

BAB II

LANDASAN TEORI

2.1 E-CRM (*Electronic Customer Relationship Management*)

Menurut Kelly, dkk (2021) Customer Relationship Managemen (CRM) adalah konsep yang paling penting dalam pemasaran modern yang dalam arti lebih luas merupakan keseluruhan proses membangun dan memelihara hubungan pelanggan yang menguntungkan dengan mengantarkan nilai dan kepuasan pelanggan yang unggul. CRM kini telah diaplikasikan dalam bentuk *online* yang disebut e-CRM (*Electronic Customer Relationship Management*).

E-CRM adalah CRM yang memanfaatkan proses bisnis dan data secara online. E-CRM secara konsisten mengatur pribadi relevan dan interaksi produktif, bermakna untuk membangun hubungan yang sukses dengan pelanggan, penjual, karyawan, investor, dan yang lainnya menggunakan teknologi terbaru.

2.2 PT. Batu Makmur

PT. Batu Makmur adalah perusahaan yang menyediakan solusi untuk konstruksi dan proyek tambang. Kami berkonsentrasi pada penyediaan bahan baku seperti batu split, pasir, peralatan berat, transportasi truk dan banyak lainnya. Di samping tambang kami sendiri dan alat berat kami juga menyediakan solusi kustom profesional untuk *stockpile* (persediaan) dan menyewa gudang.

Kami adalah perusahaan konstruksi sipil yang mengkhususkan diri dalam pembangunan transportasi dan reklamasi lahan. Proyek infrastruktur transportasi kami antara lain jalan raya dan jembatan. Kami menyediakan layanan kontraktor umum termasuk penggalian, pembukaan lahan, pengaspalan beton dan aspal, konstruksi jembatan dan struktur besar serupa, melayani klien sektor publik dan swasta.

Kami melakukan sebagian besar pekerjaan menggunakan karyawan dan peralatan kami sendiri. Bisnis konstruksi dimulai pada 1986. Kami telah memperluas layanan dan pasar kami daerah. Saat ini kami memiliki Ketua Kice di Sukaraja (Bandar Lampung) dan 6 (enam) cabang di Soekarno Hatta (Bandar Lampung),

Kaliasin, Sebalang, Bakauheni, Rangai Tritunggal (Lampung Selatan), dan Tegineneng (Pesawaran). Layanan pelanggan sangat penting bagi PT. Batu Bakmur. Kami bangga akan hal itu. Kami ingin pelanggan kami senang dan puas saat menggunakan layanan kami.

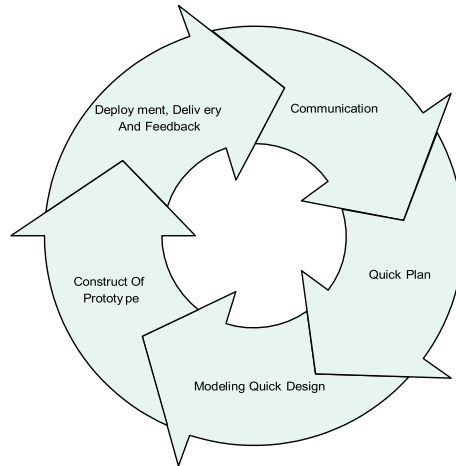
2.3 Pengertian Website

Menurut Elgamar (2020), *website* adalah sebuah media yang memiliki banyak halaman saling terhubung (*hyperlink*), dimana *website* memiliki fungsi dalam memberikan informasi berupa teks, gambar, video, suara, dan animasi atau penggabungan dari semuanya. Website pada saat sekarang ini umumnya bersifat dinamis. Karakteristik utama yang dimiliki oleh website adalah halaman-halaman yang saling terhubung, dan dilengkapi dengan domain sebagai alamat (url) atau *www (World Wide Web)* dan juga hosting sebagai media yang menyimpan banyak data. *Website* dapat diakses menggunakan jaringan internet dengan *platform* yang disebut *browser*, seperti chrome, mozilla firefox, dan sebagainya.

2.4 Metode Prototipe

Pengembangan perangkat lunak dalam sistem ini menggunakan metode prototipe. Menurut kutipan pada penelitian Puspita (2019), *prototype* merupakan metode yang efektif dalam merancang perangkat lunak. *Prototype* dimulai dengan mengumpulkan kebutuhan yang akan dirancang. Pengembang mendefinisikan objek keseluruhan dari perangkat lunak, mengidentifikasi segala aktifitas yang diketahui dan kemudian melakukan “perancangan kilat”. Perancangan kilat berfokus pada penyajian dari aspek-aspek perangkat lunak tersebut yang akan nampak bagi pelanggan atau pemakai (contohnya pendekatan *input* dan format *output*).

Menurut kutipan pada penelitian Gunawan dan Puspita (2017), *prototype model* adalah salah satu metode pengembangan perangkat lunak yang banyak digunakan. Dengan metode *prototype*, pengembangan dan pelanggan dapat saling berinteraksi selama proses pembuatan sistem. Prototipe memiliki lima tahapan siklus. Siklus atau ilustrasi dari metode prototipe dapat dilihat pada Gambar 2.1.



Gambar 2.1 Ilustrasi Model Prototipe

a. *Communication*

Pembuatan prototipe dimulai dengan dilakukannya komunikasi antara tim pengembang perangkat lunak dengan para pelanggan. Tim pengembang perangkat lunak akan melakukan pertemuan-pertemuan dengan *parastakeholder* untuk mendefinisikan sasaran keseluruhan untuk perangkat lunak yang akan dikembangkan, mengidentifikasi spesifikasi kebutuhan apa pun yang saat ini diketahui, dan menggambarkan area-area dimana definisi lebih jauh pada iterasi selanjutnya merupakan keharusan.

b. *Quick Plan*

Iterasi pembuatan prototipe direncanakan dengan cepat dan pemodelan (dalam bentuk "rancangan cepat") dilakukan. Suatu rancangan cepat berfokus pada representasi semua aspek perangkat lunak yang akan dilihat oleh para pengguna. Perencanaan cepat disini adalah menggunakan pemodelan arsitektur sistem.

c. *Modeling Quick Design*

Rancangan cepat (*quick design*) akan memulai konstruksi pembuatan prototipe. Pada penelitian ini, desain yang digunakan adalah *use case diagram*, *activity diagram*, *class diagram*, dan *interface sistem*.

d. *Construck Of Prototype*

Pada tahap ini dilakukan pembuatan pemograman web dengan menggunakan bahasa pemograman *Javascript* dan *database MySQL*.

e. *Deployment, Delivery, and Feedback*

Prototipe kemudian akan diserahkan kepada para *stakholder* dan kemudian mereka akan melakukan evaluasi-evaluasi tertentu terhadap prototipe yang telah dilakukan sebelumnya, kemudian akhirnya akan memberikan umpan-balik yang akan digunakan untuk memperhalus spesifikasi kebutuhan. Iterasi akan terjadi saat prototipe diperbaiki untuk memenuhi kebutuhan dari para *stakeholder*, sementara pada saat yang sama memungkinkan kita untuk lebih memahami kebutuhan apa yang akan dikerjakan pada saat iterasi selanjutnya




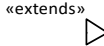
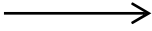
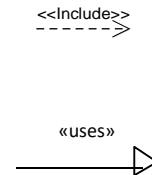
2.5 UML (*Unified Modelling Language*)

Pemodelan sistem yang digunakan menggunakan pemodelan UML. UML adalah suatu metode dalam pemodelan secara visual yang digunakan sebagai sarana perancangan sistem berorientasi objek. UML juga dapat didefinisikan sebagai suatu bahasa standar visualisasi, perancangan, dan pendokumentasian sistem, ataudikenal juga sebagai bahasa standar penulisan blueprint sebuah *software*. Jenis pemodelan UML yang digunakan dalam pengembangan sistem pada penelitian ini adalah *use case diagram*, *activity diagram*, dan *class diagram*.

2.5.1 *Use Case Diagram*

Menurut Rosa dan Shalahuddin (2018), *use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi- fungsi itu. Syarat penamaan pada *use case* adalah nama didefinisikan sesimpel mungkin dan dapat dipahami. Ada dua hal utama pada *use case* yaitu pendefinisian apa yang dibuat aktor dan *use case*. Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi, walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang. *Use case* merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.

Tabel 2.1 Simbol *Use Case Diagram*


Keterangan	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 diawal-awal frase nama <i>use case</i>
Aktor		Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar itu sendiri. Jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda diawal frase nama aktor.
Asosiasi		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		Relasi use case tambahan ke sebuah <i>use case</i> , dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan.
Generalisasi		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>		Ada dua sudut pandang yang cukup besar mengenai <i>include</i> di <i>use case</i> : a. Include berarti use case yang ditambahkan akan selalu dipanggil saat use case tambahan dijalankan b. Include berarti use case yang tambahan akan selalu melakukan pengecekan.

2.5.2 Activity Diagram

Menurut Rosa dan Shalahuddin (2018), diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau

proses bisnis. Perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.


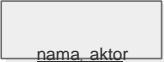
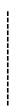
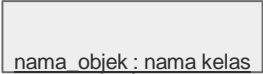

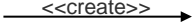
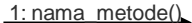
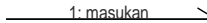
Tabel 2.2 Simbol *Activity Diagram*

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 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
Penggabungan 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
<i>Swimlane</i> 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.
Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.

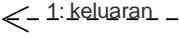
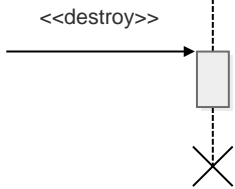
2.5.3 *Sequence Diagram*

Menurut Rosa dan Shalahuddin (2018), diagram skuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Banyaknya diagram skuen yang harus digambar adalah minimal sebanyak pendefinisian *use case*. Adapun simbol- simbol yang terdapat pada *activity diagram* adalah seperti pada Tabel 2.3.

Tabel 2.3 Simbol *Sequence Diagram*

Simbol	Deskripsi
 <p>nama 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 di luar sistem informasi itu sendiri. Jadi walaupun aktor gambar orang tetapi aktor belum tentu merupakan orang
<p>Garis hidup / <i>lifeline</i></p> 	Menyatakan kehidupan suatu objek
<p>Objek</p>  <p>nama_objek : nama_kelas</p>	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> 	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat
<p>Pesan tipe <i>call</i></p> 	Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri
<p>Pesan tipe <i>send</i></p> 	Menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya. Arah panah mengarah pada objek yang dikirim


Tabel 2.3 Simbol *Sequence Diagram* (Lanjutan)

Simbol	Deskripsi
Pesan tipe <i>return</i> 	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
Pesan tipe <i>destroy</i> 	Menyatakan suatu objek mengakhiri suatu objek yang lain. Arah panah mengarah pada objek yang diakhiri

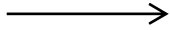

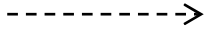

2.5.4 Class Diagram

Menurut Rosa dan Shalahuddin (2018), 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. Atribut merupakan variabel-variabel yang dimiliki suatu kelas, sedangkan operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas.

Tabel 2.4 Simbol *Class Diagram*

Simbol	Deskripsi
Kelas 	Kelas pada struktur sistem.
Natarmuka/ <i>interface</i> 	Sama dengan konsep <i>interface</i> dalam pemograman berorientasi objek.
Asosiasi 	Relasi antar kelas dalam makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .

Gambar 2.4 Simbol *Class Diagram* (Lanjutan)

Simbol	Deskripsi
Asosiasi berarah 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya jugadisertai dengan <i>multiplicity</i> .
Generalisasi 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum-khusus).
Kebergantungan 	Relasi antar kelas dengan makna kebergantungan antar kelas.
Agregasi 	Relasi antar kelas dengan makna semua bagian (<i>whole-part</i>).

2.6 Bahasa Pemograman *Javascript*

Bahasa pemograman yang digunakan dalam pembuatan web ini adalah *Javascript*. Menurut M Ibnu (2020), *javascript* adalah kode untuk menyusun halaman web yang memungkinkan pada sisi klien. *Javascript* adalah bahasa yang digunakan agar dokumen HTML yang ditampilkan dalam *browser* menjadi lebih interaktif. *Javascript* memberikan beberapa fungsionalitas ke dalam halaman web sehingga dapat menjadi program yang disajikan dengan menggunakan antarmuka web. *Javascript* merupakan bahasa *script*, bahasa yang tidak memerlukan *compiler* untuk menjalankannya, cukup dengan interpreter dan tidak perlu ada proses kompilasi terlebih dahulu agar program dapat dijalankan. Oleh karena itu, *javascript* dapat dijalankan di dalam aplikasi *browser* yang merupakan sebuah

aplikasi interpreter, asalkan *browser* tersebut telah mendukung penggunaan *javascript*.

2.7 MySQL

Pengelolaan DBMS (*Database Management System*) yang digunakan dalam pembuatan sistem informasi berbasis web ini adalah *MySQL*. Menurut Fitri (2020), *MySQL* adalah pengembangan lanjutan dari proyek UNIREG yang dikerjakan oleh Michael Monty Widenius dan TcX (perusahaan perangkat lunak asal Swedia). *MySQL* adalah DBMS yang *open source* dengan dua bentuk lisensi, yaitu *free software* (perangkat lunak bebas) dan *shareware* (perangkat lunak berpemilik yang penggunaannya terbatas). Jadi, *MySQL* adalah *database server* yang gratis dengan lisensi GNU *General Public License* (GPL) sehingga dapat dipakai untuk keperluan pribadi atau komersial tanpa harus membayar lisensi yang ada.

Pengelolaan DBMS yang digunakan dalam pembuatan aplikasi ini adalah *MySQL*. Menurut kutipan yang diambil pada penelitian yang dilakukan oleh Muslihudin dan Helmiyanto (2020), *MySQL* (*My Structure Query Language*) adalah suatu perangkat lunak *database* relasi atau *Relational Database Management System* (RDBMS) yang didistribusikan gratis dibawah lisensi GPL (*General Public License*). Dimana setiap orang bebas menggunakan *MySQL*, namun tidak boleh dijadikan produk turunan yang dijadikan *closed source* atau komersial.

Seperti yang sudah disebutkan sebelumnya, *MySQL* masuk ke dalam jenis RDBMS (*Relational Database Management Sistem*). Maka dari itu, istilah semacam baris, kolom, tabel dipakai pada *MySQL*. Menurut Enterprise (2018), *MySQL* merupakan *server* yang melayani *database*, kita dapat mempelajari pemrograman khusus yang disebut *query* (perintah) SQL. *MySQL* merupakan *database engine* atau *server database* yang mendukung bahasa SQL sebagai bahasa interaktif dalam mengelola data. *MySQL* adalah sebuah perangkat lunak sistem manajemen basis data SQL atau DBMS yang *multithread*, *multi-user*.

2.8 Penelitian Terkait

Adapun penelitian terdahulu terkait dengan pembuatan sistem informasi dengan antarmuka web adalah sebagai berikut :

- a. Menurut Kurniawan (2020), dengan penelitian yang berjudul “Sistem Informasi Monitoring Program Keluarga Harapan (PKH) Berbasis Web” menyimpulkan bahwa sistem informasi monitoring program keluarga harapan sebagai pendukung dalam melakukan kegiatan monitoring program keluarga harapan dengan memberikan solusi untuk membantu pendamping dalam pelaporan komitmen peserta program keluarga harapan dan membantupelaporan bulanan pendamping program keluarga harapan.
- b. Menurut Apriyanto dan Yoga (2020), dengan penelitian yang berjudul “Perancangan Aplikasi Web Penjualan Pakaian Muslim” menyimpulkan bahwa aplikasi ini mempermudah dan mempercepat dalam transaksi jual beli khususnya untuk produk pakaian muslim serta dapat digunakan untukmemperluas wilayah pemasaran dalam penjualan dengan penghematan pada sisi biaya operasionalnya.
- c. Menurut Afandi (2020) menyimpulkan rancang- bangun sistem informasi badan usaha milik desa (BUMDEs) di Desa Sidokayo berbasis web mobile disini berhasil dirancang menggunakan metode Extreme Programming danbahasa pemrograman PHP berhasil dibuat dengan alamat link bumdessidokayo.com. Dengan adanya sistem informasi badan usaha milik desa(BUMDes) di Desa Sidokayo berbasis web mobile menghemat waktu dalam melakukan pemesanan karena web ini dapat diakses kapan saja, Dan untuk mempermudah promosi pemasaran yang dikelola oleh BUMDes.