

BAB II

TINJAUAN PUSTAKA

2.1. Penelitian Terkait

Penelitian yang dilakukan oleh Ismail Hanif Batubara dkk (2022) dalam jurnal penelitiannya yang berjudul “Pemanfaatan Sistem Informasi Dalam Pemesanan Serta Digitalisasi Tiket Bus Berbasis Website”, menyimpulkan bahwa dengan adanya website pemesanan tiket bus sistem *booking* dapat memudahkan masyarakat dalam memesan tiket bus, dan adanya teknologi informasi pembelian atau pemesanan tiket bus menjadi lebih aman[3].

Penelitian yang dilakukan oleh Arinta Azzahra Narwastika dkk (2024) dalam jurnal penelitiannya yang berjudul “Rancang Bangun Sistem Informasi Pemesanan Tiket Bus Berbasis Website Pada Agen Bus PO.AGUNG SEJATI Terminal Sukoharjo”, tahapan-tahapan dalam penelitian ini berupa observasi, wawancara, dan studi kepustakaan untuk memperoleh data yang dibutuhkan, sedangkan untuk pengembangan sistem menggunakan metode Waterfall[4].

Penelitian yang dilakukan oleh Yudhistira Adhitya Pratama dan rekan-rekannya (2022) dalam jurnal penelitiannya yang berjudul “Perancangan Sistem Informasi Pemesanan Tiket Bus pada PT. Putra Pelangi Perkasa”, penelitian yang dibangun berupa sebuah sistem informasi pemesanan tiket bus yang dapat digunakan oleh internal perusahaan serta dapat digunakan oleh pelanggan dalam melakukan pemesanan tiket bus secara *online* sehingga dapat memperlancar proses bisnis[5].

Penelitian yang dilakukan oleh Akbar Wijaya dan Zulhipni Reno Saputra Elsi (2025) dalam jurnalnya penelitiannya yang berjudul “Rancang Bangun Sistem Informasi Pemesanan Tiket Travel Berbasis Web Pada Po Batang Hari Wisata (Studi Kasus : Po Batang Hari Wisata)”, menyatakan bahwa dengan adanya tren serba online dimasa digital membuat pelaku bisnis travel melakukan penjualan tiket/paket wisata melalui berbagai platform salah satunya adalah membuat website sendiri agar terlihat lebih profesional dan mandiri[6].

Penelitian yang dilakukan oleh Akesia Bua dan rekan-rekannya (2021) dalam jurnal berjudul “*Rancang Bangun Sistem Informasi Pemesanan Tiket Bus Batutumonga Berbasis Web*”, mengungkapkan bahwa persaingan antar perusahaan penyedia jasa transportasi semakin meningkat. Untuk menghadapi persaingan tersebut, sejumlah perusahaan telah memanfaatkan teknologi internet sebagai sarana untuk memperkenalkan layanan serta meningkatkan kualitas pelayanan kepada pelanggan[7].

2.2. Landasan Teori

Landasan teori mencakup kumpulan konsep, prinsip, teori, serta sudut pandang yang mendasari dan mendukung proses perancangan serta pengembangan sistem, termasuk di dalamnya perangkat lunak dan bahan yang digunakan dalam pembangunan sistem. Pada bagian ini akan dijelaskan dasar-dasar pemikiran yang menjadi referensi dalam perancangan Sistem Informasi Pemesanan Tiket Bus berbasis *website* pada PO.SUKA DAMAI.

Beberapa di antaranya melibatkan penggunaan perangkat lunak untuk pengolahan data seperti Visual Studio Code, *MySQL*, dan phpMyAdmin.

2.2.1. *Website*

Website adalah sekumpulan halaman situs yang tergabung dalam satu domain atau subdomain, ditempatkan di dalam *World Wide Web* (WWW) di internet. Definisi website juga mencakup halaman yang berisi berbagai data, termasuk teks, gambar, suara, dan elemen lainnya, yang dapat diakses secara daring. *Website* responsif adalah suatu desain *web* yang mampu menyesuaikan tampilan dan fungsionalitasnya secara otomatis dengan berbagai perangkat[8].

2.2.2. *MySQL*

MySQL ialah *multiuser* database yang menggunakan bahasa *structured query language* (SQL). *MySQL* merupakan *software* yang tergolong sebagai DBMS (*Database Management System*) yang bersifat open source. *MySQL* ialah database server yang gratis dengan lisensi *GNU General Public License* (GPL) sehingga dapat anda pakai untuk keperluan pribadi atau komersial tanpa harus membayar lisensi yang ada[9].



Gambar 2. 1 Logo MySQL[10]

2.2.3. Database

Kumpulan data terkait yang disimpan bersama dengan redundansi terkontrol untuk melayani satu atau lebih aplikasi secara optimal. Data disimpan sedemikian rupa sehingga terlepas dari program yang digunakan orang untuk mengakses data. Akses untuk menambahkan data baru, mengedit, dan mengambil data yang ada dari database adalah umum dan terkontrol[11].

Basis Data (Database) ialah kumpulan tabel, hubungan, dan elemen lain yang berkaitan dengan penyimpanan data. Menurut CJ. Date, database adalah koneksi data operasional yang tersimpan dan digunakan oleh sistem aplikasi atau organisasi. Menurut Gordon C. Everest, database adalah kumpulan data yang terorganisir secara mekanis, terbagi, terdefinisi secara formal, dan dikontrol secara terpusat dalam suatu organisasi[12].

2.2.4. Laravel

Laravel merupakan salah satu kerangka kerja pengembangan aplikasi web yang paling populer dan kuat dalam dunia PHP. Dirancang untuk memudahkan proses pengembangan aplikasi, Laravel menawarkan beragam fitur dan alat yang memungkinkan para pengembang untuk menghasilkan aplikasi *web* yang efisien, aman, dan mudah dikelola. Dengan sintaksis yang ekspresif, dokumentasi yang kaya, serta dukungan komunitas yang luas, Laravel menjadi pilihan utama bagi banyak pengembang yang ingin mempercepat

pengembangan aplikasi *web* mereka[13].



Gambar 2. 2 Logo Laravel[14]

2.2.5. phpMyAdmin

phpMyAdmin merupakan alat manajemen basis data open source yang ditulis dalam PHP dan digunakan untuk mengelola database *MySQL* atau *MariaDB* melalui antarmuka berbasis *web*. phpMyAdmin memungkinkan pengguna untuk melakukan operasi basis data seperti membuat, mengubah, dan menghapus tabel, serta mengelola pengguna dan hak akses. (phpMyAdmin *Development Team*, 2023) phpMyAdmin sering digunakan di lingkungan pengembangan web, terutama dalam stack LAMP (Linux, Apache, *MySQL*, PHP) atau WAMP (Windows, Apache, *MySQL*, PHP), karena kemudahan penggunaannya dan dukungannya terhadap berbagai sistem manajemen basis data[15].



Gambar 2. 3 Logo *phpMyAdmin*[16]

2.2.6. XAMPP

XAMPP merupakan salah satu server yang paling populer

digunakan untuk belajar PHP secara mandiri, khususnya bagi pemula. Aplikasi ini memiliki fitur yang cukup lengkap namun tetap mudah dioperasikan programmer yang baru mulai belajar PHP. Selain tersedia secara gratis, XAMPP juga menyediakan kemudahan dalam penggunaannya. Untuk menjalankannya, pengguna hanya perlu mengaktifkan modul Apache yang terdapat di dalam XAMPP[17].

2.2.7. Unified Modeling Language (UML)

UML merupakan suatu metode dalam pemodelan secara visual yang digunakan sebagai sarana perancangan sistem berorientasi objek. UML juga didefinisikan sebagai suatu bahasa standar visualisasi, perancangan, dan pendokumentasian sistem, atau dikenal juga sebagai bahasa standar penulisan blueprint sebuah software. UML diharapkan dapat mempermudah pengembangan perangkat lunak serta memenuhi semua kebutuhan pengguna dengan efektif, lengkap dan tepat[18].

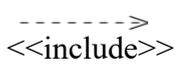
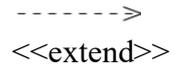
Beberapa jenis diagram UML yang umum digunakan dalam proses pengembangan sistem antara lain:

1) *Use Case Diagram*

Use Case Diagram adalah pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use Case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut[19]. Simbol-simbol yang digunakan dalam *Use Case*

Diagram yaitu:

Tabel 2. 1 *Use Case Diagram*

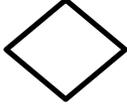
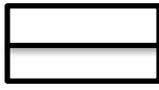
No	Simbol	Keterangan
1.		<i>Use Case</i> , abstraksi dan interaksi antara sistem dan aktor.
2.		Aktor, menggambarkan manusia atau suatu hal yang berinteraksi dengan sistem.
3.		<i>Association</i> , menghubungkan antara aktor dengan <i>use case</i> dalam sebuah diagram
4.		Generalisasi, menunjukkan aktor spesifik yang mewarisi peran aktor umum dalam <i>use case</i> .
5.		<i>Include</i> , menunjukkan <i>use case</i> yang menyertakan seluruh fungsi dari <i>use case</i> lain.
6.		<i>Extend</i> , menunjukkan bahwa suatu <i>use case</i> merupakan tambahan fungsional dari <i>use case</i> lainnya jika suatu kondisi terpenuhi.

2) *Class Diagram*

Class Diagram adalah sejenis model struktural yang bersifat statis yang digunakan untuk menggambarkan struktur sebuah sistem[20]. Simbol-simbol yang digunakan dalam *Class Diagram* yaitu:

Tabel 2. 2 *Class Diagram*

No	Simbol	Keterangan
1.		<i>Generalization</i> , Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).

No	Simbol	Keterangan
2.		<i>Nary Association</i> , Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3.		<i>Realization</i> , Operasi yang benar-benar dilakukan oleh suatu objek.
4.		<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.
5.		<i>Association</i> , Apa yang menghubungkan antara objek satu dengan objek lainnya.
6.		<i>Class</i> , Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.

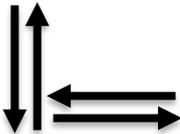
3) Activity Diagram

Activity Diagram menggambarkan aktivitas apa yang dapat dilakukan oleh sistem bukan apa yang dilakukan aktor[21].

Simbol *Activity Diagram* yaitu:

Tabel 2. 3 Activity Diagram

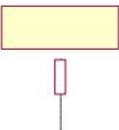
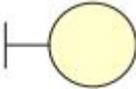
No	Simbol	Keterangan
1.		<i>Activity</i> , Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain.
2.		<i>Action</i> , State dari sistem yang mencerminkan eksekusi dari suatu aksi.
3.		<i>Start Point</i> , Diletakkan pada pojok kiri atas dan merupakan awal aktivitas.

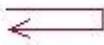
No	Simbol	Keterangan
4.		<i>End Point</i> , Akhir aktivitas.
5.		<i>Decision</i> , Digunakan untuk menggambarkan suatu keputusan / tindakan yang harus diambil pada kondisi tertentu.
6.		<i>Line Connector</i> , Digunakan untuk menghubungkan satu simbol dengan simbol lainnya.

4) *Sequence Diagram*

Sequence Diagram adalah diagram yang menjelaskan interaksi objek berdasarkan urutan waktu[22]. Simbol-simbol yang digunakan dalam *Sequence Diagram* yaitu:

Tabel 2. 4 *Sequence Diagram*

No	Simbol	Keterangan
1.		<i>Actor</i> , Merepresentasikan entitas yang berada diluar sistem dan berinteraksi dengan sistem.
2.		<i>Lifeline</i> , Menghubungkan objek selama <i>sequence</i> (<i>message</i> dikirim atau diterima dan aktifitasnya).
3.		<i>General</i> , Merepresentasikan entitas tunggal dalam <i>sequence diagram</i> .
4.		<i>Boundary</i> , Berupa tepi dari sistem, seperti <i>user interface</i> atau suatu alat yang berinteraksi dengan sistem yang lain.

No	Simbol	Keterangan
5.		<i>Control</i> , Elemen mengatur aliran dari informasi untuk sebuah scenario. Objek ini umumnya mengatur perilaku dan perilaku bisnis.
6.		<i>Entitas</i> , Elemen yang bertanggung jawab menyimpan data atau informasi. Berupa beans atau model objek.
7.		<i>Activation</i> , Suatu titik dimana objek mulai berpartisipasi dalam sebuah <i>sequence</i> yang menunjukkan kapan sebuah objek mengirim atau menerima objek.
8.		<i>Message Entry</i> , Berfungsi untuk menggambarkan pesan/hubungan antar objek yang menunjukkan urutan kejadian yang terjadi.
9.		<i>Message to Self</i> , Simbol ini menggambarkan pesan/hubungan objek itu sendiri, yang menunjukkan urutan kejadian yang terjadi.
10.		<i>Message Return</i> , Menggambarkan hasil dari pengiriman <i>message</i> dan digambarkan dengan arah dari kanan ke kiri.

2.2.8. Visual Studio Code

Visual Studio Code adalah salah satu IDE yang dapat digunakan oleh pemula dengan fitur-fitur yang lengkap dan memudahkan pengembang dalam membuat aplikasi. Visual Studio Code ini memiliki dukungan untuk JavaScript, TypeScript, Node.Js, HTML, dan CSS. Teks editor Visual Studio Code dibuat oleh Microsoft, teks editor ini menawarkan berbagai macam kemampuan, termasuk *code completion*, *syntax highlighting*, *debugging*, dan *git control*[23].