

BAB II

TINJAUAN PUSTAKA

2.1. Teori Terkait

Dalam penelitian ini, ada beberapa referensi jurnal terdahulu yang penulis gunakan untuk menjadi dasar teori pembuatan Rancangan Sistem *E-commerce* Toko Pecah Belah Hasan Dengan *Payment gateway* Midtrans. Dalam memperkuat landasan penelitian ini, beberapa referensi jurnal terdahulu yang relevan dengan topik *e-commerce* dan *payment Gateway* digunakan sebagai acuan. Rangkuman hasil kajian pustaka tersebut dapat dilihat pada Tabel 2.1 berikut.

Tabel 2.1 Teori Terkait

No	Jurnal	Penjelasan
1.	<p>“Implementasi <i>Payment gateway</i> dengan Menggunakan Midtrans pada <i>Website</i> UMKM Geberco”</p> <p>Yenni Fatman, Nadaa Khoirun Nafisah, Pugud Bendoro Jembar Pambudi, 2023</p>	<p>Permasalahan :</p> <p>Toko Geberco dalam proses bisnisnya masih menunggu beberapa pelanggan datang memesan dan pembayaran secara langsung dengan uang tunai, yang menyebabkan penurunan dalam penjualannya.</p> <p>Metode :</p> <p>Dalam penelitian ini penulis menggunakan método pengembangan perangkat lunak Agile, dimana iterasi dilakukan secara berulang.</p> <p><i>Outcome :</i></p> <p>Sistem <i>e-commerce</i> Toko Geberco dengan pengimplementasian <i>payment gateway</i> midtrans</p>

		dengan project <i>website</i> Laravel.
2.	Edy Nasri, Elu Nuryani, Nanang Gustiawan, 2021	<p>Permasalahan : Pada Toko Rumah Kemasan Pandeglang, proses penjualan hanya memiliki toko konvensional.</p> <p>Metode : Dalam penelitian ini penulis menggunakan pemrogramming ekstrim dengan tahapan <i>planing, design, coding, testing</i>.</p> <p><i>Outcome</i> : Sistem <i>E-commerce website</i> dengan <i>payment gateway</i> midtrans untuk Rumah Kemasan</p>
3.	<p>“Penerapan <i>Payment gateway</i> dan <i>Tracking</i> Barang pada <i>E-commerce</i> Toko Dazzle Berbasis <i>Website</i>”</p> <p>Muhammad Ramadhan Febrianto, 2020</p>	<p>Pemasalahan : Kesulitan yang dialami ketika pemesanan <i>online</i> yang dilakukan melalui <i>Whatsapp</i> yaitu penentuan ongkos kirim dan konsumen harus menunggu tanggapan dari pihak toko.</p> <p>Metode : Dalam penelitian ini penulis menggunakan 2 metode yaitu wawancara, Observasi, dan studi pustaka</p> <p><i>Outcome</i> : Web <i>E-commerce</i> dengan pembayaran <i>payment gateway</i> dan dapat <i>tracking</i> barang.</p>

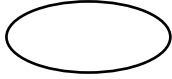


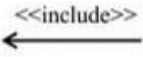
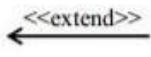
2.2. Landasan Teori

Bahasa Pemodelan Terpadu atau *Unified Modeling Language* (UML) merupakan standar visual yang digunakan untuk memodelkan dan merancang sistem perangkat lunak secara sistematis. *UML* berperan dalam membantu pengembang untuk memvisualisasikan struktur, perilaku, serta interaksi antar komponen dalam sistem yang sedang dirancang. *UML* terdiri dari berbagai jenis diagram yang merepresentasikan permasalahan beserta solusinya[6]. Dalam penelitian ini, terdapat empat jenis diagram yang akan digunakan, yaitu:

1) *Diagram Use Case*

Diagram Use Case adalah jenis diagram *Unified Modeling Language* (UML) yang digunakan untuk memodelkan interaksi antara aktor dengan sistem yang akan dibangun. Diagram ini berfungsi untuk mendeskripsikan kebutuhan fungsional sistem berdasarkan sudut pandang pengguna, sehingga mempermudah pemahaman mengenai proses bisnis yang akan dijalankan oleh sistem. Dalam diagram *Use Case*, terdapat elemen penting seperti aktor yang berinteraksi dengan sistem, dan *Use Case* yang merepresentasikan fungsionalitas yang disediakan oleh sistem[7]. Untuk mempermudah pemodelan sistem menggunakan UML, simbol-simbol pada *Use Case Diagram* dijelaskan dalam Tabel 2.2.

Tabel 2 2 Simbol UseCase





No	Gambar	Keterangan
1		Elips yang menunjukkan aktivitas atau layanan spesifik yang diberikan sistem kepada aktor.
2		<i>Actor</i> : Simbol manusia yang menggambarkan entitas luar sistem yang berinteraksi langsung dengan fungsi sistem..
3		<i>Asosiasi</i> : Garis lurus yang menghubungkan aktor dengan <i>use case</i> sebagai tanda keterlibatan atau interaksi.
4		<i>Include</i> : Panah putus-putus berlabel << <i>include</i> >> yang menunjukkan suatu proses selalu melibatkan proses lainnya.
5		<i>Extend</i> : Panah putus-putus dengan label << <i>extend</i> >> yang menggambarkan proses




		tambahan yang bersifat opsional.
--	--	----------------------------------

2) *Diagram Class*

Diagram ini digunakan untuk merepresentasikan aspek statis dari suatu sistem, mencakup kelas-kelas yang terlibat, atribut-atribut yang dimiliki, serta relasi antar kelas seperti pewarisan (*inheritance*) dan asosiasi (*association*). Simbol-simbol yang digunakan pada *Class Diagram* beserta fungsinya dijelaskan secara rinci pada Tabel 2.3.

Tabel 2.3 Simbol *Class*

No	Gambar	Nama	Keterangan
1		<i>Generation</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
2		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek
3		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama
4		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem menghasilkan suatu hasil terukur bagi suatu <i>actor</i> .

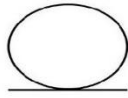
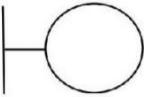
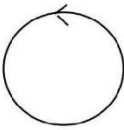
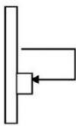


5		<i>Realization</i>	Operasi yang benar-benar dilakukan suatu objek
6		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
7		<i>Association</i>	Upaya untuk menghubungkan objek satu dengan objek yang lainnya.

3) *Diagram Sequence*

Diagram urutan (*sequence diagram*) salah satu jenis diagram dalam *Unified Modeling Language* (UML) yang digunakan untuk memvisualisasikan interaksi antar objek dalam suatu sistem berdasarkan urutan waktu. Diagram ini memperlihatkan bagaimana objek-objek saling berkomunikasi dan berinteraksi secara berurutan dalam eksekusi sebuah skenario atau proses tertentu[8].

Penjelasan mengenai simbol-simbol yang digunakan dalam *Sequence Diagram* disajikan pada Tabel 2.4 berikut.






Tabel 2.4 Simbol *Sequence*

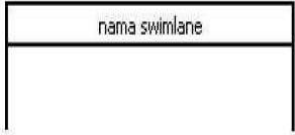
No	Gambar	Nama	Keterangan
1		<i>Entity Class</i>	Gambaran sistem sebagai landasan dalam menyusun basis data
2		<i>Boundary Class</i>	Menangani komunikasi antar lingkungan sistem
3		<i>Control Class</i>	Bertanggung jawab terhadap kelas-kelas objek yang berisi logika
4		<i>Recursive</i>	Pesan untuk dirinya
5		<i>Activation</i>	Mewakili proses durasi aktivasi sebuah operasi
6		<i>Life Line</i>	Komponen yang digambarkan garis putus terhubung dengan objek

4) *Diagram Activity*

Diagram aktivitas merupakan salah satu jenis diagram dalam *Unified Modeling Language* (UML) yang berfungsi untuk memodelkan alur kerja atau rangkaian aktivitas dalam suatu sistem atau proses. Diagram ini menggambarkan langkah-langkah, tindakan, serta pengambilan keputusan yang terjadi secara berurutan selama proses berlangsung. Simbol-simbol utama yang digunakan untuk membangun *Activity Diagram* dijelaskan dalam Tabel 2.5.

Tabel 2.5 Diagram Activity

No	Simbol	Keterangan
1		<i>Start Point</i> : Simbol lingkaran kecil berwarna hitam yang menunjukkan titik permulaan dari alur proses.
2		<i>End Point</i> : Lingkaran hitam dengan lingkaran putih di sekelilingnya, menandakan akhir dari seluruh aktivitas dalam proses.
3		<i>Activity</i> : Digambarkan sebagai persegi panjang dengan sudut membulat, menyatakan suatu langkah atau tindakan dalam proses.
4		<i>Fork/percabangan</i> : Digambarkan dengan garis tebal horizontal atau vertikal, menyatakan pemisahan satu alur menjadi beberapa proses paralel.
5		<i>Decision Points</i> : Berbentuk belah ketupat, simbol ini digunakan untuk menggambarkan percabangan alur berdasarkan kondisi tertentu.

6		<p><i>Swimlane</i> : pembagian diagram aktivitas ke dalam kolom atau baris untuk menunjukkan tanggung jawab masing-masing aktor atau bagian dalam menjalankan aktivitas tertentu.</p>
---	---	---

Berikut *tools* yang akan dipakai:

1. *HyperText Markup Language (HTML)*

Bahasa pemrograman *HyperText Markup Language (HTML)* merupakan bahasa yang digunakan untuk membangun struktur serta isi dari halaman web di internet. *HTML* menjadi fondasi utama bagi hampir seluruh situs web yang ada saat ini dan merupakan komponen penting dalam pengembangan web. Dengan *HTML*, pengembang dapat menentukan berbagai elemen penyusun halaman, seperti teks, gambar, video, tautan, dan elemen-elemen lain yang akan ditampilkan kepada pengguna[9].

2. *JavaScript*

Bahasa pemrograman *JavaScript* merupakan bahasa pemrograman yang digunakan oleh pengembang untuk menciptakan halaman web yang bersifat interaktif. Bahasa ini memungkinkan berbagai fitur dinamis, seperti pembaruan konten secara real-time, animasi, hingga integrasi peta interaktif, sehingga mampu meningkatkan pengalaman pengguna pada sebuah situs web. Sebagai bahasa *client-side scripting*, *JavaScript* menjadi salah satu komponen utama dalam pengembangan *World Wide Web*. Contohnya, saat pengguna menjelajahi internet dan

menjumpai elemen seperti *image carousel*, menu tarik-turun yang muncul saat diklik, atau perubahan warna elemen secara otomatis, semua itu merupakan hasil implementasi dari JavaScript[10].

3. *Cascading Style Sheet (CSS)*

Bahasa pemrograman *CSS* adalah bahasa yang digunakan untuk mengatur tampilan visual dari halaman web. Dengan menggunakan *CSS*, pengembang dapat mengatur gaya elemen-elemen dalam halaman seperti warna teks, ukuran font, jarak antar elemen, tata letak, dan efek visual lainnya. *CSS* memungkinkan pemisahan antara konten dengan desain, sehingga kode *HTML* yang digunakan untuk struktur halaman tetap terpisah dan lebih rapi. Dalam pengembangan web modern, *CSS* menjadi standar penting untuk menciptakan tampilan *website* yang responsif dan menarik secara estetika, sekaligus mendukung pengalaman pengguna yang lebih baik di berbagai perangkat[11].

4. MySQL

Sistem manajemen basis data MySQL adalah *Database Management System (DBMS)* yang banyak digunakan dalam aplikasi web, menggunakan bahasa *Structured Query Language (SQL)* untuk mengelola data dalam bentuk tabel baris dan kolom. MySQL tersedia dalam dua lisensi, yaitu versi bebas pakai dan versi *shareware* dengan batasan tertentu, serta berfungsi sebagai penghubung antara aplikasi dan server dengan pendekatan relasional[12].

5. *Dbeaver*

Perangkat lunak *DBeaver* adalah aplikasi yang digunakan untuk membantu proses manajemen basis data dengan tampilan antarmuka grafis yang memudahkan pengguna. Perangkat lunak ini mendukung berbagai macam jenis basis data, seperti *MySQL*, *PostgreSQL*, *MariaDB*, *SQLite*, *Oracle*, *SQL Server*, dan lainnya. Dengan *DBeaver*, pengguna dapat membuat, mengedit, dan mengelola struktur data, menjalankan perintah *SQL*, serta memantau data secara lebih mudah dan efisien tanpa harus menggunakan perintah berbasis teks[13].