

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terkait

Telah ada beberapa penelitian yang dilakukan terkait dengan penerapan *E-Customer Relationship Management*, diantaranya adalah :

1. Berdasarkan penelitian yang dilakukan oleh Abdul Rahman Ariga, Ahmad Supaidi, Iziah Aslamiah, Ali Ibrahim (2018) dengan judul “Implementasi *Customer Relationship Management* (CRM) Pelayanan Pelanggan (*Corporate*) Pada PT. Telkom Witel Sumsel”. Tujuan penelitian yang dilakukan adalah menjaga hubungan baik dengan pelanggan. Karena dari hasil penelitian mempertahankan pelanggan yang sudah ada jauh lebih mudah dibandingkan dengan mendapatkan pelanggan baru. Persaingan bisnis yang sangat ketat akan sangat memerlukan langkah-langkah strategis untuk menghadapinya. Kehadiran *Customer Relationship Management* (CRM) akan mampu mengatasi masalah yang ada salah satu langkah yang juga dapat ditempuh untuk menghadapi masalah tersebut adalah dengan memanfaatkan teknologi yang ada yaitu dengan pemanfaatan *Customer Relationship Management* (CRM) berbasis web.
2. Berdasarkan penelitian yang dilakukan oleh Andhika Octa (2019) dengan judul “*Literature Review: Meningkatkan Kepuasan Pelanggan Di Bengkel Resmi Menggunakan Sistem Manajemen Pelanggan Elektronik*”. Perusahaan dapat memanfaatkan teknologi internet untuk mendekatkan diri dengan

pelanggan secara langsung selama 24 jam dalam 7 hari seminggu dengan cepat dan mudah melalui Sistem Manajemen Pelanggan secara Elektronik. Manajemen Pelanggan Elektronik bukan sekedar sebuah sistem yang dapat memudahkan pelaku bisnis dalam menganalisa kebutuhan ataupun melakukan interaksi langsung kepada pelanggannya, melainkan menjadi kunci penting bagi pelaku bisnis yang berorientasi kepada pelanggan untuk meningkatkan kepuasan para pelanggannya. Penelitian ini ditujukan untuk melakukan kajian terhadap literatur maupun hasil penelitian serta artikel yang berhubungan dengan penerapan Sistem Manajemen Pelanggan Elektronik atau dikenal dengan nama lain e-CRM yang diterapkan pada bengkel resmi. Hasil kajian yang berasal dari jurnal-jurnal yang berkaitan dengan e-CRM di bengkel resmi dilakukan analisis yang lebih dalam lagi.

3. Berdasarkan penelitian yang dilakukan oleh Debbie Mistikaweni dan Lauw Li Hin (2019), dengan judul “Penerapan *Electronic Customer Relationship (E-CRM)* Dalam Upaya Meningkatkan Loyalitas Pelanggan Dan Pelayanan Terhadap Pelanggan *Catering Handayani*”. *Catering Handayani Mandiri* merupakan salah satu perusahaan *Catering* yang bergerak di bidang jasa khususnya pesan menu makanan, yang berusaha memberikan pelayanan terbaik tanpa mengurangi kualitas, mutu dan kenyamanan pelanggannya. Permasalahan yang dihadapi pada *Catering Handayani Mandiri* adalah perusahaan sulit mengidentifikasi pelanggan yang sering bertransaksi, Perusahaan sulit mendapatkan pelanggan baru karena kurangnya media promosi yang belum mendukung, dan belum tersedianya sarana untuk

menampung keluhan dan testimoni pelanggan. Hal ini dapat dicapai dengan memanfaatkan Teknologi dengan model *Electronic Customer Relationship Management* (E-CRM) untuk melayani dan mencapai keinginan pelanggan yang menjadikan nilai tambah bagi perusahaan. Peneliti menggunakan metode pengembangan *prototype* serta bahasa pemrograman PHP dan *database* MySQL.

4. Berdasarkan penelitian yang dilakukan oleh Hendra Darmawan (2018), dengan judul “Analisis dan Perancangan *Electronic Customer Relationship Management* pada Cetta Mom and Baby Spa Pontianak”. Tujuan dari penelitian ini untuk merancang aplikasi E-CRM pada perusahaan Cetta Mom and Baby Spa Pontianak dengan memberikan layanan informasi secara online dalam hal reservasi jasa treatment yang up-date. Metode penelitian yang digunakan dalam penelitian ini adalah metode penelitian dan pengembangan atau yang biasa dikenal dengan *Research and Development*. Hasil penelitian memperlihatkan adanya *website* E-CRM lebih mempermudah konsumen untuk melakukan transaksi dan berkomunikasi dengan pihak Perusahaan serta mempertahankan konsumen lamanya dan juga untuk memperoleh konsumen baru dengan memasarkan barang kepada masyarakat luas.
5. Berdasarkan penelitian yang dilakukan oleh Hani Murdiati, Nana Darna, dan Kasman (2019), dengan judul “Analisis Penggunaan Model E-CRM Pada Perusahaan Smartfren Kota Tasikmalaya”. Penelitian ini difokuskan pada Analisis Penggunaan Model E-CRM Pada Perusahaan Smartfren Kota Tasikmalaya (Suatu Studi Pada Distributor Smartfren Utama Jaya Kota

Tasikmalaya). Permasalahan utama yang dihadapi dalam penelitian ini yaitu adanya keluhan konsumen yang disampaikan secara langsung kepada *outlet* dan *salesman* yang berada dilapangan dan hal itu dirasa kurang efektif karena perusahaan akan lambat mendapatkan informasi yang dibutuhkan konsumen dan konsumen pun akan lambat mendapatkan respon dari perusahaan. Hasil dari penelitian dan pengolahan data menunjukkan bahwa sistem E-CRM yang dibuat oleh PT. Smartfren TelecomTbk yang berupa aplikasi *Mysmartfren* sudah sangat baik dengan mendapatkan respon “Sangat Puas” yang memiliki nilai rata-rata 80,6 atau dibulatkan 81.

II.2. *E-Customer Relationship Management*

Electronic Customer Relationship Management (E-CRM) merupakan suatu metode dalam mengelola relasi antara perusahaan dengan pelanggan dalam rangka peningkatan loyalitas pengkonsumsian produk-produk atau jasa yang ditawarkan oleh perusahaan dengan memanfaatkan media elektronis. Dengan adanya e-CRM ini pihak perusahaan dapat mengintegrasikan dan mengotomatisasi proses layanan pelanggan dalam bidang penjualan, pemasaran, dan layanan produk atau jasa yang berkaitan dengan perusahaan (Rosalina, 2018 : 1).

Electronic Customer Relationship Management (E-CRM) merupakan sebuah strategi bisnis dimana terjadi sebuah proses yang dibangun untuk mempertahankan hubungan dengan konsumen guna menciptakan suasana sama-

sama untuk baik bagi karyawan maupun bagi konsumen, dengan demikian kesuksesan penjualan pun akan lebih maksimal. (Darmawan, 2018 : 1)

Alat-alat yang digunakan untuk mendukung E-CRM adalah (Hani Murdati 2019):

1. *Personalized web page* merupakan *web page* (halaman web) dapat digunakan untuk mencatat kegiatan dan preferensi *customer*.
2. FAQs merupakan fitur *Frequently Asked Questions* (FAQs) adalah sebuah *tool* yang sederhana yang bisa digunakan untuk menangani pertanyaan *customer* yang berulang.
3. *Tracking tools* merupakan *tool* yang digunakan oleh konsumen untuk mengetahui pemesanan (*order*) yang telah dilakukan sehingga menghemat waktu dan biaya perusahaan.
4. E-Mail dan *Automated Respons* digunakan untuk mengirimkan informasi, konfirmasi dan membangun korespondensi dengan *customer*.
5. *Help Disk* atau *call center* berfungsi sebagai *customer service* yang biasanya menggunakan fasilitas telepon Faximili atau e-mail.
6. *Troubleshooting tools* merupakan *tool* yang disediakan perusahaan untuk membantu *customer* memecahkan persoalan yang dihadapi sendiri

Manfaat *Customer Relationship Management* (Ali, 2018) :

1. Mendorong loyalitas pelanggan
2. Mengurangi biaya yang dikeluarkan
3. Meningkatkan efisiensi operasional
4. Meningkatkan pendapatan

II.3. *Unified Modeling Language (UML)*

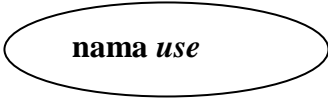
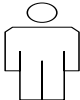

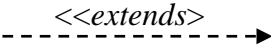
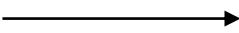
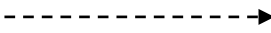
Unified Modeling Language (UML) adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan system.

Unified Modelling Language (UML) adalah bahasa pemodelan untuk sistem atau perangkat lunak yang berparadigma berorientasi objek. Abstraksi konsep dasar UML terdiri dari *structural classification*, *dynamic behavior*, dan model *management* dapat kita pahami *main concepts* sebagai *term* yang akan muncul pada saat membuat diagram dan *view* adalah kategori dari diagram tersebut. UML mendefinisikan diagram-diagram sebagai *Use case diagram*, *Class diagram*, *Statechart diagram*, *Activity diagram*, *Sequence diagram*, *Collaboration diagram*, *Component diagram*, dan *Deployment diagram* (suendri, 2018 : 2)

II.3.1. *Use case Diagram*

Use case Diagram merupakan pemodelan untuk kelakuan sistem informasi yang akan dibangun. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibangun. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada pada sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. (Simatupang, 2019).

Tabel II.1. Simbol Use case Diagram

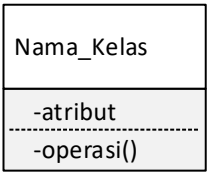


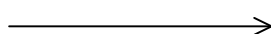

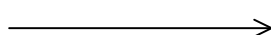
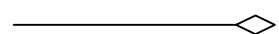
Simbol	Diskripsi
<i>Use case</i> 	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor: biasanya dinyatakan dengan menggunakan kata kerja diawal di awal frase nama <i>use case</i> .
Aktor / <i>actor</i> 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri.
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 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 yang sama dengan <i>use case</i> yang ditambahkan misal Arah panah mengarah pada <i>use case</i> yang ditambahkan.
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</i> 	Include berarti <i>use case</i> yang tambahan akan selalu melakukan pengecekan apakah <i>use case</i> yang ditambahkan telah dijalankan sebelum <i>use case</i> tambahan dijalankan

(Sumber : Julianto Simatupang, 2019)

II.3.2. Class Diagram

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. (Simatupang, 2019).

Tabel II.2. Simbol Class Diagram

Simbol	Diskripsi
<p>Kelas</p> 	Kelas pada struktur sistem.
<p>Antar muka/<i>interface</i></p>  <p>nama_<i>interface</i></p>	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek.
<p>Asosiasi / <i>association</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 juga disertai dengan <i>multiplicity</i> .
<p>Generalisasi</p> 	Relasi antar kelas dengan makna generalisasi - generalisasi spesialisasi (umum khusus).
<p>Kebergantungan / <i>dependency</i></p> 	Relasi antar kelas dengan makna kebergantungan antar kelas
<p>Agregasi / <i>aggregation</i></p> 	Relasi antar kelas dengan makna semua-bagian (<i>whole- part</i>).


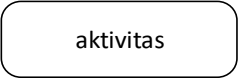



(Sumber : Julianto Simatupang, 2019)

II.3.3. Activity Diagram

Activity Diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak.

Penekanan pada diagram aktivitas adalah menggambarkan aktivitas sistem atau aktivitas yang dapat dilakukan oleh sistem, bukan apa yang dilakukan aktor (Simatupang, 2019).

Tabel II.3. Simbol Activity Diagram




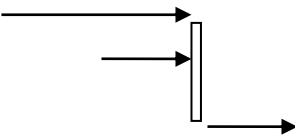

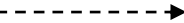
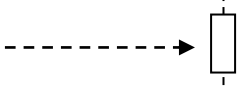
Simbol	Diskripsi
Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
Percabangan / <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari Satu
Penggabungan / <i>Join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu..
Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.

(Sumber : Julianto Simatupang, 2019)

II.3.4. Sequence Diagram

Sequence Diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambar *diagram squence* maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. (Simatupang, 2019).

Tabel II.4. Simbol *Sequence Diagram*

Simbol	Diskripsi
<p>Aktor</p>  <p>Atau nama aktor</p>	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri.
<p>Garis hidup / <i>lifeline</i></p> 	Menyatakan kehidupan suatu objek.
<p>Objek</p> <div style="border: 1px solid black; padding: 5px; width: fit-content;"> <u>nama objek</u> <u>nama kelas</u> </div>	Menyatakan objek yang berinteraksi pesan.
<p>Waktu aktif</p> 	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya
<p>Pesan tipe <i>create</i></p> <p><< create >></p> 	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat. arah panah mengarah pada objek yang memiliki operasi/metode, karena ini memanggil operasi/metode maka operasi/metode yang dipanggil harus ada pada diagram kelas sesuai dengan kelas objek yang berinteraksi.
<p>Pesan tipe <i>send</i></p> <p>1 : masukan</p> 	Menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.
<p>Pesan tipe <i>return</i></p> <p>1 : keluaran</p> 	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.
<p>Pesan tipe <i>destroy</i></p> 	Menyatakan suatu objek mengakhiri hidup objek lain, arah panah yang mengarah pada objek yang diakhiri, sebaiknya jika ada <i>create</i> maka ada <i>destroy</i> .

(Sumber : Julianto Simatupang, 2019)

II.4. PHP (*Hypertext Preprocessor*)

PHP (*Hypertext Preprocessor*) adalah bahasa *script* yang dapat ditanamkan atau disisipkan ke dalam HTML. PHP banyak dipakai untuk membuat program situs web dinamis”. PHP dapat digunakan dengan gratis (*free*) dan bersifat *Open Source*. PHP dirilis dalam lisensi *PHP license*. Untuk membuat program PHP kita diharuskan untuk menginstal web server terlebih dahulu.

Komponen Dasar PHP

1. Sintak dasar PHP

Ada beberapa aturan sintaks yang harus dipenuhi ketika membuat file program PHP.

- a. PHP *opening* dan *closing* tag
- b. PHP mendukung komentar seperti pada bahasa ‘C’, ‘C++’, dan *Unix shell-style*. (*Perl style*)

2. Konstanta dalam PHP Suatu konstanta dapat kita definisikan dengan menggunakan fungsi *define* () yang merupakan salah satu fitur *function* dari PHP.

3. Aritmatika dalam PHP Untuk mempermudah menggunakan operand dan operator pada PHP, diberikan tabel mengenai operator seperti terlihat pada gambar dibawah.

Tabel II.5. Operator Dasar Aritmatika

Tingkat	Operator	Nama Operasi	Contoh
1	+	Positif	+3
2	-	Negatif	-3
3	+	Tambah	3+3
4	-	Kurang	3-3

Tingkat	Operator	Nama Operasi	Contoh
5	*	Kali	3*3
6	/	Bagi	3/3
7	%	Mudulo	5%5

(Sumber: Fitri Ayu, 2018 : 20)

II.5. Basis Data (*Database*)

Basis Data adalah merupakan bagian dari *micro lifecycle*. Sedangkan kegiatan-kegiatan yang terdapat di dalam proses tersebut diantaranya pengumpulan data dan analisis, perancangan *database* secara konseptual, pemilihan DBMS (*Database Management System*), perancangan *database* secara logika (*data model mapping*), perancangan *database* secara fisik, dan implementasi sistem *database* (Efendy, 2018 : 2).

II.6. MySQL (*Database*)

MySQL (*My Structure Query Language*) adalah sistem manajemen *Database SQL* yang bersifat *Open Source* dan paling populer saat ini. Sistem *Database MySQL* mendukung beberapa fitur seperti *multithreaded*, *multiuser* dan *SQL Database management system* (Ayu, 2018 : 21).