

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Penelitian yang dilakukan oleh **Utami, A. F. D., Ridwan, M., Khakim, A., dan Grafvera, E. D. (2023)** dengan judul “*Pengembangan Sistem Informasi Reservasi Berbasis Website Menggunakan Metode Waterfall*” bertujuan untuk mengembangkan sistem reservasi berbasis *Website* guna meningkatkan efisiensi proses pemesanan. Metode yang digunakan adalah *Waterfall* yang meliputi analisis, desain, implementasi, pengujian, dan pemeliharaan. Hasil penelitian menunjukkan bahwa sistem dapat mempermudah pengguna dalam melakukan reservasi serta membantu pengelola dalam mengelola data secara terstruktur..[6]

Penelitian yang dilakukan oleh **Oktapiah dan Hasti (2021)** dengan judul “*Sistem Informasi Reservasi Paket Wisata Berbasis Web*” bertujuan untuk mengatasi permasalahan reservasi yang masih dilakukan secara manual. Permasalahan yang dihadapi meliputi kesalahan pencatatan, lambatnya pencarian data, dan keterlambatan pembuatan laporan. Hasil penelitian menunjukkan bahwa sistem berbasis *Website* mampu meningkatkan kecepatan pengolahan data dan kualitas pelayanan..[7]

Penelitian yang dilakukan oleh **Anharudin, Susandi, dan Saefudin (2021)** dengan judul “*Rancang Bangun Sistem Informasi Reservasi Kamar Homestay Berbasis Web*” bertujuan untuk membangun sistem reservasi

online. Sistem tersebut dapat membantu proses reservasi, pencatatan data, pembayaran, serta meningkatkan kualitas layanan kepada pengguna.[8]

Penelitian lain yang dilakukan oleh **Fery Purnama, Muhammad Ikhsan, dan Tamrin Syah (2021)** dengan judul *“Implementasi Sistem Informasi Reservasi Hotel Syariah di Kota Jambi Berbasis Website”* yang bertujuan untuk membangun sistem reservasi berbasis *Web*. Hasil penelitian menunjukkan bahwa sistem mampu meningkatkan akurasi data, mempermudah pengelolaan reservasi, serta mempercepat pembuatan laporan. Berdasarkan beberapa penelitian terdahulu tersebut, dapat disimpulkan bahwa sistem informasi reservasi berbasis *Website* memiliki peran penting dalam membantu proses pencatatan, pengelolaan data, pencarian data, dan pembuatan laporan secara lebih efektif. Penelitian-penelitian tersebut memiliki keterkaitan dengan penelitian ini karena sama-sama membahas sistem reservasi berbasis *Website*. Perbedaannya, penelitian ini difokuskan pada **reservasi tamu di PT. BSR Indonesia**, yaitu untuk membantu proses pendataan tamu, pengajuan atau pencatatan reservasi kunjungan, pengelolaan jadwal tamu, serta pembuatan laporan reservasi tamu secara terstruktur..[9]

2.2 Landasan Teoretis

2.2.1 Sistem Informasi

Sistem informasi merupakan suatu kombinasi terorganisir dari manusia, perangkat keras, perangkat lunak, jaringan komunikasi, dan

prosedur yang digunakan untuk mengumpulkan, menyimpan, mengolah, serta menyebarkan informasi guna mendukung proses pengambilan keputusan dan pengendalian dalam organisasi. Sistem informasi berfungsi untuk meningkatkan efisiensi operasional, efektivitas layanan, dan kualitas pengelolaan data dalam berbagai aktivitas digital.[10]

2.2.2 Sistem Reservasi

Sistem reservasi adalah sistem informasi yang dirancang untuk mengelola proses pemesanan layanan atau fasilitas secara digital, termasuk pengaturan jadwal, ketersediaan fasilitas, serta pencatatan data pemesanan. Sistem ini mendukung otomatisasi alur reservasi sehingga proses pemesanan dapat dilakukan secara lebih cepat, akurat, dan terdokumentasi dengan baik, serta dapat diakses secara *real-time* oleh pengguna internal maupun pengelola organisasi. Penerapan sistem reservasi berbasis *Website* terbukti mampu meningkatkan efisiensi operasional dan mengurangi risiko kesalahan pencatatan dibandingkan dengan proses non-digital.[11]

2.2.3 Payment Gateway (Midtrans)

Payment gateway merupakan teknologi jembatan yang menghubungkan situs *Web* pedagang (*merchant*) dengan lembaga keuangan untuk memproses dan mengotorisasi transaksi pembayaran daring (*online*). Fungsi utamanya adalah melegalkan informasi transaksi, mengamankan data pembayaran melalui enkripsi, serta

mendeteksi potensi penipuan (*fraud detection*). Dalam penelitian ini, *Payment Gateway* yang digunakan adalah *Midtrans*. *Midtrans* merupakan penyedia layanan gerbang pembayaran yang memfasilitasi berbagai metode transaksi seperti transfer bank (*Virtual Account*), kartu kredit, hingga dompet digital (*E-Wallet*). *Midtrans* menyediakan fitur lingkungan pengujian yang disebut *Sandbox*. Fitur ini memungkinkan pengembang melakukan simulasi transaksi pembayaran menggunakan data tiruan (*dummy*) untuk memastikan fungsionalitas sistem berjalan sesuai spesifikasi sebelum diterapkan pada transaksi riil [16]

2.2.4 Ruang Tamu

Kegiatan PT BSR Indonesia menginisiasi pengembangan "Sistem Informasi Reservasi Tamu Berbasis *Website*" untuk mengoptimalkan pemanfaatan ruang kolaboratif yang aman, nyaman, dan profesional. Sistem ini dibuat untuk mengatasi masalah aksesibilitas dan pengelolaan ruang dengan memberikan solusi digital yang memungkinkan pengguna melakukan pemesanan secara langsung, jelas, dan cepat, sehingga bisa mengurangi hambatan terkait ketidakpastian lokasi serta jadwal yang saling tumpang tindih. Melalui integrasi teknologi informasi ini, Penelitian ini bertujuan untuk merancang dan membangun sistem informasi reservasi tamu berbasis *Website* yang digunakan untuk membantu proses pendaftaran tamu serta penggunaan fasilitas internal di PT. BSR Indonesia.[17]

2.2.5 *Web dan Teknologi Web*

Website adalah kumpulan halaman digital yang terhubung satu sama lain dan bisa dilihat menggunakan *browser* melalui protokol *HTTP* atau *HTTPS*. Situs *Web* berperan sebagai alat untuk menampilkan informasi, berkomunikasi, dan melakukan transaksi melalui internet. Dalam sistem informasi organisasi atau perusahaan, *Website* digunakan sebagai platform utama untuk mengelola layanan digital, seperti menyediakan informasi, mengelola data, serta melakukan proses reservasi dan administrasi secara online. Pemanfaatan *Website* dalam sistem informasi memungkinkan pengguna internal dan pengelola organisasi untuk mengakses layanan secara lebih mudah, cepat, dan terintegrasi.[12]

2.2.6 *Visual Studio Code*

Visual Studio Code adalah sebuah teks editor ringan dan handal yang dibuat oleh *Microsoft* untuk sistem operasi *multiplatform*, artinya tersedia juga untuk versi *Linux*, *Mac*, dan *Windows*. Teks editor ini secara langsung mendukung bahasa pemrograman *JavaScript*, *TypeScript*, dan *Node.js*, serta bahasa pemrograman lainnya dengan bantuan plugin yang dapat dipasang melalui *marketplace Visual Studio Code* seperti *C++*, *C#*, *Python*, *Go*, *Java*, dan *PHP* (*Microsoft*, 2023; Nugroho, 2021).[13]



Gambar 2. 1 *Visual Studio Code*

2.2.7 **PHP**

PHP (Hypertext Preprocessor) adalah bahasa pemrograman *server-side* yang sering digunakan dalam pengembangan aplikasi *Web* dinamis. Keunggulan *PHP* terletak pada kemudahan integrasinya dengan berbagai sistem basis data, seperti *MySQL* dan *PostgreSQL*, serta kompatibilitas yang tinggi dengan server *Web* seperti *Apache*. Selain itu, *PHP* juga mendukung berbagai *framework* modern seperti *Laravel* dan *CodeIgniter* yang mempercepat proses pengembangan aplikasi *Web* (Welling & Thomson, 2003).[14]



Gambar 2. 2 *PHP My Admin*

2.2.8 *Laravel*

Laravel adalah *framework PHP open-source* yang mengadopsi pola arsitektur MVC (*Model-View-Controller*) untuk mempermudah pengembangan aplikasi *Web*. *Laravel* menyediakan berbagai fitur seperti *routing*, *authentication*, *templating*, serta *ORM (Eloquent)* yang memudahkan *developer* dalam membangun sistem yang aman, cepat, dan terstruktur. *Framework* ini juga mendukung pengembangan berbasis modular dan berorientasi objek, sehingga cocok digunakan untuk aplikasi skala menengah hingga besar (Syafitri, A., Angraeni, A., & Wibowo, A. (2025).[15])



Gambar 2. 3 *Laravel*

2.2.9 *MySQL Database*

MySQL adalah *database management system (DBMS)* berbasis *SQL* yang digunakan untuk menyimpan, mengelola, dan memproses data secara terstruktur. *MySQL* mendukung operasi *CRUD*, relasi

antar tabel, transaksi, serta keamanan data sehingga sangat cocok digunakan untuk aplikasi reservasi yang membutuhkan penyimpanan data tiket, pengguna, transaksi, dan log aktivitas secara terpusat.



Gambar 2. 4 *MySQL*

2.2.10 *Laragon*

Laragon adalah lingkungan pengembangan lokal (*local development environment*) yang digunakan untuk menjalankan aplikasi *Web* berbasis *PHP*. *Laragon* menyediakan paket server seperti *Apache/Nginx*, *MySQL*, *PHP*, serta mendukung instalasi *framework* seperti *Laravel*. *Laragon* memberikan kemudahan dalam pengembangan karena ringan, cepat, stabil, dan mudah dikonfigurasi.



Gambar 2. 5 Laragon

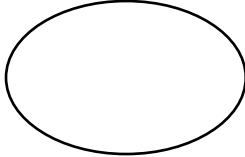
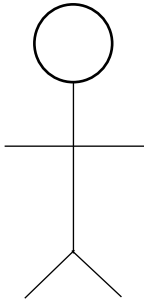
2.2.11 ***UML (Unified Modeling Language)***

Unified Modeling Language (UML) adalah standar pemodelan visual yang digunakan untuk mendesain dan mendokumentasikan sistem perangkat lunak. *UML* digunakan untuk membuat *Diagram* seperti *Use Case Diagram*, *Class Diagram*, *Sequence Diagram*, dan *Activity Diagram*, yang membantu dalam memahami struktur dan alur kerja sistem sebelum pengembangan dilakukan. Penggunaan *UML* dalam penelitian ini bertujuan untuk menyusun perencanaan sistem secara lebih sistematis dan mendalam sebelum tahap implementasi. Terdapat beberapa *Diagram UML* yang sering digunakan dalam pengembangan sebuah sistem yaitu *Use Case Diagram*, *Class Diagram*, *Sequence Diagram*, dan *Activity Diagram* [16].

1. *Use Case Diagram*

Use Case Diagram salah satu jenis *Diagram* yang termasuk dalam *Unified Modeling Language (UML)* yang digunakan untuk menunjukkan fungsi-fungsi utama suatu sistem yang didasarkan pada sudut pandang pengguna (aktor). *Diagram* ini menunjukkan bagaimana aktor berinteraksi dengan sistem untuk mencapai tujuan tertentu. Oleh karena itu, *Diagram* ini digunakan untuk membantu memahami kebutuhan sistem secara umum tanpa menampilkan detail alur kerja. *Diagram* ini juga bersifat statis karena tidak menggambarkan urutan waktu atau proses secara rinci.

Tabel 2. 1 *Use Case Diagram*

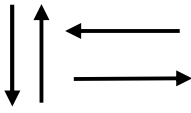
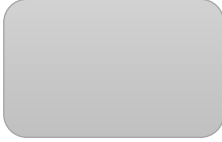
No	Simbol	Keterangan
1		<i>Use Case</i> merupakan penjelasan rinci merupakan penjelasan rinci dari sebuah <i>Use Case</i> yang menggambarkan fungsionalitas yang disediakan oleh sistem.
2		<i>Actor</i> yang berinteraksi dengan sistem untuk menjalankan fungsi tertentu.

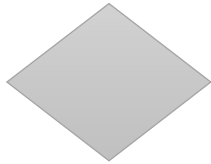
No	Simbol	Keterangan
3		<i>Association</i> garis yang menghubungkan aktor dengan <i>Use Case</i> untuk menunjukkan adanya interaksi antara keduanya.
4	 << <i>include</i> >>	<i>Include</i> , merupakan gambaran jika <i>Use Case</i> dipanggil <i>Usecase</i> lain.
5	 << <i>Extends</i> >>	<i>Extends</i> , merupakan Gambaran jika memperluas <i>Usecase</i> target.
6		<i>Dependency</i> merupakan garis panah yang menunjukan jika <i>actor</i> berinteraksi secara pasif.

2. Activity Diagram

Activity Diagram yang digunakan untuk menggambarkan alur kerja suatu sistem. *Diagram* ini menunjukkan perilaku dinamis sistem melalui aliran control dari satu aktivitas lainnya.

Tabel 2. 2 *Activity Diagram*

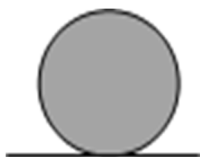
No	Simbol	Keterangan
1		<i>End Point</i> atau <i>Final Node</i> , merupakan Gambaran akhir dari suatu aktivitas.
2		<i>Start Point</i> , merupakan awal dari suatu aktivitas yang peletakkannya pada pojok kiri atas.
3		<i>Line Connector</i> , digunakan untuk menghubungkan satu symbol dengan symbol lainnya.
4		<i>Activity</i> , merupakan Gambaran dari suatu proses.
5		<i>Action</i> , State dari sistem yang mencerminkan eksekusi dari suatu aksi.

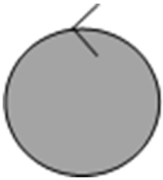
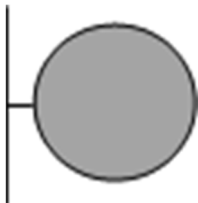
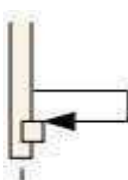
No	Simbol	Keterangan
6		<i>Decision</i> , merupakan pilihan pengambilan suatu Keputusan <i>false or true</i> .

3. *Sequence Diagram*

Sequence Diagram adalah *Diagram* yang menunjukkan interaksi antar objek dalam sistem seiring waktu. *Diagram* ini menunjukkan bagaimana pesan dikirim dan diterima antar objek untuk menjalankan suatu proses atau skenario tertentu. *Sequence Diagram* biasanya digunakan untuk menjelaskan langkah-langkah yang terjadi sebagai respons terhadap suatu event hingga menghasilkan output. *Diagram* ini menampilkan urutan komunikasi antar aktor dan sistem secara kronologis. Untuk simbol *Diagram* urutan bisa dilihat pada table 3.

Tabel 2. 3 *Sequence Diagram*

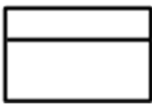
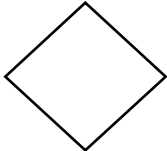
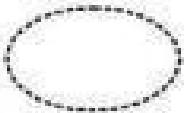
No	Simbol	Keterangan
1		<i>Entity Class</i> merupakan bagian sistem yang membentuk deskripsi awal sistem dan berisi Kumpulan kelas dalam bentuk.

No	Simbol	Keterangan
2		<i>Control Class</i> merupakan Gambaran penghubung antara <i>Boundary Class</i> dengan suatu <i>table</i> .
3		<i>Boundary Class</i> merupakan Gambaran dari penggambaran from.
4		Pesan atau <i>message</i> , menunjukkan pengiriman pesan antar <i>Class</i>
5		Self Message, menunjukkan pengiriman suatu pesan yang akan dikirim ke objek itu sendiri
6		Activation menggambarkan suatu objek yang melakukan sebuah aksi/eksekusi operasi.
7		<i>Lifeline</i> garis titik yang terhubung ke objek disepanjang garis <i>lifeline</i> memiliki aktivitas.

4. *Class Diagram*

Diagram Class merupakan *Diagram* yang bersifat statis, dalam *Diagram* ini memperlihatkan himpunan kelas, antar muka, serta relasi. Berikut simbol dari *Class Diagram* yang dapat dilihat pada table 4.

Tabel 2. 4 *Class Diagram*

No	Simbol	Keterangan
1		<i>Generallization</i> , merupakan Dimana objek <i>descendent</i> membagikan perilaku dan struktur data objek induknya.
2		<i>Class</i> adalah kumpulan objek yang saling berbagi
3		<i>Nary Assocation</i> , digunakan untuk asosiasi terhindar dengan objek lainnya.
4		<i>Collaboration</i> merupakan deskripsi urutan aksi yang menampilkan interaksi sistem.

No	Simbol	Keterangan
5		Merupakan operasi yang <i>valid</i> dilakukan oleh suatu objek.
6		Merupakan garis panah yang menunjukkan jika actor berinteraksi secara pasif.
7		<i>Association</i> merupakan garis yang menghubungkan objek satu dengan objek lainnya.