



BAB II

TINJAUAN PUSTAKA

2.1. Teori Judul

2.1.1 Sistem

Menurut Rusmawan (2019:29), “Sistem adalah suatu jaringan atau komponen-komponen yang membentuk satu kesatuan yang terkumpul untuk menyelesaikan suatu tujuan tertentu”.

Menurut Marani dan Hatta (2018:30), “Sistem adalah sekelompok elemen yang terintegrasi dengan maksud yang sama dengan untuk mencapai suatu tujuan”.

Berdasarkan pengertian tersebut, dapat disimpulkan bahwa sistem adalah komponen-komponen yang saling berhubungan dan bekerja bersama-sama untuk mencapai tujuan tertentu.

2.1.2 Informasi

Menurut Rusmawan (2019:32), “Informasi adalah sekumpulan data yang diolah sehingga berguna bagi seseorang atau organisasi yang membutuhkan informasi tersebut”.

Menurut Sitohang (2018:6), “Informasi adalah data yang telah diolah menjadi bentuk yang lebih berguna bagi yang menerima”.

Berdasarkan pengertian tersebut, dapat disimpulkan bahwa informasi merupakan sekumpulan data yang diolah sehingga dapat digunakan bagi pihak yang membutuhkan informasi tersebut.

2.1.3 Pengelolaan

Menurut Widiasworo (2018:11), “Pengertian Pengelolaan atau manajemen pada umumnya mengacu pada kegiatan-kegiatan yang meliputi perencanaan, pengorganisasian, pengarahan, pengoordinasian, pengawasan, dan penilaian”.

Menurut Kamus Besar Bahasa Indonesia, “Pengelolaan adalah proses, cara, perbuatan mengelola, dan/atau proses melakukan kegiatan tertentu dengan



menggerakkan tenaga orang lain, dan/atau proses melakukan yang membantu merumuskan kebijaksanaan dan tujuan organisasi, dan/atau proses yang memberikan pengawasan pada semua hal yang terlibat dalam pelaksanaan kebijaksanaan dan pencapaian tujuan”.

Berdasarkan pengertian tersebut, dapat disimpulkan bahwa pengelolaan adalah proses, cara, perbuatan mengelola pada semua hal yang terlibat dalam pelaksanaan kebijaksanaan dan pencapaian tujuan.

2.1.4 Kendaraan Operasional

Menurut Hoetoro dalam pane dan kurniawan (2020:9), “Kendaraan Operasional adalah kendaraan milik pemerintah yang digunakan untuk mendukung pelaksanaan tugas dan fungsi Operasional”.

Menurut Juhari dan hardianto (2017:81), “Kendaraan Operasional adalah salah satu pendukung pelaksanaan pelayanan yang mana kendaraan itu dioperasikan dalam melaksanakan tugasnya”.

Berdasarkan pengertian tersebut, dapat disimpulkan bahwa kendaraan operasional adalah kendaraan yang digunakan untuk pelayanan dan pelaksanaan tugas.

2.1.5 Metode Prototype

Menurut Andikos dan Gusteti dalam Michael dan Gustina (2019:61), “*Prototype* merupakan suatu metode dalam pengembangan sistem yang menggunakan pendekatan untuk membuat suatu program dengan cepat dan bertahap sehingga segera dapat dievaluasi oleh pemakai. *Prototype* mewakili model produk yang akan dibangun atau mensimulasikan struktur, fungsional, dan operasi sistem”.

Menurut Darmawan dalam kusumah dan permana (2018:112-113), “*Prototype* adalah suatu versi dari sebuah sistem potensial yang memberikan ide bagi para pengembang dan calon pengguna, bagaimana sistem akan berfungsi dalam bentuk yang telah selesai”.



Berdasarkan pernyataan tersebut, dapat disimpulkan bahwa Metode *prototype* adalah metode dalam pengembangan system yang didasarkan pada konsep model kerja sehingga dapat digunakan dengan baik. Prototype juga bisa dibangun melalui beberapa tool pengembangan untuk menyederhanakan proses. Adapun tahapan-tahapan dalam metode prototype adalah sebagai berikut :

1. Analisis Kebutuhan Sistem
2. Desain Sistem
3. Pengujian Sistem
4. Implementasi

Kelebihan :

1. Prototype melibatkan user dalam analisa dan desain.
2. Punya kemampuan menangkap requirement secara konkret.
3. Digunakan untuk memperluas SDLC.

Kekurangan :

1. Proses analisis dan perancangan terlalu singkat.
2. Mengesampingkan alternatif pemecahan masalah.
3. Biasanya kurang fleksible dalam menghadapi perubahan.
4. Prototype yang dihasilkan tidak selamanya mudah dirubah dan cepat selesai.

2.1.6 Asuransi

Menurut Kamus Besar Bahasa Indonesia, “pertanggungan (perjanjian antara dua pihak, pihak yang satu berkewajiban membayar iuran dan pihak yang lain berkewajiban memberikan jaminan sepenuhnya kepada pembayar iuran apabila terjadi sesuatu yang menimpa pihak pertama atau barang miliknya sesuai dengan perjanjian yang dibuat)”.

Menurut Guntara (2016:30), “Asuransi atau Pertanggungan adalah Perjanjian dengan mana penanggung mengikatkan diri kepada tertanggung dengan menerima premi untuk memberikan penggantian kepadanya karena kerugian, kerusakan atau kehilangan keuntungan yang diharapkan yang mungkin dideritanya akibat dari suatu evenemen (peristiwa tidak pasti)”.



Berdasarkan pernyataan tersebut, dapat disimpulkan bahwa asuransi adalah adalah pertanggungungan atau perjanjian antara dua belah pihak, di mana pihak satu berkewajiban membayar iuran/kontribusi/premi.

2.1.7 Sistem Informasi Pengelolaan Kendaraan Operasional Pada PT. Asuransi Bangun Askrida Cabang Palembang Menggunakan Metode *Prototype*

Sistem Informasi Pengelolaan Kendaraan Operasional Pada PT. Asuransi Bangun Askrida Cabang Palembang Menggunakan Metode *Prototype* merupakan suatu sistem informasi berbasis web yang menggunakan bahasa pemrograman php digunakan untuk mengelola kendaraan operasional.

2.2. Teori Khusus

2.2.1 Kamus Data

Menurut Jogiyanto dalam Purwati dan Aziz (2019:20), “Kamus data adalah katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi”.

Menurut Rusmawan (2019:36), “Kamus data merupakan katalog fakta data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi. Kamus data dibuat berdasarkan data yang ada pada diagram alur data”.

Berdasarkan pernyataan tersebut, dapat disimpulkan bahwa kamus data adalah katalog fakta dan kebutuhan-kebutuhan informasi dari suatu sistem informasi.

Simbol-simbol yang digunakan dalam kamus data sebagai berikut:

Tabel 2.1 Simbol-simbol Kamus Data

No.	Simbol	Uraian
1.	=	Terdiri atas, mendefinisikan, diuraikan menjadi, artinya
2.	+	Dan
3.	()	Optional (Boleh atau boleh tidak)
4.	{ }	Pengulangan

**Lanjutan Tabel 2.1** Simbol-simbol Kamus Data

No.	Simbol	Uraian
5.	[]	Memilih salah satu dari jumlah alternative, seleksi.
6.	* *	Komentar
7.	@	Identifikasi atribut
8.	!	Pemisah sejumlah alternatif pilihan antara simbol []

(Sumber : Supardi, 2015:7-8)

2.2.2 *Unified Modeling Language (UML)*

Menurut Mubarak (2019:20), “UML (*Unified Modeling Language*) adalah sebuah bahasa yang berdasarkan grafik/gambar untuk memvisualisasi, menspesifikasikan, membangun, dan pendokumentasian dari sebuah sistem pengembangan software berbasis OO (Object-Oriented)”.

Menurut Mulyani (2016: 42), “UML adalah teknik pengembangan sistem yang menggunakan bahasa grafis sebagai alat untuk pendokumentasian dan melakukan spesifikasi pada sistem”.

Berdasarkan pengertian tersebut, dapat disimpulkan bahwa *Unified Modeling Language (UML)* adalah teknik untuk pemodelan desain program dan aplikasinya.

2.2.3 *Use Case Diagram*

Menurut Destiningrum dan Adrian (2017: 34), “ *Use Case Diagram* merupakan pemodelan untuk kelakuan (behavior) sistem informasi yang akan dibuat”.

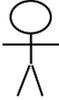
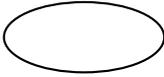
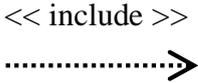
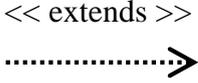
Menurut Mulyani (2016: 42), “ *Use Case Diagram* yaitu diagram yang digunakan untuk menggambarkan hubungan antara sistem dengan aktor”.

Berdasarkan pengertian tersebut, dapat disimpulkan bahwa *Use Case Diagram* adalah pemodelan mengenai gambaran dari sistem informasi yang akan dibuat.



Adapun simbol-simbol dari Use Case Diagram.

Tabel 2.2 Simbol-simbol *Use Case Diagram*

No.	Simbol	Nama	Keterangan
1.		Actor	Seseorang atau apa saja yang berhubungan dengan sistem yang sedang dibangun
2.		Use case	Menggambarkan bagaimana seseorang menggunakan sistem
3.		Relasi asosiasi	Relasi yang dipakai untuk menunjukkan hubungan antara aktor dan use case
4.		Relasi include	Memungkinkan satu use case menggunakan fungsionalitas yang disediakan oleh use case lainnya
5.		Relasi extend	Memungkinkan suatu use case secara optional menggunakan fungsionalitas yang disediakan oleh use case lainnya

(Sumber : Rusmawan, 2019:72-73)

2.2.4 Class Diagram

Menurut Zulkifli dan Farabi (2018:43), “*Class Diagram* adalah diagram yang digunakan untuk menampilkan beberapa kelas serta paket-paket yang ada dalam system atau perangkat lunak yang sedang kita gunakan. Class diagram memberi kita gambaran (diagram statis) tentang sistem/perangkat lunak dan relasi-relasi yang ada didalamnya”.

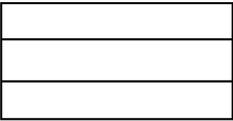
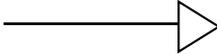
Menurut Saputra (2018:182), “*Class diagram* adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek”.



Berdasarkan pengertian tersebut, dapat disimpulkan bahwa *Class Diagram* (Diagram Kelas) adalah diagram yang digunakan untuk menampilkan beberapa kelas yang ada dalam system atau perangkat lunak yang sedang kita gunakan.

Berikut adalah simbol-simbol yang digunakan pada *Class Diagram*:

Tabel 2.3 Simbol-simbol *Class Diagram*

No.	Simbol	Deskripsi
1.	Kelas 	Kelas pada struktur sistem
2.	Antarmuka/ <i>Interface</i>  nama_interface	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek
3.	Asosiasi/ <i>association</i> 	Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
4.	Assosiasi berarah/ directed association 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
5.	Generalisasi 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus)
6.	Kebergantungan/ dependency 	Relasi antar kelas dengan makna kebergantungan antarmuka
7.	Agregasi/ <i>aggregation</i> 	Relasi antar kelas dengan makna semua bagian (whole-part)

(Sumber : Mukhtar, 2019:85)



2.2.5 Activity Diagram

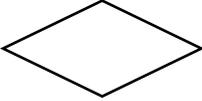
Menurut Zulkifli dan Farabi (2018:43), “*Activity Diagram* adalah diagram yang menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis”.

Menurut Saputra (2018:182), “*Activity diagram* menggambarkan aktivitas sistem bukan apa yang dilakukan oleh aktor, jadi aktivitas yang dapat dilakukan oleh sistem”.

Berdasarkan pengertian tersebut, dapat disimpulkan bahwa *Activity Diagram* adalah diagram yang menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah sistem.

Adapun simbol-simbol yang sering digunakan dalam *Activity Diagram* dapat dilihat pada tabel berikut ini:

Tabel 2.4 Simbol-simbol *Activity Diagram*

No.	Simbol	Nama	Keterangan
1.		Start state	Titik Awal atau permulaan
2.		End state	Titik akhir atau akhir dari aktivitas
3.		Activity	<i>Activity</i> atau aktivitas yang dilakukan oleh actor
4.		Decision	Pilihan untuk mengambil keputusan
5.		Interaction	Alur

(Sumber : Rusmawan, 2019:80)



2.2.6 Sequence Diagram

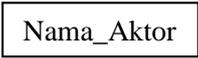
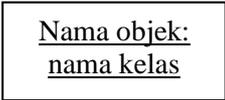
Menurut Haviluddin dalam Suendri (2018:3), *Sequence Diagram* adalah gambaran tahap demi tahap, termasuk kronologi (urutan) perubahan secara logis yang seharusnya dilakukan untuk menghasilkan sesuatu sesuai dengan use case diagram.

Menurut Saputra (2018:182), “*Sequence diagram* adalah suatu diagram yang memperlihatkan atau menampilkan interaksi-interaksi antar objek di dalam sistem yang disusun pada sebuah urutan atau rangkaian waktu”.

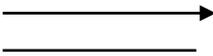
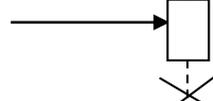
Berdasarkan pengertian tersebut, dapat disimpulkan bahwa *Sequence Diagram* adalah diagram yang menampilkan hubungan antar objek yang dijelaskan dengan rangkaian waktu.

Berikut adalah simbol-simbol yang digunakan pada *Sequence Diagram*:

Tabel 2.5 Simbol-simbol *Sequence Diagram*

No.	Simbol	Deskripsi
1.	<p>Aktor</p>  <p>Atau</p>  <p>Nama_Aktor</p> <p>Tanpa waktu aktif</p>	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tetapi aktor belum tentu berupa orang, biasanya dinyatakan menggunakan kata benda diawal frase nama aktor
2.	<p>Garis hidup/<i>lifeline</i></p> 	Menyatakan kehidupan suatu objek
3.	<p>Objek</p>  <p><u>Nama objek:</u> <u>nama kelas</u></p>	Menyatakan objek yang berinteraksi pesan
4.	<p>Waktu aktif</p> 	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan didalamnya

Lanjutan Tabel 2.5 Simbol-simbol *Sequence Diagram*

No.	Simbol	Deskripsi
5.	Pesan tipe <i>create</i>  <<create>>	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat
6.	Pesan tipe <i>call</i> 1 : nama_metode() 	Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri
7.	Pesan tipe <i>send</i> 1 : masukan 	Menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim
8.	Pesan tipe <i>return</i> 1 : keluaran 	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian
9.	Pesan tipe <i>destroy</i> <<destroy>> 	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada create maka ada destroy

(Sumber : Mukhtar, 2019:84-85)

2.3. Teori Program

2.3.1 PHP

Menurut Abdulloh (2018:127), “PHP merupakan kependekan dari *Hypertext Preprocessor* yaitu bahasa pemrograman web yang dapat disisipkan di skrip HTML dan bekerja di sisi server”.

Menurut Sulistiono (2018:5), “PHP (*Hypertext Preprocessor*) adalah bahasa pemrograman yang digunakan untuk membuat website atau situs dinamis dan menangani rangkaian bahasa pemrograman antara *client side scripting* dan *server side scripting*”.



Berdasarkan pernyataan tersebut, dapat disimpulkan bahwa PHP atau *Hypertext Preprocessor* adalah bahasa pemrograman web atau situs dinamis dan menangani rangkaian bahasa pemrograman antara *client side scripting* dan *server side scripting*.

2.3.2 MySQL

Menurut Rusmawan (2019:97), “MySQL adalah sebuah implementasi dari sistem manajemen basis data relasional (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (*General Public License*)”.

Menurut Raharjo dalam Siregar dan Melani (2018:115), “MySQL merupakan sistem database yang banyak digunakan untuk pengembangan aplikasi web. Karena pengolahan datanya sederhana, memiliki tingkat keamanan yang bagus, dan mudah diperoleh”.

Berdasarkan pernyataan tersebut, dapat disimpulkan bahwa MySQL adalah sebuah tempat penyimpanan data yang mudah digunakan dan dibutuhkan saat ini untuk sebuah website secara gratis.

2.3.3 CodeIgniter

Menurut Sulistiono (2018:7), “Codeigniter adalah sebuah aplikasi *open source* yang berupa kerangka kerja atau *framework* untuk membangun website menggunakan bahasa pemrograman PHP. Tujuannya memungkinkan pengembangan proyek yang lebih cepat daripada penulisan kode dasar atau kode terstruktur, dengan menyediakan banyak *library* yang biasanya digunakan dalam pengerjaan”.

Menurut Betha Sidik dalam Destiningrum dan Ardian (2017:32), “CodeIgniter merupakan Sebuah framework php yang bersifat open source dan menggunakan metode MVC (Model, View, Controller) untuk memudahkan developer atau programmer dalam membangun sebuah aplikasi berbasis web tanpa harus membuatnya dari awal.”



Berdasarkan pernyataan tersebut, dapat disimpulkan bahwa Codeigniter adalah sebuah web application network yang bersifat open source yang digunakan untuk membangun aplikasi php dinamis.

2.3.4 XAMPP

Menurut Sitohang (2018:8), “XAMPP adalah salah satu paket instalasi apache, PHP, dan MySQL secara instant yang dapat digunakan untuk membantu proses instalasi ketiga produk tersebut”.

Menurut Mawaddah dan Fauzi (2018:2), “XAMPP adalah *software web server* apache yang di dalamnya tertanam server MySQL yang didukung dengan bahasa pemrograman PHP untuk membuat website yang dinamis. XAMPP sendiri dapat berjalan pada berbagai macam platform seperti Windows, Linux, Mac OS X dan Solaris”.

Berdasarkan pernyataan tersebut, dapat disimpulkan bahwa XAMPP adalah sebuah *tool* yang menyediakan beberapa perangkat lunak seperti apache, PHP, dan MySQL secara instant yang dapat digunakan untuk membantu proses instalasi ketiga produk tersebut.

2.3.5 HTML

Menurut Abdulloh (2018:7), “HTML merupakan singkatan dari *HyperText Markup Language* yaitu bahasa standar web yang dikelola penggunaannya oleh W3C (*World Wide Web Consortium*) berupa tag-tag yang menyusun setiap elemen dari website”.

Menurut Sulistiono (2018:2), “HTML (*HyperText Markup Language*) adalah sebuah bahasa markah yang digunakan untuk membuat sebuah halaman web, menampilkan berbagai informasi seperti gambar, teks, video dan suara pada penjelajah web Internet, yang ditulis dalam berkas format ASCII agar dapat menghasilkan tampilan wujud yang terintegrasi”.

Berdasarkan pernyataan tersebut, dapat disimpulkan bahwa HTML atau *HyperText Markup Language* adalah bahasa web yang digunakan untuk membuat



halaman web berupa tag-tag yang menyusun setiap elemen dari halaman tersebut seperti gambar, text, video, maupun suara.

2.3.6 CSS

Menurut Abdulloh (2018:45) “CSS adalah singkatan dari *Cascading Style Sheet* yaitu dokumen web yang berfungsi mengatur elemen HTML dengan berbagai properti yang tersedia sehingga dapat tampil dengan berbagai gaya yang diinginkan”.

Menurut Sulistiono (2018:2), “CSS (Cascading Style Sheet) merupakan aturan untuk mengendalikan beberapa komponen dalam sebuah web sehingga akan lebih terstruktur dan seragam. CSS bukan bahasa pemrograman. Sama halnya seperti style dalam aplikasi pengolahan kata, Microsoft Word yang dapat mengatur beberapa style, misalnya heading, subbab, bodytext, footer, images, dan style lainnya untuk dapat digunakan bersama-sama dalam beberapa berkas (file)”.

Berdasarkan pernyataan tersebut, dapat disimpulkan bahwa CSS atau Cascading Style Sheet merupakan aturan untuk mengendalikan elemen-elemen HTML dengan beberapa style seperti header, footer, bodytext, dan style lainnya agar dapat digunakan disemua halaman web.

2.3.7 Bootstrap

Menurut Sulistiono (2018:17), “Bootstrap adalah sebuah pustaka *open source* yang merupakan *framework* CSS dan *Javascript* untuk membuat website yang responsif”.

Menurut Hermanto dan Suyudi (2018:22), “Bootstrap adalah sebuah *library framework* CSS yang dibuat khusus untuk bagian pengembangan front-end website”.

Berdasarkan pernyataan tersebut, dapat disimpulkan bahwa bootstrap merupakan sebuah *framework* CSS dan *Javascript* yang digunakan untuk mengembangkan website agar lebih *responsive*.



2.3.8 JavaScript

Menurut Abdullah (2018:193), “JavaScript merupakan bahasa pemrograman web yang pemrosesannya dilakukan di sisi *client*. Karena berjalan di sisi *client*, JavaScript dapat dijalankan hanya dengan menggunakan browser”.

Menurut Sulistiono (2018:4), “Javascript adalah bahasa pemrograman tingkat tinggi untuk membuat website dinamis, dan JQuery adalah *library* atau pustaka dari JavaScript yang dirancang untuk memudahkan penerapan client side scripting dan menyajikan sebuah paradigma baru pada penanganan *event* pada JavaScript”.

Berdasarkan pernyataan tersebut, dapat disimpulkan bahwa javascript merupakan bahasa pemrograman tingkat tinggi untuk berfungsi membuat aksi dari interaksi user dan web.