

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Penelitian oleh Hendrawati (2020) berjudul "Sistem Informasi Pemesanan Tiket Bus Berbasis Web di PO. Karunia Bakti" bertujuan untuk menciptakan sistem informasi yang memudahkan pemesanan tiket dan memberikan informasi yang diperlukan tentang PO. Karunia Bakti melalui sebuah situs web. Ini akan memungkinkan pelanggan untuk dengan mudah memesan dan membeli tiket secara online, serta memperoleh jadwal bis, ketersediaan tiket, dan informasi perusahaan[5].

Tujuan dari proyek berbasis *web* dengan menggunakan Metode Waterfall ini adalah untuk memudahkan pelanggan dalam membeli tiket bus secara online, berdasarkan penelitian Geby Yolanda Sinaga yang dimuat dalam jurnal penelitiannya, bertujuan untuk mempermudah, mempercepat, dan meningkatkan efisiensi proses pemesanan tiket bus. Pengembangan sistem pemesanan tiket berbasis web sangat penting untuk mencapai tujuan ini. Fungsi pencarian yang canggih pada sistem ini memungkinkan pengguna untuk memilih rute, tanggal, jumlah penumpang, dan opsi lainnya[6].

Sutabri (2012) menyatakan: "Data harus diolah lebih lanjut karena masih dalam keadaan belum diolah dan belum dapat mengungkapkan banyak hal." Informasi dibuat dengan mengolah data menggunakan suatu

model. Prosedur tertentu digunakan untuk mengolah data. Data dunia nyata adalah fakta yang belum diolah yang dapat direpresentasikan sebagai teks, angka, karakter, gambar, atau video yang dapat diolah untuk menghasilkan informasi[7].

2.2 Landasan Teori

2.2.1 Sistem Informasi

Sebuah proses dilakukan menggunakan teknologi (seperti komputer), prosedur, data, dan orang Sistem informasi adalah alat yang dibuat untuk membantu dalam pengambilan keputusan. Sistem ini terdiri dari bagian-bagian yang saling berfungsi di dalam organisasi agar dapat mencapai tujuan yang diinginkan tertentu, yaitu untuk menyediakan informasi[9].

2.2.2 Website

Di WWW (*World Wide Web*) dan tentu saja di Internet, situs web adalah kumpulan berbagai jenis halaman situs yang diringkas menjadi domain atau subdomain. Dokumen yang ditulis dalam format *Hyper Text Markup Language* (HTML) biasanya membentuk halaman situs web[10].

2.2.3 Tiket

Salah satu dokumen perjalanan yang diterbitkan oleh maskapai penerbangan adalah tiket, yang merupakan kontrak tertulis antara dua pihak yang mencakup informasi penerbangan untuk periode waktu tertentu dan syarat yang harus diikuti oleh pelanggan saat

memanfaatkan layanan perjalanan udara. Tiket adalah dokumen perjalanan yang mencakup kontrak dengan ketentuan yang telah disepakati dan diikuti oleh baik wisatawan maupun perusahaan maskapai[11].

2.2.4 Pemesanan

Pelanggan selalu memesan sebelum melakukan pembelian. Cara lain untuk mendefinisikan pemesanan adalah dengan meminta penjual untuk membeli barang atau jasa. Biasanya, hal ini dilakukan selama pembelian dan penjualan. Komunikasi langsung dengan penjual merupakan langkah pertama dalam proses pemesanan, setelah itu pelanggan melakukan pembelian untuk barang yang diinginkan. Pelanggan membayar setelah barang yang diminta tersedia[12].

2.2.5 Pembayaran

Pembayaran adalah transfer uang yang dilakukan untuk memenuhi kewajiban yang timbul dari kegiatan ekonomi melalui sistem lembaga, peraturan, dan prosedur[13].

2.2.6 Visual Studio Code

Marketplace *Visual Studio Code* memungkinkan pengguna untuk memasang plugin untuk berbagai bahasa pemrograman, termasuk C++, C#, Python, Go, Java, dan lainnya. *Visual Studio Code* (VS Code) adalah editor teks untuk menulis kode program yang mendukung *JavaScript*, *Typescript*, dan *Node.js*. *Intellisense*,

Git Integration, Debugging, dan alat ekstensi yang memperluas fungsionalitas editor teks merupakan beberapa fitur yang ditawarkan oleh *Visual Studio Code*. Dengan setiap edisi baru *Visual Studio Code*, fungsionalitas ini akan terus berkembang.



Gambar 2.1 *Visual Studio Code*

2.2.7 XAMPP

Program *XAMPP*, yang merupakan singkatan dari Apache, MySQL, PHP, dan Perl, digunakan untuk menjalankan situs *web* berbasis PHP di komputer lokal dengan MySQL sebagai pengolah data utamanya. Secara umum, *XAMPP* didefinisikan sebagai perangkat lunak pemrograman *web* yang komprehensif yang memfasilitasi pembuatan situs *web* di komputer lokal, sehingga prosesnya lebih aman dan cepat[14].



Gambar 2.2 *Xampp*

2.2.8 *CodeIgniter*

CodeIgniter adalah kerangka kerja yang dirancang PHP yang mempercepat pengembangan web. Kerangka kerja itu sendiri dapat dilihat sebagai serangkaian infrastruktur, Kerangka ini dirancang untuk membantu pengembang web dalam menciptakan aplikasi web dengan lebih mudah. Ia menawarkan pustaka waktu yang memudahkan dalam pengelolaan dan pengembangan proyek. Dengan menggunakan kerangka ini, pengembang dapat bekerja lebih efektif dan efisien dalam membuat aplikasi menghasilkan aplikasi daring yang andal dengan lebih cepat tanpa mengorbankan fleksibilitas. Karena dikembangkan oleh komunitas sumber terbuka dan, menurut pendapat penulis, cukup mudah digunakan, kerangka kerja *CodeIgniter* dipilih[15].

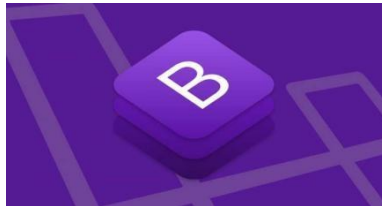


Gambar 2.3 *CodeIgniter*

2.2.9 *Bootstrap*

Untuk mempercepat dan menyederhanakan proses pengembangan situs web, *Bootstrap* adalah kerangka kerja front-end yang fantastis yang mengutamakan tampilan situs web di perangkat seluler, seperti telepon pintar. *Bootstrap* menyediakan *HTML*, *CSS*,

dan *JavaScript* yang siap digunakan dan mudah untuk dikembangkan. Ini adalah kerangka kerja yang dikenal sebagai Bootstrap dan digunakan untuk membuat situs web yang responsif[16].



Gambar 2.4 *Bootstrap*

2.2.10 PHPMyAdmin

Bahasa pemrograman PHP digunakan untuk membuat *PhpMyAdmin*, aplikasi/perangkat lunak sumber terbuka gratis yang mengelola basis data MySQL melalui jaringan lokal atau internet. phpMyAdmin memfasilitasi sejumlah fungsi MySQL, seperti manajemen basis data, tabel, bidang, relasi, indeks, pengguna, dan izin[17].



Gambar 2.5 *PHPMyAdmin*

2.2.11 UML (*Unified Modeling Language*)

Unified Modeling Language Alat rekayasa perangkat lunak yang disebut Unified Modeling Language, atau disingkat UML,

digunakan untuk menjelaskan cara kerja suatu sistem, cara mengalirnya, tujuannya, dan mekanisme kontrolnya.

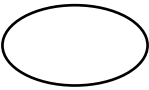
Kasus penggunaan, aktivitas, urutan, dan diagram kelas yang terdapat dalam Empat model UML adalah alat yang sangat efektif untuk menggambarkan desain sistem dalam metodologi desain sistem informasi. Keempat teknik pemodelan ini sering disebut sebagai teknik pemodelan dasar yang berbasis UML. Penerapan pemodelan UML dalam sistem berorientasi objek banyak menggunakan keempat teknik ini.

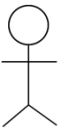


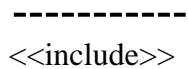
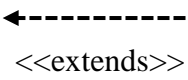
Alat-alat berikut digunakan dalam desain berbasis objek UML:

1. *Use Case* diagram

Diagram *use case* berfungsi sebagai model untuk membantu memahami perilaku potensial dari sistem informasi yang relevan. *Use case* digunakan untuk memverifikasi fungsi sistem informasi mana yang dapat diakses dan siapa yang diizinkan untuk menggunakannya. Tabel di bawah ini mencantumkan simbol-simbol yang digunakan dalam diagram *Use Case*.

Tabel 2. 1 Simbol Use Case Diagram




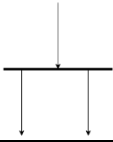
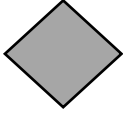
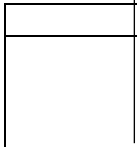
No	Simbol	Keterangan
1		<i>Use Case</i> menjelaskan fungsionalitas yang ditawarkan oleh sistem sebagai unit yang berkomunikasi secara verbal satu sama lain dan dengan akto.

No	Simbol	Keterangan
2		Penggambaran seseorang atau sistem lain yang memulai fungsi sistem target disebut aktor. Untuk mengidentifikasi aktor, tanggung jawab dan pembagian kerja yang terkait dengan peran dalam sistem target harus ditentukan. Orang atau sistem dapat memiliki lebih dari satu peran. Penting untuk dicatat bahwa use case dan aktor berinteraksi.
3		Alih-alih menampilkan fakta, hubungan antara aktor dan kasus penggunaan ditunjukkan oleh garis tanpa panah menunjukkan siapa atau apa yang secara langsung meminta interaksi.
4		Hubungan antara aktor dan kasus penggunaan digambarkan menggunakan panah terbuka, yang juga menunjukkan saat aktor berinteraksi dengan sistem secara pasif.
5		<i>Include</i> , diperlukan dalam kasus penggunaan lain atau dipanggil oleh kasus penggunaan lain, seperti saat fungsi perangkat lunak dipanggil.
6		<i>Extend</i> , Jika kondisi atau persyaratan terpenuhi, itu merupakan perluasan dari kasus penggunaan lainnya.

2. Activity diagram

Diagram aktivitas menunjukkan bagaimana proses bisnis atau sistem beroperasi. Simbol-simbol yang digunakan dalam diagram aktivitas ditunjukkan dalam tabel di bawah.

Tabel 2. 2 Simbol *Activity Diagram*

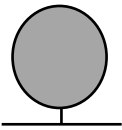
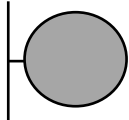
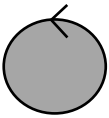

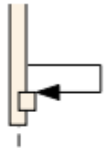
No	Simbol	Keterangan
1		Awal tugas ditunjukkan oleh Titik Awal, yang terletak di sudut kiri atas.
2		<i>End Point</i> , akhir aktivitas.
3		<i>Activities</i> , Buat sketsa aktivitas atau proses bisnis.
4		<i>Fork</i> / Ketika dua tindakan bersamaan digabungkan menjadi satu, hal itu disebut percabangan.
5		<i>Decision Points</i> , menjelaskan kemungkinan dalam pengambilan keputusan, apakah keputusan tersebut akurat atau tidak.
6		<i>Swimlane</i> , diagram aktivitas pembagian untuk menampilkan siapa yang bertanggung jawab atas apa.



3. *Sequence diagram*

Sequence diagram atau diagram urutan menggambarkan perilaku objek dalam suatu kasus penggunaan dengan merinci siklus hidup objek serta pesan yang dikirim dan diterima

antarobjek. Tabel di bawah ini menunjukkan simbol-simbol yang digunakan dalam *Sequence Diagram*.

Tabel 2. 3 Simbol *Sequence Diagram*

No	Simbol	Keterangan
1		<i>Entity Class</i> , adalah bagian dari suatu sistem yang memiliki kumpulan kelas dalam bentuk entitas yang memberikan gambaran awal tentang sistem dan merupakan dasar untuk menyusun suatu basis data.
2		<i>Boundary Class</i> , mencakup sejumlah kelas yang berfungsi sebagai antarmuka atau sarana komunikasi antara sistem dan satu atau beberapa aktor, seperti tampilan formulir cetak dan entri.
3		<i>Control class</i> , Perhitungan dan aturan bisnis yang melibatkan banyak objek adalah contoh objek yang menyertakan logika aplikasi yang tidak bertanggung jawab kepada entitas mana pun.
4		<i>Message</i> , simbol yang digunakan untuk berkomunikasi antar kelas.
5		<i>Recursive</i> , mencirikan pengiriman pesan kepada diri sendiri.

No	Simbol	Keterangan
6		Panjang kotak ini berbanding lurus dengan periode aktivasi suatu operasi, yang diwakili oleh aktivasi.
7		<i>Lifeline</i> , garis titik-titik yang terhubung dengan objek, sepanjang <i>lifeline</i> terdapat <i>activation</i> .

4. Class diagram

Diagram kelas adalah hubungan antara kelas dan penjelasan masing-masing kelas dalam model desain sistem. Diagram kelas juga menggambarkan tugas dan peraturan entitas yang memengaruhi perilaku sistem. Diagram kelas juga menampilkan batasan yang berkaitan dengan objek yang terhubung serta karakteristik dan fungsi kelas.

Elemen umum dari diagram kelas termasuk kelas, asosiasi, generalisasi dan tautan agregasi, atribut, operasi/metode, dan visibilitas akses objek eksternal terhadap suatu tindakan atau atribut.

Tabel 2. 4 Simbol Class Diagram

No	<i>Multiplicity</i>	Keterangan
1	1	Satu dan hanya satu
2	0..*	Boleh tidak ada atau 1 atau lebih
3	1..*	Setidaknya satu

No	<i>Multiplicity</i>	Keterangan
4	0..1	Bisa jadi ada satu, atau tidak ada sama sekali
5	n..n	Batasan antara. Contoh 2..4 mempunyai arti minimal 2 maksimal 4