

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Beberapa penelitian terkait diuraikan sebagai berikut:

Penelitian yang dilakukan oleh Mia Fitriawati dan Damaiyanti Ema Novelina Sihombing dengan judul “Sistem Informasi Pemesanan Wisata dan Ekspedisi Berbasis Web Pada Kampung Batu Malakasari” penelitian yang dilakukannya bertujuan untuk merancang dan membangun sistem informasi yang berguna untuk meningkatkan proses pemasaran seperti promosi, pemesanan dan pengelolaan data nya yang berbasis web. Terdapat persamaan dari penelitian yang dilakukan oleh Mia Fitriawati dan Damaiyanti Ema Novelina Sihombing yaitu dalam proses yang sedang berjalan dimana pemesanan paket wisata masih secara manual dan untuk promosinya masih menggunakan media internet ataupun brosur yang mneghamburkan kertas. Terdapat juga perbedaan yang terlihat dari proses bisnisnya yang hanya membahas mengenai pemesanan paket wisata yang pengelolaannya yang dicatat dalam buku kunjungan secara manual, lalu pembatalan pengunjung, dan proses pembuatan laporan [1].

Penelitian yang dilakukan oleh Yusuf Nugraha dengan judul “Sistem Informasi Pemesanan Paket Wisata Pada Bintang Car Tour dan Travel Bandung” penelitiannya bertujuan untuk membangun suatu sistem informasi yang dapat meningkatkan kualitas dan kinerja pada sistem transaksi pemesanan paket wisata. Persamaan dari penelitian yang dilakukan oleh Yusuf Nugraha yaitu pada sistem pemesanan paket wisata yang masih dilakukan dengan pencatatan data

pemesanan pelanggan secara manual yang menggunakan ms.excel untuk merekap datanya sehingga butuh ketelitian. Perbedaan yang terdapat pada penelitian ini pada proses bisnisnya yang ruang lingkup nya hanya melakukan proses pelayanan pemesanan paket wisata dan perancangan sistem informasi nya berbasis desktop menggunakan aplikasi netbeans, mysql dan javascript [2].

Penelitian yang dilakukan oleh Julianto Simatupang dan Setiawan Sianturi dengan judul “Perancangan Sistem Informasi Pemesanan Tiket Bus Pada Po. Handoyo Berbasis Online” penelitiannya bertujuan untuk membangun sistem informasi pemesanan tiket berbasis online, dengan menerapkan siklus hidup pengembangan sistem yaitu metode *waterfall*. Dengan adanya sistem informasi pemesanan tiket bus ini diharapkan dapat menyelesaikan masalah-masalah yang terjadi dan dapat meningkatkan efisiensi dan efektifitas agar pelayanannya menjadi lebih baik dibanding dengan sistem yang terdahulu [3].

Penelitian yang dilakukan oleh Sede, dkk dengan judul “Rancang Bangun Aplikasi Pemesanan Tiket Online Kapal Laut Berbasis Android” penelitiannya bertujuan untuk merancang aplikasi pemesanan tiket online kapal laut berbasis android dengan menggunakan metodologi DAD (*Disciplined Agile Delivery*) dan Memberikan informasi bagi masyarakat yang akan melakukan perjalanan menggunakan transportasi kapal laut, baik informasi mengenai kapal, jadwal pelayaran, pemesanan tiket ke masing-masing tujuan. Rancang bangun pemesanan tiket online kapal laut berbasis android telah berhasil di bangun dan di

implementasikan dengan fitur aplikasi jadwal kapal, cek tiket, booking tiket dan mendaftar sebagai member [4].

Penelitian yang dilakukan oleh Muhammad Fajri Ilmanda dan Tony Sugiarto dengan judul “Aplikasi Pemesanan Tiket Pariwisata Berbasis Website Pada Leuwi Pangaduan” penelitiannya bertujuan untuk mengembangkan beberapa aspek, salah satunya adalah dengan penjualan tiket secara online untuk memudahkan sistem pemesanan tiket saat ini. Metodologi penelitian ini dimulai dari mengumpulkan data terlebih dahulu. Pada tahap pengumpulan data dilakukan melalui wawancara dan studi pustaka. Kemudian pada tahap pengembangan metode Software Development Life Cycle (SDLC) yaitu dengan model Waterfall. Dengan adanya aplikasi ini diharapkan dapat membantu wisatawan untuk dapat melakukan pemesanan tiket kapanpun dan dimanapun sehingga efektif dan efisien [5].

2.2 Landasan Teori

2.2.1 Perancangan

Menurut Sinta & Listiana [1], perancangan merupakan unsur suatu tahapan dari keseluruhan pembangunan sistem komputerisasi. Perancangan sistem untuk pengembangan sistem biasanya memerlukan jangka waktu yang lebih lama dari perancangan masalah pada umumnya, serta memberikan gambaran bagi *user* atau manajemen secara rinci, juga menggambarkan sistem yang diusulkan dari sistem yang sedang dibangun.

2.2.2 Sistem Informasi

Sistem merupakan sekumpulan komponen atau jaringan yang saling berkaitan dan bekerjasama sehingga membentuk jaringan kerja yang saling berkaitan untuk mencapai tujuan tertentu. Informasi merupakan data yang diolah menjadi suatu bentuk sebagai penunjang keputusan.

Sistem informasi adalah sekumpulan prosedur atau komponen-komponen yang saling berhubungan dan bekerjasama dalam mengumpulkan, mengolah, menyimpan dan mendistribusikan data menjadi informasi untuk mendukung proses pengambilan keputusan pada suatu organisasi untuk mengendalikan dan koordinasi [2].

Sistem informasi adalah sekumpulan komponen yang digunakan untuk mengelola data sehingga menjadi sebuah informasi untuk mengambil sebuah keputusan.

2.2.3 Pemesanan

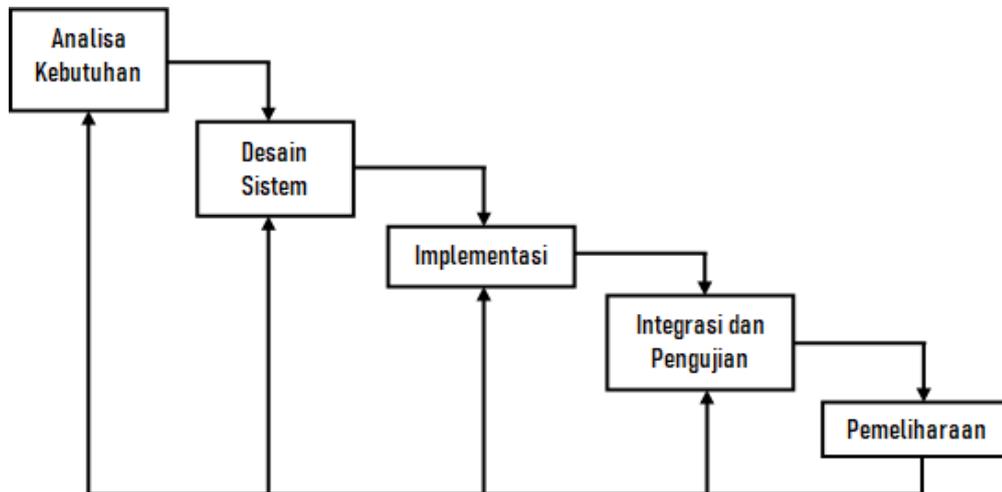
Menurut penulis pemesanan adalah suatu kegiatan yang dilakukan berupa permintaan untuk memperoleh barang atau jasa.

2.2.4 Metode Pengembangan Sistem

Metode Waterfall adalah salah satu metode bersifat natural yang digunakan dalam perancangan dan pengembangan sistem perangkat lunak yang sistematis. Disebut *waterfall* karena tahapan-tahapannya yang berurutan.

Metode *waterfall* adalah metode yang melakukan pendekatan terhadap pengembangan terhadap perangkat lunak secara sistematis atau berurutan dengan

beberapa tahapan yaitu Dari *analisis sistem, desain, coding, testing* dan *maintenance* [3].



Gambar 2.1 Metode Waterfall

1. Analisis Kebutuhan

Tahapan ini diperlukan dalam pengembangan sistem untuk mendapatkan informasi kebutuhan dan dilakukan analisis kemudian diidentifikasi kebutuhan sistem yang akan dibangun. Tahapan ini dilakukan untuk memenuhi kebutuhan sistem yang diharapkan oleh pengguna. Informasi kebutuhan ini dapat diperoleh melalui survei langsung lokasi, diskusi dan wawancara agar data yang dibutuhkan lebih tepat.

2. Desain Sistem

Pada tahapan ini proses desain dilakukan untuk menggambarkan tampilan pada sistem yang akan dibangun. Hal ini dilakukan untuk

membantu dalam mendefinisikan keseluruhan arsitektur sistem dan *hardware* yang akan digunakan.

3. Implementasi

Pada tahapan ini untuk mengimplementasikan sistem yang telah didesain akan dikerjakan dengan menggunakan bahasa pemrograman yang telah ditentukan.

4. Interaksi dan Pengujian

Pada tahapan pengujian ini dilakukan untuk mengetahui apakah sistem yang dirancang sudah sesuai dengan yang diinginkan serta masih adakah kesalahan pada sistem atau tidak.

5. Pemeliharaan

Pada tahapan yang terakhir ini sistem yang telah melalui proses pengujian dan dapat dijalankan serta perlunya dilakukan pemeliharaan sistem. Pemeliharaan pengembangan pada suatu sistem, hal ini dilakukan apabila adanya perubahan seperti pengantian operator sistem atau penambahan file.

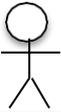
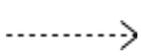
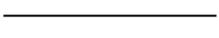
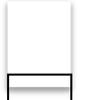
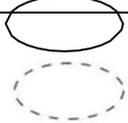
2.2.6 UML (*Unified Modeling Language*)

UML adalah sekumpulan notasi grafis yang membantu menggambarkan pendeskripsia informasi, antara pengguna, desain sistem perangkat lunak, dan manajemen dengan efisien khususnya sistem yang dibangun menggunakan pemrograman berbasis objek [5].

1. Use Case Diagram

Use Case merupakan suatu simbol yang digunakan dalam pengembangan sistem, juga merupakan sebuah kegiatan yang berkaitan antara aktor dan sistem yang berkaitan. Dalam perancangan sistem simbol *use case diagram* digunakan untuk menggambarkan hubungan antara pengguna dan system [4].

Tabel 2.1 Simbol Use Case Diagram

Simbol	Nama	Keterangan
	<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case diagram</i>
	<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang tidak mandiri
	<i>generalization</i>	Hubungan dalam objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada diatas objek induk (<i>ancestor</i>).
	<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara eksplisit
	<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan
	<i>Association</i>	Yang menghubungkan antara objek yang satu dengan objek yang lain
	<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas
	<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menampilkan suatu hasil yang terukur bagi suatu aktor.
	<i>Collaboration</i>	Interaksi aturan-aturan dari elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah elemen-elemennya (<i>sinergis</i>)
	<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

2. Activity Diagram

Activity diagram atau dalam bahasa Indonesia diagram aktivitas menggambarkan rangkaian aliran dari aktivitas. Diagram aktivitas merupakan pengembangan dari *use case*, diagram untuk memodelkan menu-menu atau proses yang digambarkan secara vertikal yang ada pada sistem [4]

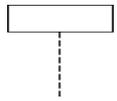
Tabel 2.2 Simbol *Activity Diagram*

Simbol	Nama	Keterangan
	<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antara muka saling berinteraksi satu sama lain
	<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
	<i>Activity Final Node</i>	Bagaimana objek dibentuk dan diakhiri.
	<i>Decision</i>	Digunakan untuk menggambarkan suatu keputusan yang harus diambil pada kondisi tertentu.
	<i>Line Connector</i>	Digunakan untuk menghubungkan suatu simbol dengan simbol lainnya.
	<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi.

3. Sequence Diagram

Sequence diagram digunakan untuk menggambarkan interaksi antara objek, juga untuk menunjukkan pesan yang dikirim antara objek dan juga rangkaian interaksi antara objek pada titik tertentu dalam sistem [4].

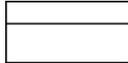
Tabel 2.3 *Sequence Diagram*

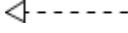
No	Simbol	Nama	Keterangan
1		<i>LifeLine</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
2		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi
3		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi

4. Class Diagram

Class Diagram yang menggambarkan keadaan suatu sistem, sistem yang dimodelkan dengan berbagai komponen untuk menggambarkan struktur dan deskripsi *class* [5].

Tabel 2.4 *Class Diagram*

No	Simbol	Nama	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>).
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 yang menghasilkan suatu hasil yang terukur bagi suatu actor

No	Simbol	Nama	Keterangan
5		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempegaruhi elemen yang bergantung padanya elemen yang tidak mandiri
7		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

2.2.7 ERD (Entity Relationship Diagram)

ERD merupakan sebuah model yang digunakan untuk menyusun sebuah database agar agar dapat menggambarkan hubungan antara database suatu sistem yang dirancang[6].

Tabel 2.5 Simbol *Entity Relationship Diagram* (ERD)

No	Simbol	Nama	Keterangan
1		entitas	Entitas adalah suatu objek yang dapat diidentifikasi dalam lingkungan pemakai
2		relasi	Relasi yang menunjukkan adanya hubungan antara entitas yang satu dengan entitas yang lain
3		atribut	Sibol ini berfungsi untuk mendeskripsikan karakter entitas (atribut yang berfungsi sebagai key diberi garis bawah).
4		garis	Garis sebagai penghubung atara relasi dan entitas atau relasi dan entitas dengan atribut

2.2.8 Metode Pengujian Sistem

Pengujian sistem yang digunakan dalam penelitian ini untuk memastikan sistem yang telah dirancang berjalan sesuai dengan perancangan dan kebutuhan yang diinginkan. Metode pengujian sistem yang digunakan yaitu [5]:

1. *Blackbox Testing*

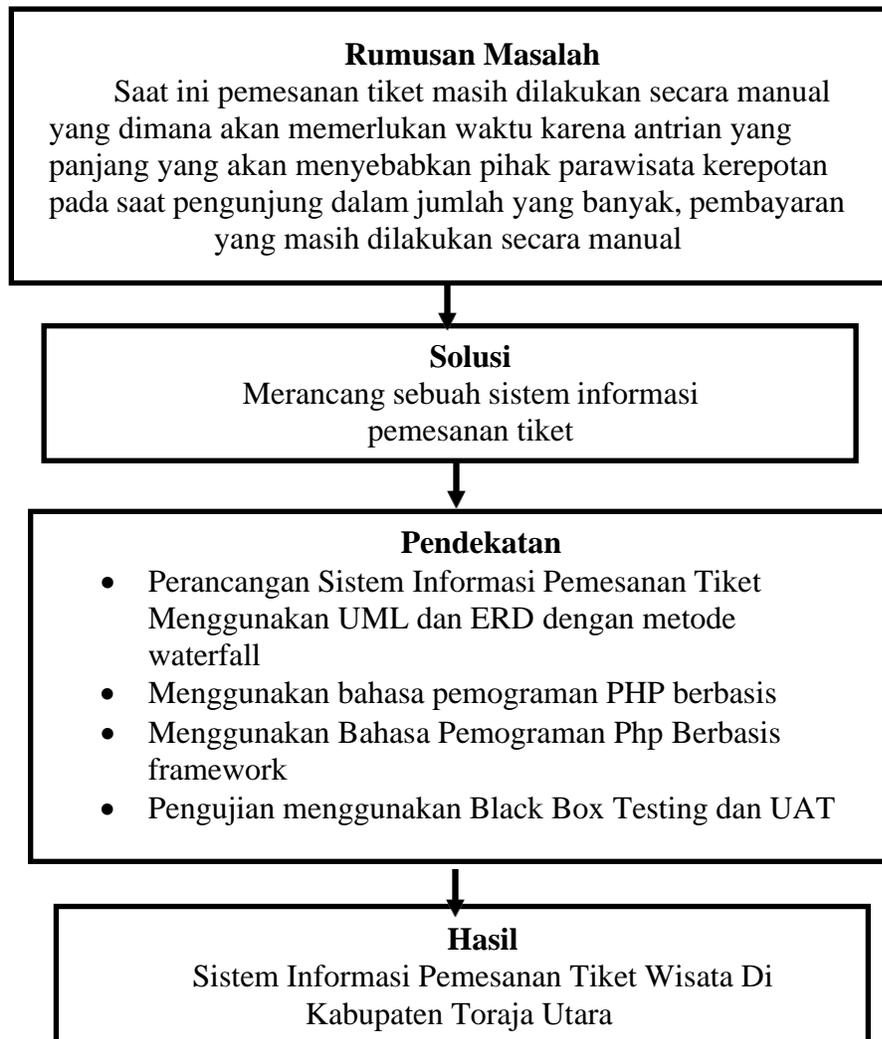
Metode pengujian *Blackbox Testing* adalah pengujian terhadap fungsional sistem apakah sudah sesuai dengan kebutuhan dan spesifikasi perangkat lunak untuk meninjau kondisi input dan output sistem dengan serangkaian persyaratan fungsional suatu program.

2. UAT (*User Acceptance Test*)

Metode pengujian UAT adalah suatu aktivitas pengujian yang dilakukan untuk mengetahui apakah sistem yang telah dikembangkan sesuai dengan kebutuhan dan dapat menyelesaikan masalah dari *user*. Proses ini adalah tahapan akhir yang dilakukan oleh pengguna perangkat lunak yang harus dilakukan untuk memastikan user merasa nyaman saat menggunakan dan sesuai dengan kebutuhannya.

2.3 Kerangka Pikir

Kerangka pikir atau tahapan penelitian yang akan dilakukan penulis digambarkan seperti pada Gambar dibawah ini :



Gambar 2.2 Kerangka Pikir