

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1. Sistem**

Sistem adalah kumpulan atau himpunan dari unsur atau variable-variabel yang saling terkait, saling berinteraksi, dan saling tergantung satu sama lain untuk mencapai tujuan (Tohari, 2017).

Sistem dapat didefinisikan sebagai kumpulan dari elemen-elemen berupa data, jaringan kerja dari prosedur-prosedur yang saling berhubungan, sumber daya manusia, teknologi baik hardware dan software yang saling berinteraksi sebagai kesatuan untuk mencapai tujuan atau sasaran tertentu yang sama (Maniah and Haminidin, 2017).

Dari beberapa kutipan di atas maka penulis dapat menyimpulkan bahwa sistem informasi adalah sistem di dalam suatu instansi atau organisasi perusahaan yang mempertemukan kebutuhan pengolahan transaksi harian dan memberikan laporan-laporan atau informasi yang dibutuhkan.

#### **2.2. Informasi**

Informasi merupakan kumpulan data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerima. Tanpa suatu informasi, suatu sistem tidak akan berjalan dengan lancar dan akhirnya bisa mati. Suatu organisasi tanpa adanya suatu informasi maka organisasi tersebut tidak bisa berjalan dan tidak bisa beroperasi (Kristanto, 2018).

### 2.3. Sistem Informasi

Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi organisasi yang bersifat manajerial dalam kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan (Kristanto,2018).

### 2.4. Pemesanan

Pemesananyaitu pemesanan fasilitas yang diantaranya akomodasi, meal, seat pada pertunjukan, pesawat terbang, kereta api, bus, hiburan, *night club*, *discoutegue* dan sebagainya (Suartana, 2013). Sedangkan menurut (McLoad, 2014) yang dimaksud pemesanan adalah proses, perbuatan, cara memesan (tempat, barang, dsb) kepada orang lain.

Berdasarkan pengertian para ahli diatas dapat disimpulkan bahwa *booking* adalah suatu aktifitas yang dilakukan oleh konsumen sebelum membeli atau melakukan *service* untuk mewujudkan kepuasan konsumen maka perusahaan harus mempunyai sebuah sistem pemesanan yang baik.

Pemesanan dibagi menjadi dua jenis yaitu pemesanan *online* dan *order offline*:

1) Pemesanan *Online*.

Kemajuan teknologi saat ini mengakibatkan sistem *order* juga mengalami perkembangan kearah sistem *order online*. *Order online* bisa diakses oleh siapapun dan dimanapun mereka berada yang memiliki akses *internet*.

2) Pemesanan *Offline*

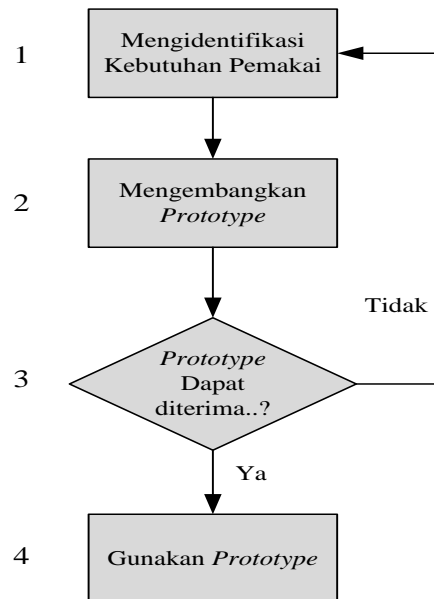
*Sistem order* yang menggunakan pengiriman *order* langsung ke tempat dengan media *order* seperti *telepon*, *fax*, *e-mail*, dan *walk in*.

## **2.5. Pariwisata**

Pariwisata merupakan kegiatan perjalanan yang dilakukan secara sukarela yang bersifat sementara dengan tujuan menikmati objek wisata untuk mencari kenikmatan dan pemenuhan rasa ingin tahu tentang suatu tempat baru. Pariwisata berkembang karena adanya gerakan manusia dalam mencari sesuatu yang belum diketahuinya, menjelajahi wilayah baru, mencari perubahan suasana, atau untuk mendapatkan kesenangan dari perjalanan baru. Pariwisata ini biasa menyediakan alternatif transportasi, penginapan, tour guide, rekomendasi landmark daerah tujuan wisata, dan lainnya, yang bertujuan untuk memudahkan orang dalam berwisata (Ervina *et al.*, 2019).

## **2.6. Metode Pengembangan Sistem *Prototype***

Menurut McLoad (2017) mendefinisikan *Prototype* adalah suatu versi sistem potensial yang memberikan ide bagi para pengembang dan calon pengguna yang memberikan gambaran bagaimana sistem akan berfungsi dalam bentuk yang telah selesai akan berfungsi bila telah disusun dalam bentuk yang lengkap. Proses dalam memproduksi suatu prototipe disebut *Prototyping*".



**Gambar 2. 1 Metode Prototyping Evolutionary**

Sumber: Mc Leod (2017)

Ada empat tahapan dalam pengembangan sistem model *Evolutionary Prototype*, yaitu sebagai berikut:

1. Identifikasi kebutuhan pemakai  
Pengembang mengidentifikasi terhadap pemakai untuk memperoleh suatu gagasan mengenai apa yang dibutuhkan dari sistem yang akan digunakan.
2. Mengembangkan *prototype*  
Pengembang menggunakan satu atau lebih perkakas *prototyping* untuk mengembangkan satu *prototype*. Contoh perkakas *prototyping* adalah *integrated application generator* (perangkat pembuat aplikasi terintegrasi), yaitu sistem perangkat lunak *prewritten* yang mampu memproduksi semua fasilitas-fasilitas yang diharapkan ada dalam sistem baru serta dengan membuat perancangan sementara yang berfokus pada penyajian kepada pemakai (misalnya dengan membuat *input* dan format *output*).

3. Menentukan apakah *prototype* bisa diterima atau tidak

Tahap ini dilakukan oleh pemakai sistem apakah *prototyping* yang sudah dikembangkan bisa diterima atau tidak. Jika sudah sesuai maka langkah empat akan diambil, jika tidak *Prototyping* direvisi dengan mengulangi langkah satu, dua, dan tiga dengan pemahaman yang lebih baik mengenai kebutuhan pemakai.

4. Gunakan *prototype*

Tahap ini dilakukan oleh pemakai sistem untuk menggunakan sistem yang telah dibangun.

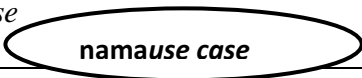
## 2.7. Pengertian Bahasa Pemodelan Pengembangan Sistem (*UML*)






Bahasa Pemodelan Pengembangan Sistem (*Unified Modeling Language*) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis & desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek. (Rosa and Shalahudin, 2018). Beberapa jenis diagram *UML* antara lain sebagai berikut:

### 2.7.1. *Use Case Diagram*

*Use case* diagram atau diagram *use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat (Rosa and Shalahudin, 2018), simbol-simbol yang ada pada diagram *use case* dapat dilihat pada gambar 2.1 di bawah ini:

**Tabel 2. 1 Simbol Diagram *Use Case***

Simbol	Deskripsi
<i>Use Case</i> 	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar

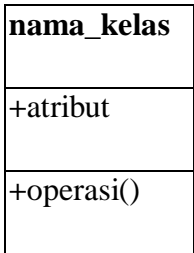



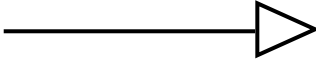

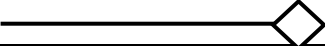
	pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>use case</i>
Aktor/ <i>actor</i> 	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda di awal frase nama <i>actor</i>
Asosiasi/ <i>association</i> 	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan <i>actor</i>
Ekstensi/ <i>extend</i>  	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek biasanya <i>use case</i> tambahan memiliki nama depan
Generalisasi/ <i>generalization</i>  	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
Menggunakan/ <i>Include/uses</i>   <i>&lt;&lt;include&gt;&gt;</i>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya

**Sumber:** (Rosa and Shalahudin, 2018)

### 2.7.2. Class Diagram

Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi (Rosa and Shalahudin, 2018), simbol-simbol yang ada pada diagram kelas pada tabel *class diagram 2.2* di bawah ini:

Tabel 2. 2 Simbol *Class Diagram*

Simbol	Deskripsi
Kelas 	Kelas pada struktur system
Antarmuka/ <i>Interface</i>  <b>nama_interface</b>	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek
Asosiasi/ <i>association</i> 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
Asosiasi berarah/ <i>directed association</i> 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya disertai dengan <i>multiplicity</i>
Generalisasi 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus)
Kebergantungan/ <i>dependecy</i> 	Relasi antar kelas dengan makna kebergantungan antar kelas
Agregasi/ <i>agregation</i> 	Relasi antar kelas dengan makna semua bagian ( <i>whole-part</i> )





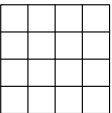


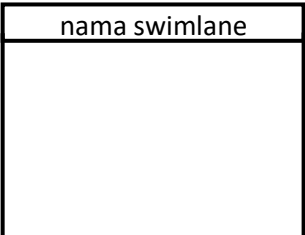
Sumber : (Rosa and Shalahudin, 2018)

### 2.7.3. Activity Diagram

Activity diagram atau Diagram aktivitas menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Yang perlu diperhatikan

disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem (Rosa and Shalahudin, 2018), simbol-simbol yang ada pada *activity diagram* dapat dilihat pada tabel 2.4 di bawah ini :

**Tabel 2. 3 Simbol Activity Diagram**

Simbol	Deskripsi
Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
Percabangan/ <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
Penggabungan/ <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu
Tabel 	Suatu file komputer dari mana data bisa dibaca atau direkam selama kejadian bisnis
Dokumen 	Menunjukkan dokumen sumber atau laporan
Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
<i>Swimlane</i> 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi



## 2.8. Pengertian *Mobile*

Aplikasi adalah program yang digunakan orang untuk melakukan sesuatu pada sistem komputer. *Mobile* dapat diartikan sebagai perpindahan yang mudah dari satu tempat ke tempat yang lain, misalnya telepon *mobile* berarti bahwa terminal telepon yang dapat berpindah dengan mudah dari satu tempat ke tempat lain tanpa terjadi pemutusan atau terputusnya komunikasi. Aplikasi *mobile* merupakan aplikasi yang dapat digunakan walaupun pengguna berpindah dengan mudah dari satu tempat ketempat lain lain tanpa terjadipemutusan atau terputusnya komunikasi. Aplikasi ini dapat diakses melalui perangkat nirkabel seperti pager, seperti telepon seluler dan PDA (Yuhefizar, 2016).

Adapun karakteristik perangkat *mobile* yaitu:

1. Ukuran yang kecil: Perangkat *mobile* memiliki ukuran yang kecil. Konsumen menginginkan perangkat yang terkecil untuk kenyamanan dan mobilitas mereka.
2. Memory yang terbatas: Perangkat *mobile* juga memiliki memory yang kecil, yaitu *primary* (RAM) dan *secondary* (*disk*).
3. Daya proses yang terbatas: Sistem *mobile* tidaklah setangguh rekan mereka yaitu desktop.
4. Mengonsumsi daya yang rendah: Perangkat *mobile* menghabiskan sedikit daya dibandingkan dengan mesin desktop
5. Kuat dan dapat diandalkan: Karena perangkat *mobile* selalu dibawa kemana saja, mereka harus cukup kuat untuk menghadapi benturan-benturan, gerakan, dan sesekali tetesan-tetesan air.

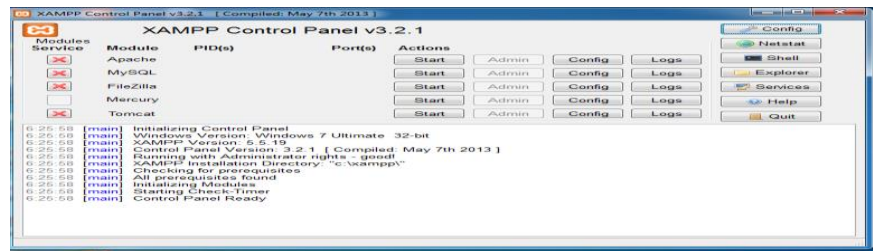
6. Konektivitas yang terbatas: Perangkat *mobile* memiliki bandwidth rendah, beberapa dari mereka bahkan tidak tersambung.
7. Masa hidup yang pendek : Perangkat-perangkat konsumen ini menyala dalam hitungan detik kebanyakan dari mereka selalu menyala.

## 2.9. Pengertian MySQL

Menurut Rosa and Shalahudin (2018)MySQL adalah bahasa yang digunakan untuk mengelola data pada *Relation DBMS (Database Management System)*.

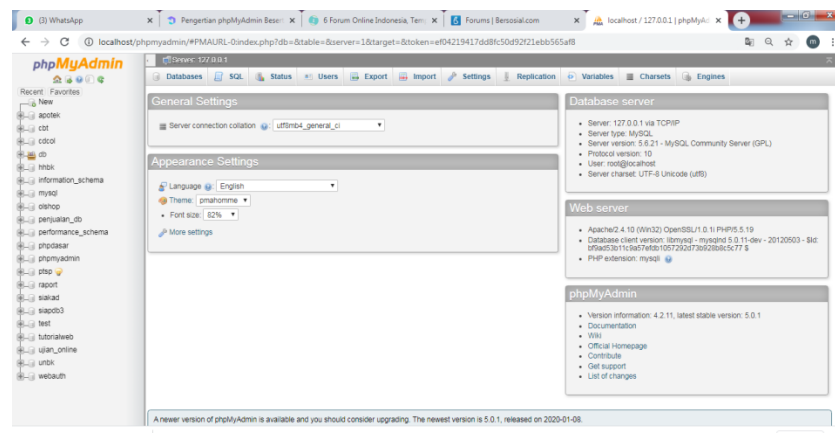
## 2.10. XAMPP

XAMPP merupakan perangkat lunak bebas (*open source*) yang mendukung banyak sistem operasi, merupakan campuran dari beberapa *program*. Yang mempunyai fungsi sebagai server yang berdiri sendiri (*localhost*), yang terdiri dari program *MySQL database, Apache HTTP Server*, dan penerjemah ditulis dalam bahasa pemrograman PHP dan Perl. Nama XAMPP merupakan singkatan dari X (empat sistem operasi), Apache, MySQL, PHP dan Perl. Program ini tersedia di bawah GNU (*General Public License*) dan bebas, adalah mudah untuk menggunakan web server yang dapat melayani tampilan halaman web yang dinamis. Menurut (Hidayatullah and Kawistara, 2017) XAMPP *support* untuk banyak sistem operasi seperti *Windows, Linux, Mac OS* dan *Solaris* sehingga tidak terdapat masalah ketika melakukan perpindahan sistem operasi dan Menurut (Nugroho, 2015) XAMPP adalah paket web programming, akan tetapi kita bisa memanfaatkan *database MySQL server*-nya untuk belajar Programming Visual, juga disana telah tersedia *tools PHPMyadmin* yang hanya berjalan disisi *server* web seperti *Apache Server*.



**Gambar 2. 2 Gambar Xampp**

*PhpMyAdmin* merupakan salah satu software penting dalam pengelolaan database dengan menggunakan MySQL (Bahasa SQL). Sesuai dengan namanya *phpMyAdmin* ini ditulis dalam Bahasa pemrograman PHP. *PhpMyAdmin* khusus menangani pengelolaan database dalam lingkup website (World Wide Web). Sama halnya dengan MySQL, software yang rilis perdana pada tahun 1998 ini juga berlisensi GNU (*General Public License*) dan sudah mensupport *multilingual* (multi bahasa) dalam user interfacenya.



**Gambar 2. 3 Gambar PhpMyadmin**

## 2.11. Pengujian *Black – Box*

Pendekatan pengujian *Black-Box* adalah metode pengujian di mana data tes berasal dari persyaratan fungsional yang ditentukan tanpa memperhatikan struktur program akhir. Karena hanya fungsi dari modul perangkat lunak yang menjadi perhatian, pengujian *Black-Box* juga mengacu pada uji fungsional, metode pengujian

menekankan pada menjalankan fungsi dan pemeriksaan inputan dan data output (Howden, 2017).

Pengujian *black-box* berusaha menemukan kesalahan dalam kategori sebagai berikut :

- a. Fungsi – fungsi yang tidak benar atau hilang,
- b. Kesalahan interface
- c. Kesalahan dalam struktur data atau akses eksternal
- d. Kesalahan kinerja
- e. Inisialisasi dan kesalahan terminasi

Pada *black box testing* terdapat jenis teknik design tes yang dapat dipilih berdasarkan pada tipe testing yang akan digunakan, diantaranya sebagai berikut:

1. *Equivalence Class Partitioning*
2. *Boundary Value Analysis*
3. *State Transitions Testing*
4. *Cause-Effect Graphing*

## 2.12. Penelitian Sebelumnya

Berikut ini adalah beberapa literature yang digunakan dalam penelitian, dapat dilihat pada tabel berikut :

**Tabel 2. 4 Penelitian Sebelumnya**

NO	Penulis (Tahun)	Judul	Metode	Masalah	Hasil
1.	(Riskiono, 2018)	Sistem Informasi Pelayanan Jasa Tour Dan Travel Berbasis Web (Studi	Metode <i>extreme programming</i>	Smart Tour ini sudah berpengalaman di banyak industri bisnis pariwisata bukan	Sistem informasi pelayanan jasa tour dan travel berbasis web (Studi kasus

NO	Penulis (Tahun)	Judul	Metode	Masalah	Hasil
		Kasus Smart Tour)		hanya biro perjalanan, tetapi Smart Tour ini juga mempunyai jasa property dan Rental Mobil. Selama ini sistem yang ada masih belum sesuai dengan kebutuhan masyarakat, dikarenakan proses pemesanan tiket masih dilakukan secara manual. Masyarakat harus datang langsung ke Smart Tour untuk memesan tiket liburan lokal maupun domestic. Smart Tour juga menyediakan jasa Penyewaan Mobil dimana sistem yang ada sebelumnya masih dicatat secara manual dan belum terkomputerisasi.	Smart Tour) dapat mampu mengurangi resiko kesalahan pada pengelolaan data pada admin dan Sistem informasi pelayanan jasa tour dan travel berbasis web (Studi kasus Smart Tour) telah mampu mengurangi waktu yang terbuang pada saat pelanggan memesan tiket tour
2.	(Yakanita, Mahaputra and Lana, 2020)	Aplikasi Pemesanan Jasa Tour Guide Dan Vacation Planner Di Bali Berbasis Mobile	Metode <i>waterfall</i>	Pemasaran masih dilakukan secara manual	menghasilkan aplikasi sistem informasi pemesanan jasa Tour Guide dan Vacation Planner berbasis mobile yang nantinya diharapkan bermanfaat bagi para wisatawan

NO	Penulis (Tahun)	Judul	Metode	Masalah	Hasil
					dalam membuat perencanaan dan mempermudah kunjungan mereka saat berada di Bali. Aplikasi
3	(Naatonis and Bisilisin, 2020)	Aplikasi Pemandu Pariwisata Di Kota Kupang Berbasis Mobile Website	Metode <i>waterfall</i>	Masih sering mengalami kesulitan baik dalam menemukan tempat wisata dan masih kesulitan dalam mengetahui fasilitas apa yang terdapat pada tempat wisata tersebut.	Hasil penelitian yang telah dilaksanakan berupa pemandu pariwisata berbasis mobile website yang dapat memberikan informasi kepada wisatawan berupa informasi fasilitas, tempat, lokasi, serta memberikan keterangan tempat wisata lengkap dengan denah wisata.
4	(Ervina <i>et al.</i> , 2019)	Analisis Dan Perancangan Sistem Reservasi Paket Wisata Untuk Internal Karyawan Pt. Garuda Maintenance Facility (Gmf) Tbk	<i>object oriented analysis and design</i> (OOAD)	memiliki kendala dalam melakukan reservasi paket wisata karena kurangnya pengetahuan tentang tempat wisata	Hasil penelitian adalah berupa rancangan sistem yang nantinya dapat memberikan kemudahan bagi karyawan yang akan melakukan reservasi paket wisata, serta solusi bisnis bagi KopKar

NO	Penulis (Tahun)	Judul	Metode	Masalah	Hasil
					GMF AAS. Karyawan
5	(Rodiah, Hanggara and Ratnawati, 2020)	Pengembangan Sistem Informasi Pemesanan Paket Wisata ( Studi Kasus : Agen Wisata Liburan Sekolah )	metode waterfall	Untuk melakukan pemesanan paket wisata, pengunjung harus menghubungi customer service melalui telpon atau whatsApp dan customer service mendata setiap pesanan yang masuk secara manual. Hal	Dihasilkan suatu sistem informasi pemesanan paket wisata yang dapat memudahkan wisatawan dan admin dan proses transaksi