

BAB II LANDASAN TEORI

2.1. Tinjauan Pustaka

Dari tinjauan pustaka yang dilakukan terdapat penelitian sejenis yang pernah dilakukan sebelumnya, diantaranya dapat dilihat pada tabel 2.1 sebagai berikut :

Tabel 2.1 Penelitian Terdahulu

Literature	Penulis	Informasi publikasi	Judul
Literature 1	Utomo, Arifin and Agustin	Jurnal Sisfotek Global. ISSN : 2088 – 1762 Vol. 8 No. 1, Maret 2018	Perancangan Sistem Informasi Pencatatan Laporan Harian Komponen Rusak di PT Broco Mutiara Electrical Industry
Literature 2	Lis Suryadi	Prosiding SENTIA 2015 –Politeknik Negeri Malang, Volume 7 – ISSN: 2085-2347	Rancang Bangun Sistem Informasi Monitoring Pelaksanaan Pekerjaan Studi Kasus : Suku Dinas Pekerjaan Umum Tata Air Kota Administrasi Jakarta Selatan Dengan Metodologi Berorientasi Obyek
Literature 3	Sanjaya and Sumboro	Jurnal Ilmiah Go Infotech Volume 21 No. 1, Juni 2015, ISSN : 1693-590x	Rancang Bangun Aplikasi Penjadwalan Kuliah Stmik Aub Surakarta Berbasis Web
Literature 4	Juradin, Muis and Purnamawati	Jurnal Program Studi Pendidikan Teknologi dan Kejuruan, Program Pascasarjana, Universitas Negeri Makassar 2018	Sistem informasi monitoring praktek kerja industri berbasis web
Literature 5	Chusna and Fatrianto	Jurnal Jurusan Teknik Informatika, Fakultas Teknik, Universitas Negeri Surabaya 2017	Rancang Bangun Sistem Penjadwalan Guru Mengajar Berbasis Web (Studi Kasus : Smpn 2 Dawarblandong, Mojokerto)

Pembahasan dari masing-masing literature adalah sebagai berikut:

1. Literature 1

Pembuatan pencatatan laporan harian komponen rusak ini menjadi salah satu data penting bagi perusahaan karena bisa menjadi perbandingan untuk membuat kembali komponen agar mengurangi kerusakan pada hasil yang akan di lakukan produksi. Data yang bisa mendukung untuk menjadi sebuah laporan harian komponen rusak adalah sebuah buku khusus untuk mencatat komponen yang rusak per hari yang ditulis oleh salah satu karyawan dari masing-masing regu, yang hasilnya akan di data ulang dan diperiksa sesuai bahan komponen sehingga menjadi laporan yang baik dan benar. Hasil penelitian ini diketahui bahwa dengan adanya sistem informasi rancangan aplikasi pencatatan laporan harian komponen rusak yang berbasis desktop base, dapat menginput data sesuai dengan laporan hasil komponen rusak dari bagian produksi. Sistem informasi aplikasi pencatatan laporan harian ini, terdapat keterangan bagian karyawan yang belum tercatat hasilnya dilihat dari laporan data komponen, dan bagian karyawan yang sudah dan belum menginput data laporan harian komponen rusak. Sistem pembuatan laporan harian komponen rusak ini akan menyimpan data kedalam sistem yang sudah dikelola oleh user, untuk mengecek kecocokan laporan dan komponen rusak harus dicek list laporan dan mencocokkan dengan komponen yang ada.

2. Literature 2

Dengan adanya permasalahan yang terjadi di Suku Dinas Pekerjaan Umum Tata Air Kota Administrasi Jakarta Selatan, maka timbullah kesadaran untuk membangun sebuah sistem yang dapat mempermudah dalam pengolahan data,

mempercepat proses penyajian informasi yang akurat, mengurangi penumpukan arsip dan memudahkan pengelola dalam pencarian data, yaitu sebuah Sistem Informasi Monitoring Pelaksanaan Pekerjaan Pada Suku Dinas Pekerjaan Umum Tata Air Kota Administrasi Jakarta Selatan, metode pengembangan sistem dengan metodologi berorientasi obyek, aplikasi dibangun berbasis web dengan databases Mysql Server.

3. Literature 3

Permasalahan penjadwalan terkait erat dengan masalah optimasi. Oleh karena itu, pengembangan sistem penjadwalan kuliah dilakukan dengan melalui beberapa perbaikan untuk menghindari bentrokan jadwal. Kesimpulan dalam penelitian ini yaitu telah dirancang dan dibuat aplikasi penjadwalan kuliah STMIK AUB Surakarta menggunakan PHP dan Mysql. Aplikasi penjadwalan kuliah ini dapat menjadwalkan matakuliah sesuai dengan batasan-batasan yang ada pada STMIK AUB Surakarta. Masih terdapat beberapa kekurangan pada aplikasi penjadwalan kuliah ini yang diharapkan dapat disempurnakan oleh peneliti selanjutnya.

4. Literature 4

Hasil penelitian menunjukkan bahwa: (1) pengembangan sistem informasi monitoring prakerin terdiri dari empat tahap yaitu: analisis, desain, pengembangan/pengkodean dan implementasi/pengujian. Sistem ini memiliki empat pengguna yaitu admin/koordinator prakerin, guru pembimbing, siswa, dan pembimbing industri dengan fitur sesuai dengan hak akses yang dimiliki. Sistem ini memberikan informasi mengenai prakerin secara online. Kegiatan pencatatan data-data prakerin akan dilakukan oleh administrasi prakerin yang nantinya data

tersebut akan dikirim dan disimpan pada web; (2) sistem informasi monitoring prakerin berbasis web berada pada kategori sangat valid, sehingga layak digunakan untuk uji coba lapangan terhadap pengguna, kepraktisan dan keefektifan sistem informasi monitoring prakerin berbasis web pada saat uji coba lapangan berada pada kategori sangat praktis dan sangat efektif sehingga layak digunakan dan memberikan manfaat bagi pengguna untuk pengelolaan serta pelaksanaan prakerin di SMKN 4 Makassar.

5. Literature 5

SMP Negeri 2 Dawarblandong merupakan jenjang pendidikan dasar pada pendidikan formal di Indonesia setelah lulus Sekolah Dasar. Pada SMP Negeri 2 Dawarblandong, penjadwalan guru mengajar menjadi permasalahan yang sangat rumit apabila hanya ditangani menggunakan komputer saja dan masih menggunakan microsoft excel dan banyak keluhan yang di alami. Ketika ingin menyusun jadwal guru mengajar dan terjadi bentrokan data atau kesamaan jam, waktu, dan kelas solusi yang saat ini berjalan masih secara manual dengan cara memberi warna pada jadwal guru yang bentrokan data atau kesamaan jadwal.. Untuk menyusun jadwal yang baik, pembuat jadwal perlu memperhatikan bahwa jadwal yang di buat tidak ada bentrokan atau kesamaan jam antar guru satu dengan guru yang lain dalam satu waktu dan kelas tertentu, atau satu guru yang berada di lebih dari satu kelas pada satu waktu tertentu. Begitu pula dengan bentuk penginformasian jadwal masih berjalan secara manual dengan menggunakan media kertas yang di tempel di madding ruang guru.

2.2. Sistem Informasi

Menurut (Al Fatta, 2007), “Sistem adalah suatu kumpulan atau himpunan dari unsur atau variabel yang saling terorganisasi, saling berinteraksi, dan saling bergantung satu sama lain”.

Dari pengertian di atas, dapat disimpulkan bahwa sistem adalah sekumpulan dari elemen, unsur, atau variable yang saling terorganisasi dan bekerja sama untuk mencapai suatu tujuan.

Unsur-unsur yang mewakili suatu sistem secara umum adalah masukan (*input*), proses (*process*), dan keluaran (*output*). Selain itu, sebuah sistem memiliki karakteristik atau sifat-sifat tertentu yang mencirikan bahwa hal tersebut dapat dikatakan sebagai sebuah sistem. Menurut Jogiyanto (2005), suatu sistem mempunyai karakteristik tertentu, antara lain :

1. Komponen sistem (*Components*)

Suatu sistem terdiri dari komponen yang saling berinteraksi yang saling berkaitan dan bekerja sama membentuk suatu kesatuan. Komponen-komponen sistem dapat berupa suatu subsistem atau bagian bagian dari sistem.

2. Batasan sistem (*Boundary*)

Batasan sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem lainya atau dengan lingkungan luarnya.

3. Lingkungan luar sistem (*Environtment*)

Lingkungan luar sistem apapun di luar batasan sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem yang dapat bersifat menguntungkan dan juga merugikan.

4. Penghubung sistem (*Interface*)

Penghubung sistem merupakan media yang menghubungkan antara satu subsistem dengan subsistem lainnya. Melalui penghubung ini memungkinkan sumber daya mengalir dari satu subsistem ke subsistem lainnya. Dengan melalui penghubung, suatu subsistem dapat berinteraksi dengan subsistem lainnya untuk membentuk suatu kesatuan.

5. Masukan sistem (*Input*)

Masukan sistem adalah energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan dan masukan sinyal. Masukan perawatan adalah energi yang dimasukkan supaya sistem tersebut dapat berjalan. Masukan sinyal adalah energi yang diproses untuk mendapatkan keluaran dari sistem.

6. Keluaran sistem (*Output*)

Keluaran sistem adalah energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran dapat merupakan masukan untuk subsistem lainnya.

7. Pengolahan sistem (*Process system*)

Suatu sistem dapat mempunyai suatu bagian pengolah yang akan merubah masukan menjadi keluaran.

8. Sasaran sistem (*Goal system*)

Suatu sistem mempunyai tujuan atau sasaran, jika sistem tidak mempunyai sasaran maka operasi sistem tidak akan ada gunanya.

2.3. Pengertian Informasi

Menurut (Sutabri, 2012) “Informasi adalah data yang telah diproses kedalam suatu bentuk yang mempunyai arti bagi sipenerima dan mempunyai nilai nyata dan terasa bagi keputusan saat itu atau keputusan mendatang”. Sedangkan (Sutarman, 2012) menyatakan bahwa, “Informasi merupakan sekumpulan fakta (data) yang diorganisasikan dengan cara tertentu sehingga mempunyai arti bagi si penerima”.

Dari pengertian di atas dapat disimpulkan bahwa informasi adalah fakta (data) yang telah diproses menjadi bentuk yang lebih berharga bagi si penerima. Fungsi utama informasi adalah menambah pengetahuan atau mengurangi ketidakpastian pemakai informasi.

Nilai dari informasi ditentukan oleh manfaat dan biaya untuk mendapatkannya. Menurut Sutabri (2012), kualitas dari informasi tergantung dari tiga hal, yaitu :

1. Akurat (*Accurate*)

Informasi harus mencerminkan keadaan yang sebenarnya, artinya informasi harus bebas dari kesalahan-kesalahan dan tidak bisa menyesatkan. Akurat juga berarti bahwa informasi harus jelas mencerminkan maksudnya.

2. Tepat Waktu (*Timelines*)

Informasi yang sampai pada penerima tidak boleh terlambat, informasi yang sudah usang tidak akan mempunyai nilai lagi.

3. Relevan (*Relevance*)

Relevan berarti informasi tersebut mempunyai manfaat untuk pemakainya. Relevansi informasi untuk setiap orang berbeda-beda.

2.4. Pengertian Sistem Informasi

Sistem informasi merupakan susunan yang disusun secara sistematis dan teratur dari jaringan-jaringan aliran informasi yang menghubungkan setiap bagian dari suatu sistem, sehingga memungkinkan diadakannya komunikasi antar bagian atau suatu fungsional. Sementara pengertian umum mengenai sistem adalah bagian-bagian dari suatu rangkaian kegiatan yang saling berkaitan dan beroperasi secara bersamaan dalam suatu prosedur atau cara kerja tertentu untuk mencapai suatu tujuan.

Jogiyanto (2014) menjelaskan bahwa: “Sistem informasi adalah suatu sistem di dalam suatu kelompok yang mempertemukan kebutuhan pengelolaan transaksi yang dilakukan secara harian, bersifat manajerial dan kegiatan strategi dari suatu kelompok dan menyediakan laporan-laporan terkait untuk pihak eksternal tertentu”.

Sutabri (2012) berpendapat bahwa, “Sistem informasi yaitu suatu sistem yang dibuat oleh manusia yang terdiri dari komponen-komponen dalam organisasi untuk menyajikan informasi yang dibutuhkan”.

Dari pengertian diatas dapat disimpulkan bahwa sistem informasi adalah suatu sistem yang dibuat oleh manusia yang terdiri dari komponen-komponen dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang bersifat manajerial yang bertujuan untuk menyajikan informasi dan laporan-laporan yang diperlukan.

2.4.1. Kegiatan Sistem Informasi

Kegiatan dari sistem informasi mencakup hal-hal sebagai berikut:

1. *Input*, menggambarkan suatu kegiatan untuk menyediakan data untuk di proses.
2. *Process*, menggambarkan bagaimana suatu data di proses untuk menghasilkan informasi yang bernilai tambah.
3. *Output*, suatu kegiatan untuk menghasilkan suatu informasi dari data tersebut.
4. Penyimpanan, suatu kegiatan untuk memelihara dan menyimpan data.
5. Kontrol, suatu aktivitas untuk menjamin bahwa sistem informasi tersebut berjalan sesuai dengan yang diharapkan.

Dari pengertian di atas dapat disimpulkan bahwa *framework* merupakan kerangka kerja yang di dalamnya terdapat kode program dan fungsi dasar untuk melakukan tugas tertentu yang memudahkan *programmer* untuk membuat aplikasi dan dapat digunakan kembali untuk aplikasi lain yang sejenis.

2.5. Framework

Menurut (Rosa and Shalahudin, 2018), *framework* merupakan kerangka kerja yang memudahkan *programmer* untuk membuat sebuah aplikasi sehingga *programmer* akan lebih mudah melakukan perubahan (*customize*) terhadap aplikasinya dan dapat memakainya kembali untuk aplikasi lain yang sejenis. Menurut (Yuhefizar, 2014) mengatakan bahwa “*Framework* adalah kerangka kerja atau sekumpulan file-file yang sudah ter-*include*, yang mana di dalam file

tersebut terdapat perintah kode program dan fungsi dasar untuk melakukan tugas tertentu”.

Dari pengertian di atas dapat disimpulkan bahwa *framework* merupakan kerangka kerja yang di dalamnya terdapat kode program dan fungsi dasar untuk melakukan tugas tertentu yang memudahkan *programmer* untuk membuat aplikasi dan dapat digunakan kembali untuk aplikasi lain yang sejenis.

2.6. Codeigniter

Menurut (Yuhefizar, 2014) mengatakan bahwa “*Codeigniter* adalah sebuah *framework* yang dibuat menggunakan bahasa pemrograman *PHP* yang bertujuan untuk memudahkan para programmer web untuk membuat atau mengembangkan aplikasi berbasis web”. *Codeigniter* memiliki eksekusi tercepat dibandingkan dengan *framework* lainnya. *Codeigniter* bersifat open source dan menggunakan model basis *MVC (Model View Controller)*, yang merupakan model konsep modern saat ini. *Codeigniter* juga menawarkan banyak library yang dapat digunakan (Saputra, 2011)

Menurut (Hidayatullah and Kawistara, 2017) metode *MVC* terdapat tiga komponen, yaitu :

1. *Model*, mengelola basis data (RDBMS) seperti *MySQL* ataupun *Oracle* RDMS. *Model* berhubungan dengan *database* sehingga biasanya dalam *model* akan berisi *class* ataupun fungsi untuk membuat (*create*), melakukan pembaruan (*update*), menghapus data (*delete*), mencari data (*search*), dan mengambil data (*select*) pada *database*. Selain itu juga *model* akan berhubungan dengan perintah-perintah *query* sebagai tindak

lanjut dari fungsi-fungsi (*create, update, delete, select*).

2. *View*, bagian *User Interface* atau bagian yang nantinya merupakan tampilan untuk *end-user*. *View* bisa berupa halaman *HTML, CSS, Javascript, JQuery* dan *AJAX*, karena metode yang dipakai merupakan *MVC* sehingga *view* tidak boleh terdapat pemrosesan data ataupun pengaksesan yang berhubungan dengan *database*, sehingga *view* hanya menampilkan data-data hasil dari *model* dan *controller*.
3. *Controller*, penghubung antara *view* dan *model*, maksudnya ialah karena *model* tidak dapat berhubungan langsung dengan *view* ataupun sebaliknya, jadi *controller* inilah yang digunakan sebagai jembatan keduanya. Sehingga tugas *controller* ialah sebagai pemrosesan data atau alur logic program, menyediakan *variable* yang akan ditampilkan di *view*, pemanggilan *model* sehingga *model* dapat mengakses *database*, *error handling*, validasi atau *check* terhadap suatu input data.

2.7. Basis Data

Himpunan kelompok data yang saling berhubungan yang diorganisasikan sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah merupakan definisi dari Basis Data. Secara lebih lengkap pemanfaatan basis data dilakukan untuk memenuhi tujuan Kecepatan, kemudahan, efisiensi ruang penyimpanan, keakuratan, ketersediaan, kelengkapan, keamanan dan pemakaian bersama (Hidayatullah dan Kawistara, 2017).

Sedangkan menurut (Rosa and Shalahudin, 2018), “Basis Data adalah media untuk menyimpan data agar dapat diakses dengan mudah dan cepat”. Jadi dapat disimpulkan basis data adalah kumpulan data yang saling berhubungan agar dapat

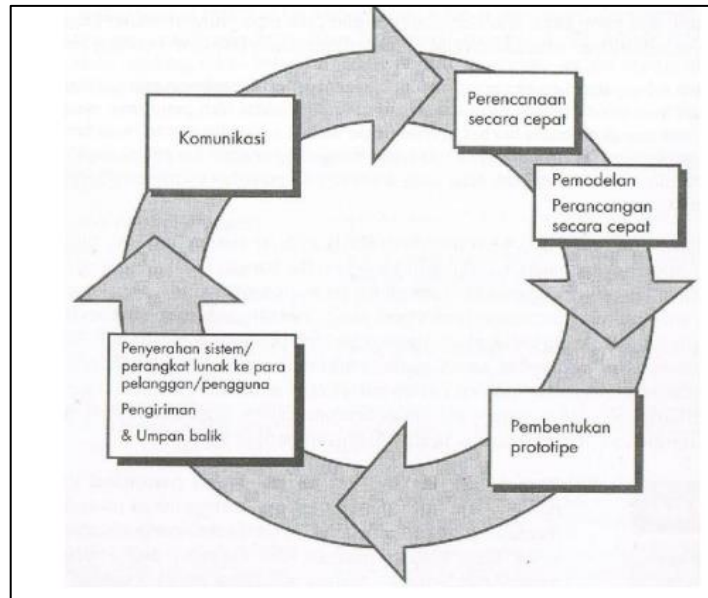
diakses oleh software tertentu.

2.7.1. Pengertian MySQL

Menurut (Nugroho, 2013) “*MySQL* adalah suatu sistem basis data dan relation atau *Relational Database Management System (RDBMS)* yang mampu bekerja secara cepat dan mudah digunakan. *MySQL* juga program pengakses database yang bersifat jaringan, sehingga dapat digunakan untuk aplikasi multi user (banyak pengguna). *MySQL* didistribusikan gratis di bawah lisensi GPL (*General Public Licence*). Dimana setiap orang bebas menggunakan *MySQL*, namun tidak boleh dijadikan produk yang dijadikan *closed source* atau komersial.”

2.8. Prototype

Menurut (Pressman & Roger, 2012) *prototype* mengatakan bahwa “Dalam melakukan perancangan sistem yang akan dikembangkan dapat menggunakan metode *prototype*. Metode ini cocok digunakan untuk mengembangkan sebuah perangkat yang akan dikembangkan kembali. Metode ini dimulai dengan pengumpulan kebutuhan pengguna, dalam hal ini pengguna dari perangkat yang dikembangkan adalah peserta didik. Kemudian membuat sebuah rancangan kilat yang selanjutnya akan dievaluasi”. Metode pengembangan sistem yaitu *prototype* dapat dilihat pada gambar 2.1 :



Gambar 2. 1 Metode Pengembang Sistem

Prototype bukanlah merupakan sesuatu yang lengkap, tetapi sesuatu yang harus dievaluasi dan dimodifikasi kembali. Segala perubahan dapat terjadi pada saat prototype dibuat untuk memenuhi kebutuhan pengguna dan pada saat yang sama memungkinkan pengembang untuk lebih memahami kebutuhan pengguna secara lebih baik. Berikut adalah tahapan dalam metode *prototype* :

- A. Komunikasi dan pengumpulan data awal, yaitu analisis terhadap kebutuhan pengguna.
2. Perencanaan yaitu rencana kegiatan-kegiatan sebelum dilakukanya pemodelan dan pembentukan sistem.
3. Pemodelan digunakan untuk menggambarkan suatu sistem yang akan dikembangkan dengan menggunakan UML dan Interface.
4. Pembentukan yaitu proses pengkodean untuk membentuk dari gambaran pemodelan yang dirancang.

5. Penyerahan dan umpan balik yaitu hasil pengujian yang telah dilakukan jika layak lalu dilakukan penyerahan ke *client* dan dilakukan tahap evaluasi dari hasil umpan balik saat penerapan.

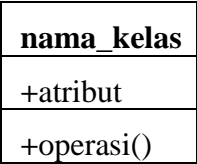


2.9. UML (*Unified Modelling Language*)


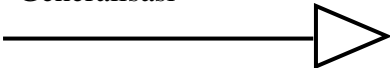

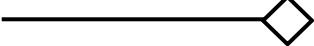
UML (*Unified Modeling Language*) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis & desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek Rosa dan Shalahuddin (2018).

1. *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 2.2 di bawah ini:

Tabel 2.2 Simbol *Class Diagram*

Simbol	Deskripsi
Kelas 	Kelas pada struktur sistem
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>)

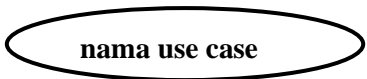
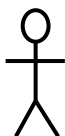
2. Use Case Diagram





Menurut Rosa dan Shalahuddin (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 2.3 di bawah ini:

Tabel 2.3 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 dinyatakan dengan menggunakan kata kerja di awal frase nama use case
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




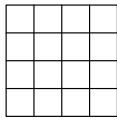


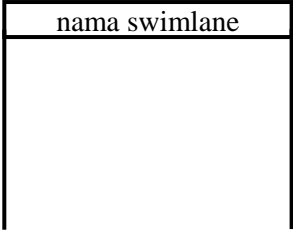
	merupakan orang, biasanya dinyatakan menggunakan kata benda di awal frase
Asosiasi/ <i>association</i> 	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor
Ekstensi/ <i>extend</i>	Relasi <i>use case</i> tambahan ke 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 atau sebagai syarat dijalankan <i>use case</i> ini

3. 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.
<i>Swimlane</i> 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

Sumber: Rosa dan Shalahuddin (2018)

2.10. Pengertian Analisis

Menurut (Whitten, Bentley and Ditman, 2016) analisis sistem adalah teknik pemecahan masalah yang diuraikan suatu sistem menjadi bagian-bagian komponen yang bertujuan untuk mengetahui seberapa baik bagian-bagian komponen tersebut bekerja dan berinteraksi untuk mencapai tujuannya.

2.11. Analisis *PIECES*

Untuk mengidentifikasi masalah, maka harus dilakukan analisis terhadap kinerja, ekonomi, pengendalian, efisiensi, dan pelayanan atau juga sering disebut dengan analisis *pieces*. Adapun pengertian dari analisis *pieces* sebagai berikut (Al-Fatta, 2012)

1. Analisis Kinerja Sistem (*Performnace*)

Kinerja adalah suatu kemampuan sistem dalam menyediakan tugas dengan cepat sehingga sasaran dapat segera tercapai. Kinerja diukur dengan jumlah produksi (*troughput*) dan waktu tanggap (*response time*).

2. Analisis Informasi (*Information*)

Informasi adalah hal yang paling penting bagi pihak manajemen dalam mengambil suatu keputusan dan merencanakan langkah selanjutnya untuk kesuksesan usaha yang dijalankan.

3. Analisis Ekonomi (*Economy*)

Analisis ekonomi merupakan suatu motivasi yang paling umum bagi suatu proyek dan berkaitan dengan jumlah sumber daya yang digunakan. Analisis ekonomi digunakan untuk menganalisis biaya yang diperlukan dalam mengembangkan sistem serta menganalisis manfaat secara ekonomi jika sistem sudah digunakan.

4. Analisis Pengendalian (*Control*)

Analisis ini digunakan untuk untuk meningkatkan kinerja sistem, mencegah atau mendeteksi kesalahan sistem.

5. Analisis Efisiensi (*Efficiency*)

Efisiensi berhubungan dengan bagaimana sumber daya yang ada dapat

digunakan secara optimal.

6. Analisis Pelayanan (*Service*)

Peningkatan pelayanan memperlihatkan kategori yang beragam. Proyek yang dipilih merupakan peningkatan pelayanan yang lebih baik bagi manajemen, user dan bagian lain yang merupakan simbol kualitas dari sistem informasi.


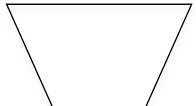

2.12. Bagan Alir (*Flowchart*)


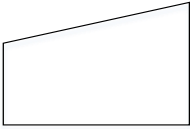
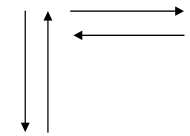


Menurut (Jogiyanto, 2014) mendefinisikan bahwa “Bagan alir (*flowchart*) adalah bagan (*chart*) yang menunjukkan alir (*flow*) di dalam program atau prosedur sistem secara logika”. Bagan alir digunakan terutama untuk alat bantu komunikasi dan untuk dokumentasi.

2.12.1. Bagan Alir Dokumen (*Document Flowchart*)

Menurut (Jogiyanto, 2014) “Bagan alir dokumen merupakan bagan alir yang menunjukkan arus dari laporan dan formulir termasuk tembusan – tembusannya. Bagan alir dokumen ini menggunakan simbol – simbol yang sama dengan yang digunakan di dalam bagan alir sistem”. Simbol-simbol bagan alir dokumen dapat dilihat pada tabel 2.5 sebagai berikut:

Tabel 2.5 Simbol Bagan Alir Dokumen

No	Simbol	Nama	Keterangan
1		Dokumen	Menunjukkan dokumen input/output baik untuk proses manual, mekanik atau komputer.
2		Kegiatan manual	Menunjukkan pekerjaan manual
3		Simpanan <i>offline</i>	File non komputer yang diarsip

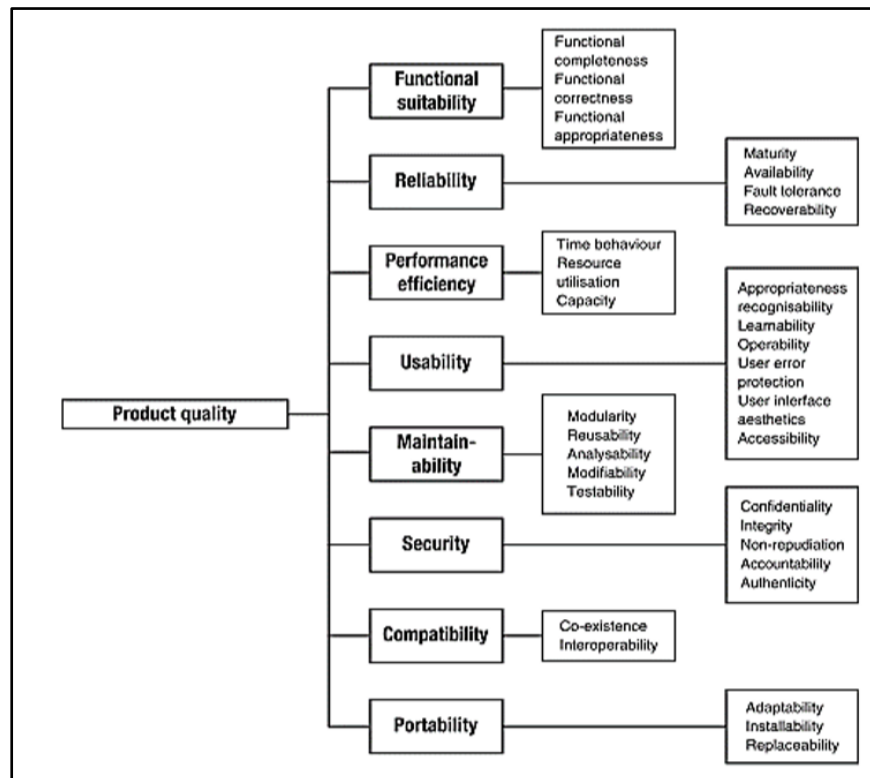
4		Proses	Menunjukkan kegiatan proses dari operasi program komputer
5		Keyboard	Menunjukkan input yang menggunakan on-line keyboard
6		Garis Alir	Menunjukkan arus dari proses
7		Penghubung	Menunjukkan penghubung ke halaman yang masih sama atau ke halaman lain
8		Titik Terminal	Menunjukkan awal dan akhir dari suatu proses

(Sumber : Jogiyanto, Analisis dan Desain Sistem)

2.13. Pengujian ISO 25010

ISO/IEC 25010 merupakan model kualitas sistem dan perangkat lunak yang menggantikan ISO/IEC 9126 tentang software engineering (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



Gambar 2. 2 Model kualitas produk ISO/IEC 25010

Adapun dimensi yang pertama terdapat beberapa faktor elemen diantaranya :

- 1) *Functionality* (Fungsionalitas). Kemampuan perangkat lunak untuk Merupakan tingkatan dimana perangkat lunak dapat 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. Karakteristik ini dibagi menjadi beberapa karakteristik yaitu.

- a. *Functional completeness*, sejauh mana fungsi yang disediakan mencakup semua tugas dan tujuan pengguna secara spesifik.

- b. *Functional correctness*, sejauh mana produk atau sistem menyediakan hasil yang benar sesuai kebutuhan.
- c. *Functional appropriateness*, sejauh mana fungsi yang disediakan mampu memfasilitasi penyelesaian tugas dan tujuan tertentu.

2) *Compatibility*

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

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

3) *Usability*

Sejauh mana sebuah produk atau sistem dapat digunakan oleh user tertentu untuk mencapai tujuan dengan efektif, efisiensi, dan kepuasan tertentu dalam konteks penggunaan. Karakteristik ini terbagi menjadi beberapa karakteristik yaitu.

- a. *Appropriateness recognizability*, sejauh mana pengguna dapat mengetahui apakah sistem atau produk sesuai kebutuhan mereka.
 - b. *Learnability*, sejauh mana produk atau sistem dapat digunakan oleh pengguna untuk mencapai tujuan tertentu yang belajar menggunakan sistem atau produk dengan efisien, efektif, kebebasan dari resiko dan kepuasan dalam konteks tertentu.
 - c. *Operability*, sejauh mana produk atau sistem mudah dioperasikan dan dikontrol.
 - d. *User error protection*, sejauh mana produk atau sistem melindungi pengguna terhadap membuat kesalahan.
 - e. *User interface aesthetics*, sejauh mana antarmuka pengguna dari produk atau sistem memungkinkan interaksi yang menyenangkan dan memuaskan pengguna.
 - f. *Accessibility*, sejauh mana produk atau sistem dapat digunakan oleh semua kalangan untuk mencapai tujuan tertentu sesuai konteks penggunaan.
- 4) *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. Karakteristik ini terbagi menjadi beberapa subkarakteristik yaitu.

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

5) *Security*

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

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

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

6) *Portability*

Sejauh mana keefektifan dan efisiensi sebuah sistem, produk atau komponen dapat dipindahkan dari satu perangkat keras, perangkat lunak atau digunakan pada lingkungan yang berbeda. Karakteristik ini dibagi menjadi beberapa karakteristik yaitu.

- a. *Adaptability*, sejauh mana produk atau sistem dapat secara efektif dan efisien disesuaikan pada perangkat lunak, perangkat keras dan lingkungan yang berbeda.
- b. *Installability*, sejauh mana produk atau sistem dapat berhasil dipasang atau dihapus dalam lingkungan tertentu.
- c. *Replaceability*, sejauh mana produk atau sistem dapat menggantikan produk atau sistem lain yang ditentukan untuk tujuan yang sama pada lingkungan yang sama.

7) *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. Kinerja relatif terhadap sumber daya yang digunakan dalam kondisi tertentu. Karakteristik ini terbagi menjadi beberapa subkarakteristik yaitu.

- a. *Time behaviour*, sejauh mana respon dan pengolahan waktu produk atau sistem dapat memenuhi persyaratan ketika menjalankan fungsi.
 - b. *Resource utilization*, sejauh mana jumlah dan jenis sumber daya yang digunakan oleh produk atau sistem dapat memenuhi persyaratan ketika menjalankan fungsi.
 - c. *Capacity*, sejauh mana batas maksimum parameter produk atau sistem dapat memenuhi persyaratan.
- 8) *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. Sejalan mana keefektifan dan efisiensi dari sebuah produk atau sistem dapat dirawat. Karakteristik ini terbagi menjadi beberapa subkarakteristik yaitu.

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

2.13.1 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, 2017) Skala Likert umumnya menggunakan lima titik dengan label netral pada posisi tengah (ketiga).Skala Likert apat dilihat pada Tabel 2.6.

Tabel 2.6 Skala Likert

Jawaban	Skor
Sangat Setuju	5
Setuju	4
Netral	3

Tidak Setuju	2
Sangat Tidak Setuju	1

Sumber : (Sugiyono, 2017)

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 2.7.

Tabel 2.7 Skala Konversi Nilai

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

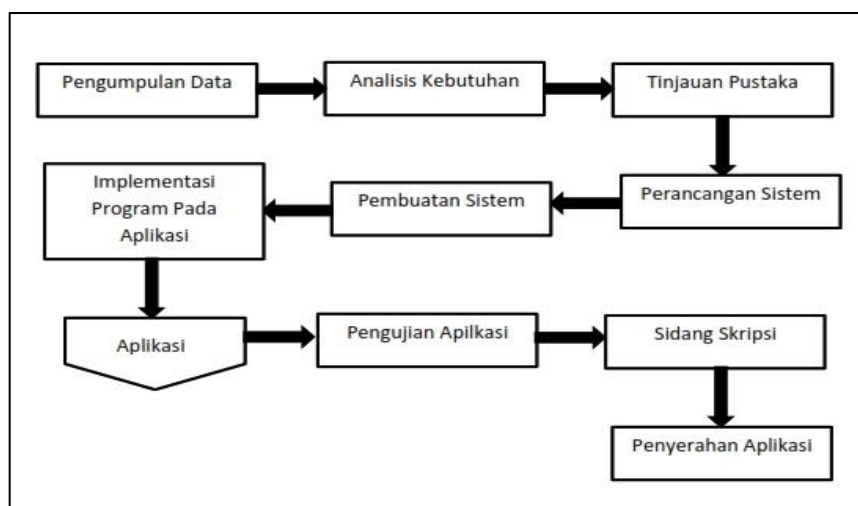
Sumber : (Sugiyono, 2017)

Keterangan : x = persentase hasil pengujian.

2.14. *State of The Art*

State of the Art adalah suatu gambaran yang fokus pada apa yang terbaru dari sebuah teori yang ada. *State of the Art* merupakan sebuah hasil sampai pencapaian tertinggi dalam suatu proses masalah yang akan diselesaikan, biasanya itu merupakan hasil dari penggunaan metode terbaru, alat yang terbaru, atau sebuah metode yang paling canggih yang dapat menyelesaikan sebuah masalah yang dihadapi dalam sebuah penelitian. Dalam penelitian ini gambaran *state of the Art* dimulai dari melakukan pengumpulan data dan tinjauan pustaka yang digunakan, selanjutnya penelitian menganalisis kebutuhan sistem yang

digunakan dalam penelitian dan merancang sistem. Setelah melakukan perancangan sistem dibuatlah coding atau proses tahapan implementasi, dan melakukan uji program yang nantinya akan diimplementasikan. Gambar State of the Art yang akan dibangun dalam penelitian ini dapat dilihat pada gambar dibawah ini :

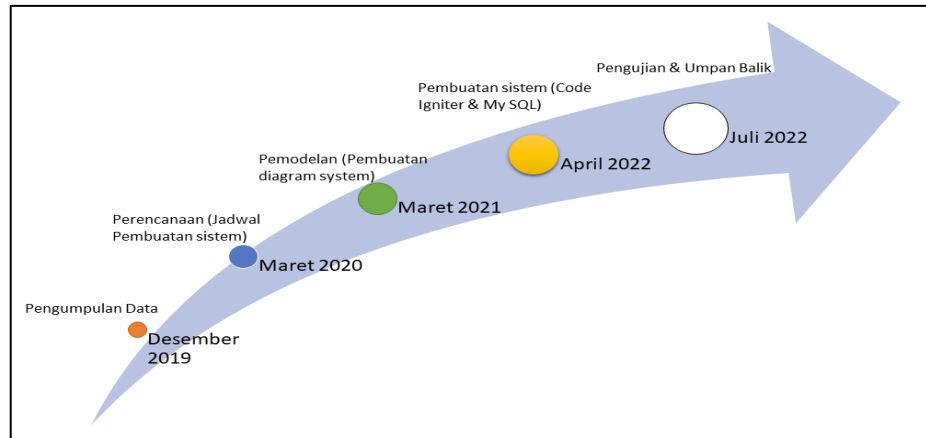


Gambar 2. 3 *State Of The Art* Penelitian

2.15. Road Map

Road Map adalah dokumen strategi pemandu dan perencanaan untuk melaksanakan strategi yang ditampilkan secara ringkas dengan visual tinggi yang memetakan apa saja arah yang akan dilakukan.

Hasil penggambaran *Road Map* dalam penelitian ini dimulai dari melakukan pengumpulan data, selanjutnya menganalisis kebutuhan sistem yang akan dibangun menggunakan *framework* codeigniter dan mysql, setelah itu akan dilakukan pengujian sistem menggunakan black box untuk menerima umpan-balik dari pihak perusahaan dan tahapan terakhir melakukan penyerahan sistem. Gambar *Road Map* dapat dilihat dibawah ini :



Gambar 2. 4 *Road Map* Penelitian