*interface*)

Antarmuka atau interface sangat mirip dengan kelas, tapi hanya atribut kelas dan memiliki metode yang dideklarasikan tanpa isi.

9. *Reusabilily*

Pemanfaatan kembali objek yang sudah didefinisikan untuk suatu permasalahan pada permasalahan lainya yang melibatkan objek tersebut.

10. Generalisasi dan Spesialisasi

Menunjukkan hubungan antara kelas dan objek yang umum dengan kelas dan objek yang khusus. Misalnya kelas yang lebih umum (generalisasi) adalah kendaraan darat dan kelas khususnya (spesialisasi) adalah mobil,motor, dan kereta.

11. Komunikasi Antar Objek

Komunikasi antar objek dilakukan lewat pesan (message) yang dikirim dan satu objek keobjek lainnya.

## 12. Polimorfisme (*Polymorphism*)

Kemampuan suatu objek untuk digunakan dibanyak tujuan yang berbeda dengan nama yang sama sehingga menghemat baris program.

## 13. Package

*Package* adalah sebuah kontainer atau kemasan yang dapat digunakan untuk mengelompokkan kelas-kelas sehingga memungkinkan beberapa kelas yang bernama sama disimpan dalam package yang berbeda.

### 2.2.5. *Unified Modeling Language (UML)*

Bahasa Pemodelan Pengembangan Sistem (*Unified Modeling Language*) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek (Rosa dan Shalahudin, 2018) (Mukti et al., 2021). Beberapa jenis diagram *UML* antara lain sebagai berikut:

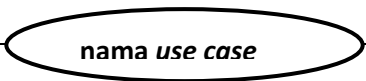
#### A. *Use Case Diagram*

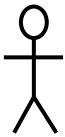


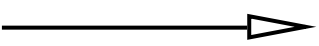

Menurut (Rosa dan Shalahudin, 2018).

“Use case diagram atau diagram use case merupakan pemodelan untuk kelakuan (behavior) sistem informasi yang akan dibuat”.

Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. 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
<i>Use Case</i> 	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya

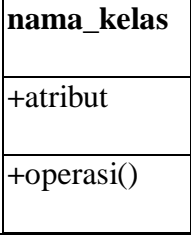



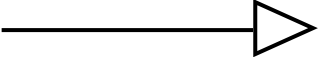

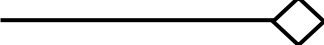
Simbol	Deskripsi
	dinyatakan dengan menggunakan kata kerja di awal frase nama <i>use case</i>
Aktor/ <i>actor</i> 	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda di awal frase nama <i>actor</i>
Asosiasi/ <i>association</i> 	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>
Ekstensi/ <i>extend</i>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan
<< <i>extend</i> >> 	dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek biasanya <i>use case</i> tambahan memiliki nama depan
Generalisasi/ <i>generalization</i> 	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
Menggunakan/ <i>Include/uses</i> << <i>include</i> >> 	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

Sumber: (Rosa dan Shalahudin, 2018).

## B. *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. simbol-simbol yang ada pada diagram kelas pada tabel *class diagram 2.3* di bawah ini:

Tabel 2. 3 Simbol *Class Diagram*



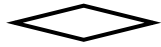

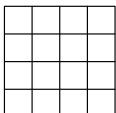


Simbol	Deskripsi
<p>Kelas</p> 	Kelas pada struktur sistem
<p>Antarmuka/<i>Interface</i></p>  <p><b>nama_interface</b></p>	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek
<p>Asosiasi/<i>asociation</i></p> 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan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

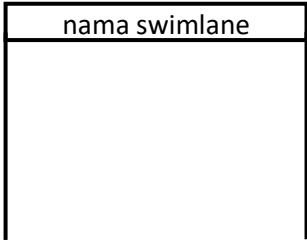
Sumber: (Rosa dan Shalahudin, 2018).

### C. *Activity Diagram*

Activity diagram atau Diagram aktivitas menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem. 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.

Simbol	Deskripsi
<p data-bbox="316 309 443 338"><i>Swimlane</i></p> 	<p data-bbox="777 309 1366 450">Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi</p>

**Sumber:** (Rosa dan Shalahudin, 2018).

### 2.2.6. Pengertian MySQL

MySQL merupakan RDBMS (atau server database) yang mengelola database dengan cepat menampung dalam jumlah sangat besar dan dapat di akses oleh banyak user, MySQL adalah sebuah *software open source* yang digunakan untuk membuat sebuah *database*, berdasarkan pendapat yang dikemukakan di atas dapat ditarik kesimpulan bahwa MySQL adalah suatu *software* atau program yang digunakan untuk membuat sebuah database yang bersifat *open source* (Firman et al, 2016)

### 2.2.7. Framework

Menurut (Hakim, 2018) *Framework* secara sederhana dapat diartikan kumpulan dari fungsi-fungsi/ prosedur-prosedur dan *class-class* untuk tujuan tertentu yang sudah siap digunakan sehingga bisa lebih mempermudah dan mempercepat pekerjaan seorang programmer, tanpa harus membuat fungsi atau class dari awal. Ada beberapa alasan mengapa menggunakan *Framework* :

1. Mempercepat dan mempermudah pembangunan sebuah aplikasi web.



2. Relatif memudahkan dalam proses *maintenance* karena sudah ada pola tertentu dalam sebuah *Framework* (dengan syarat programmer mengikuti pola standar yang ada).
3. Umumnya *Framework* menyediakan fasilitas-fasilitas yang umum dipakai sehingga tidak perlu membangun dari awal.
4. Lebih bebas dalam pengembangan jika dibandingkan dengan CMS.

### **2.2.8. CodeIgniter**

Menurut (Hakim, 2018) *CodeIgniter* adalah sebuah *framework* PHP yang dapat membantu mempercepat *developer* dalam pengembangan aplikasi *web* berbasis PHP dibanding jika menulis semua kode program dari awal. Adapun beberapa keuntungan menggunakan *CodeIgniter*, diantaranya:

1. Gratis

*CodeIgniter* berlisensi dibawah *Apache/BSD opensorce*.

2. Ditulis Menggunakan PHP 4

Meskipun *CodeIgniter* dapat berjalan di PHP 5, namun sampai saat ini kode program *CodeIgniter* masih dibuat dengan menggunakan PHP 4.

3. Berukuran Kecil

Ukuran *CodeIgniter* yang kecil merupakan keunggulan tersendiri. Dibanding dengan *framework* lain yang berukuran besar.

4. Menggunakan Konsep MVC

*CodeIgniter* menggunakan konsep MVC yang memungkinkan pemisahan *layer application-logic* dan *presentation*.

5. URL yang Sederhana

Secara *default*, URL yang dihasilkan *CodeIgniter* sangat bersih dan *Search Engine Friendly* (SEF).

6. Memiliki Paket *Library* yang Lengkap

*CodeIgniter* mempunyai *library* yang lengkap untuk mengerjakan operasi-operasi yang umum dibutuhkan oleh sebuah aplikasi berbasis web, misalnya mengakses *database*, mengirim email, memvalidasi form, menangani *session* dan sebagainya.

7. *Extensible*

Sistem dapat dikembangkan dengan mudah menggunakan *plugin* dan *helper*, atau dengan menggunakan *hooks*.

8. Tidak Memerlukan *Template Engine*

Meskipun *CodeIgniter* dilengkapi dengan *template* parser sederhana yang dapat digunakan, tetapi hal ini tidak mengharuskan kita untuk menggunakannya.

9. Dokumentasi Lengkap dan Jelas

Dari sekian banyak *framework*, *CodeIgniter* adalah satu-satunya *framework* dengan dokumentasi yang lengkap dan jelas.

### 2.2.9. Pengujian ISO 25010

ISO/IEC 25010 merupakan model kualitas sistem dan perangkat lunak yang menggantikan ISO/IEC 9126 (Iqbal, 2016). 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

### 3. Data model kualitas

Adapun dimensi yang pertama terdapat beberapa faktor elemen diantaranya :

- 1) *Functionality* (Fungsionalitas). Kemampuan perangkat lunak untuk menyediakan fungsionalitas yang dibutuhkan ketika perangkat lunak digunakan pada kondisi spesifik tertentu dalam hal ini perangkat lunak dapat memenuhi kelayakan dari sebuah fungsi untuk melakukan pekerjaan yang spesifik bagi pengguna dan dapat memberikan hasil yang tepat dan ketelitian terhadap tingkat kebutuhan pengguna.
- 2) *Reliability* Merupakan tingkatan dimana perangkat lunak dapat bertahan pada tingkatan tertentu ketika digunakan oleh pengguna pada kondisi yang spesifik dalam hal ini perangkat lunak dapat beroperasi dan siap ketika dibutuhkan untuk digunakan dan juga dapat bertahan pada tingkat kemampuan tertentu terhadap kegagalan, kesalahan serta perangkat lunak kembali pada tingkat tertentu dalam mengembalikan pengembalian data yang disebabkan kegagalan atau kesalahan pada perangkat lunak.
- 3) *Performance efficiency* Merupakan tingkatan dimana perangkat lunak dapat memberikan kinerja terhadap sejumlah sumber daya yang digunakan pada kondisi tertentu dalam hal ini *performance efficiency* dapat memberikan reaksi dan waktu yang dibutuhkan ketika melakukan aksi dari sebuah fungsi dan perangkat lunak dapat menggunakan sejumlah sumber daya ketika melakukan aksi dari sebuah fungsi.
- 4) *Usability* Perangkat lunak dapat dimengerti, dipelajari, digunakan, dan menarik pengguna ketika digunakan dalam hal ini perangkat lunak

mudah dipelajari oleh pengguna, perangkat lunak dapat digunakan dan dioperasikan oleh pengguna.

- 5) *Security* Merupakan perlindungan terhadap perangkat lunak dari berbagai ancaman atau keganjalan dalam hal ini perangkat lunak memiliki perlindungan terhadap data atau informasi dari pengguna dan merupakan dari kelengkapan, ketepatan dari sejumlah *asset* yang telah dijaga sehingga aksi atau tindakan yang dilakukan telah terbukti dan hal tersebut tidak dapat ditolak.
- 6) *Compability* Faktor ini merupakan kemampuan dari dua atau lebih komponen perangkat lunak dapat melakukan pertukaran informasi dan melakukan fungsi yang dibutuhkan ketika digunakan pada *hardware* atau lingkungan perangkat lunak yang sama.
- 7) *Maintainability* Merupakan tingkat dimana sebuah perangkat lunak dapat dimodifikasi. Dalam hal ini modifikasi adalah perbaikan, perubahan atau penyesuaian perangkat lunak untuk dapat berubah pada lingkungan , kebutuhan dan fungsionalitas yang spesifik. Selain itu perangkat lunak dapat dianalisis untuk mengetahui apa yang menyebabkan kegagalan pada perangkat lunak untuk mengidentifikasi bagian yang dapat dimodifikasi.
- 8) *Transferability*. Merupakan kemudahan dimana sistem atau komponen dapat berpindah dari lingkungan satu ke lingkungan yang lain dalam hal ini perangkat lunak dapat beradaptasi dengan cepat pada spesifikasi lingkungan yang berbeda tanpa menerapkan aksi atau cara lain dari pada memberikan tujuan tertentu terhadap perangkat lunak yang telah ada.

### 2.2.10. Skala Pengukuran

Skala pengukuran yang digunakan adalah skala Likert, skala yang didasarkan pada penjumlahan sikap responden dalam merespon pernyataan berkaitan indikator-indikator suatu konsep atau variable yang sedang diukur (Sugiyono, 2018). Skala Likert umumnya menggunakan lima titik dengan label netral pada posisi tengah (ketiga). Skala Likert dapat dilihat pada Tabel 2.5 :

**Tabel 2. 5** Skala Likert (Sugiyono, 2018)

Jawaban	Skor
Sangat Setuju	5
Setuju	4
Netral	3
Tidak Setuju	2
Sangat Tidak Setuju	1

Hasil penilaian responden akan dihitung *persentase* kelayakannya dengan menggunakan perhitungan, dapat dilihat dibawah ini:

$$\text{Persentase} = \frac{\text{Skor Aktual (f)}}{\text{Skor Ideal (n)}} \times 100\%$$

Persentase kelayakan yang diperoleh kemudian dibandingkan dengan Tabel konversi yang berpedoman pada acuan konversi nilai, dapat dilihat pada Tabel 2.6.

**Tabel 2. 6** Skala Konversi Nilai (Sugiyono, 2018)

Persentase Pencapaian (%)	Interpretasi
$90 \leq x$	Sangat Baik
$80 \leq x < 90$	Baik
$70 \leq x < 80$	Cukup
$60 \leq x < 70$	Kurang
$X < 60$	Sangat Kurang

Keterangan: x = persentase hasil pengujian