

BAB II

LANDASAN TEORI

2.1 Tinjauan Pustaka

Dalam menyusun Tugas Akhir ini, penyusun menggunakan referensi tinjauan pustaka, antara lain adalah

1. Tugas akhir Rira Nurul Sani (2016) Program studi D-III Teknik Informatika Fakultas MIPA UNS dengan judul Pembuatan Sistem Informasi Pemesanan Paket Perjalanan Wisata Berbasis *Web* . Abadi Tour & Travel adalah salah satu biro perjalanan wisata yang beralamatkan di Jl. Raya Palur Karanganyar, Surakarta. Selama ini, untuk promosi paket wisata yang baru beserta perubahan harga, masih disampaikan melalui telepon dan brosur, sehingga informasi yang disampaikan kurang cepat dan lengkap. Sedangkan untuk pemesanan paket wisata masih dilakukan dengan cara yang konvensional. Selain sebagai media untuk mempromosikan paket wisata, sistem informasi ini memberikan kemudahan kepada masyarakat untuk memesan paket wisata yang diinginkan. Dengan adanya sistem ini diharapkan pelanggan dapat melihat informasi terbaru pada Abadi Tour & Travel jika terjadi perubahan paket beserta harganya, serta dapat melakukan pemesanan kapan saja dan dimana saja tanpa harus mendatangi kantor.
2. Tugas akhir Rahmat Nur Maulidin (2015) yang berjudul “Sistem Informasi Paket Wisata Pada CV. KHANS TOUR BERBASIS *WEB*” Program studi Informatika Fakultas Komunikasi dan Informatika UMS. Di dalam tugas akhir tersebut dibahas mengenai proses pembuatan dan pengelolaan paket wisata, pembuatan form pemesanan, pembuatan itinerary, penghitungan harga, dan proses reservasi guna meningkatkan pelayanan terhadap pelanggan CV. Khans Tour sistem informasi pemesanan paket wisata secara online ini dibuat dengan tujuan untuk membantu memperluas promosi dan meningkatkan keuntungan perusahaan. Sistem berbasis website ini dibangun menggunakan PHP dan MySQL. Tahapan – tahapan dalam metode ini

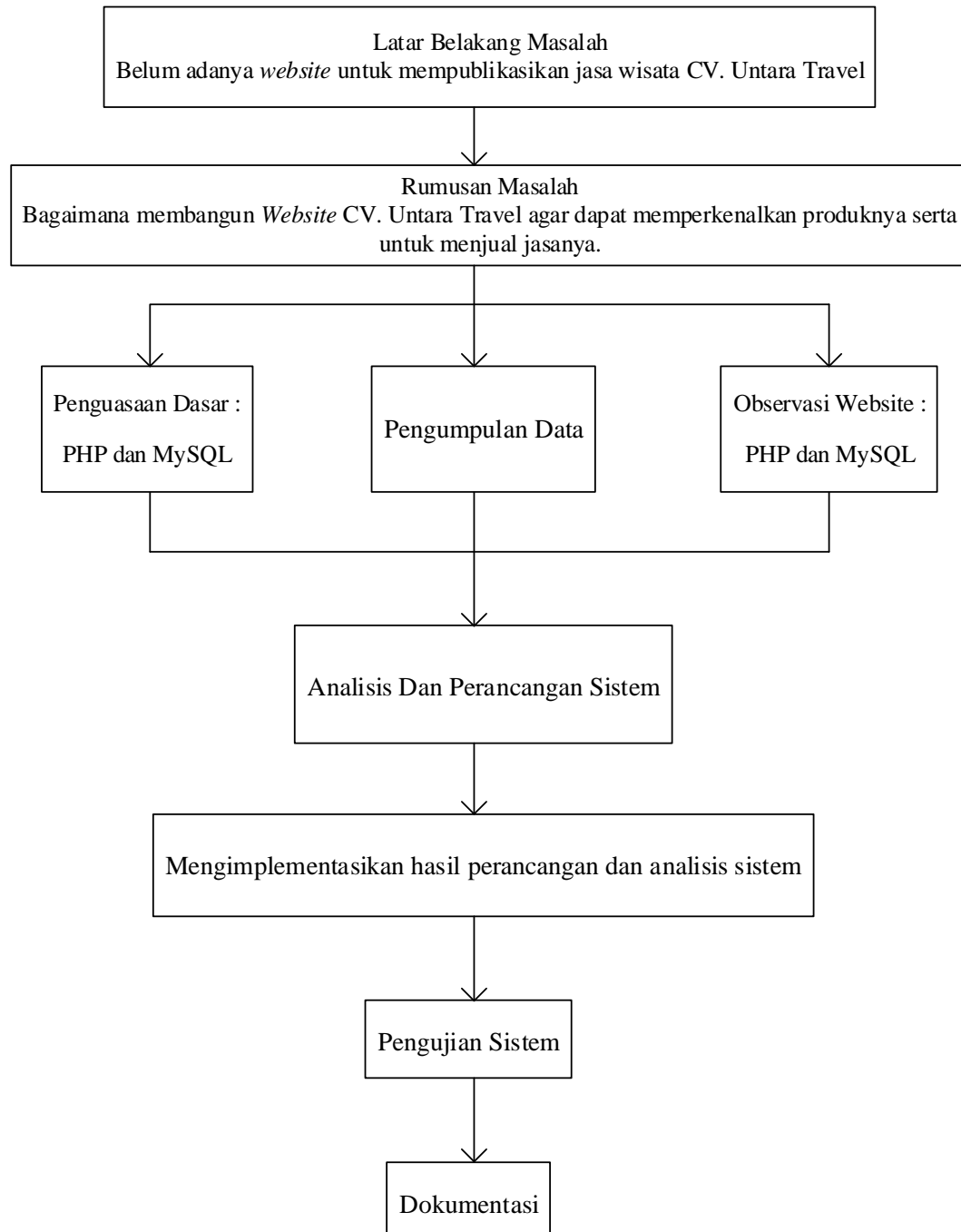
adalah analisis kebutuhan, desain sistem, implementasi, pengujian dan maintenance. Sistem Informasi yang telah dibangun memberikan kemudahan untuk mendapatkan informasi bagi masyarakat khususnya yang ingin mengetahui paket wisata. Hasil dari penelitian ini admin bisa mengelola data-data paket, manajemen *website* (*about us*, berita, *chatting*) dan cetak pemesanan.

3. Tugas akhir Maylita Rachmadin Meliala (2015) di Sekolah Vokasi Universitas Gadjah Mada dengan judul “Manajemen Paket Wisata di Bima Sena Wisata Tour dan Travel Guna Meningkatkan Pelayanan terhadap Pelanggan”. Di dalam tugas akhir tersebut dibahas mengenai proses pembuatan dan pengelolaan paket wisata, pembuatan *form* pemesanan, pembuatan *itinerary*, penghitungan harga, dan proses reservasi guna meningkatkan pelayanan terhadap pelanggan.
4. Tugas akhir Aristi Santi Sundari (2012) Program Studi Ilmu Komunikasi Universitas Sahid Surakarta yang berjudul “EFEKTIVITAS KOMUNIKASI ORGANISASI “ menganalisis Pengaruh Variabel – Variabel Komunikasi Organisasi Terhadap Loyalitas Perusahaan Rizma Tour & Travel. Di dalam tugas akhir tersebut dibahas pengolahan data kuantitatif antar variabel X (independent) dan Y (dependen) adalah $r = 0.257$ yang berarti bahwa terdapat hubungan yang lemah berarti tidak terdapat pengaruh antar variabel komunikasi organisasi terhadap loyalitas karyawan pada Rizma Tour & Travel

Kesimpulan perbandingan referensi nomer 1, 2, dan 3 dari beberapa *website* yang sudah ada dengan yang baru adalah sama – sama mempromosikan jasanya melalui *website* dan memberi informasi mengenai tujuan dan paket wisata. Sedangkan perbedaannya yaitu sudah menyediakan penginapan yang bisa dipilih sesuai *budget* dan selera wisatawan. Referensi nomer 4 hanya menganalisis efektivitas komunikasi yang tidak berhubungan dengan pembuatan *website* atau media promosi.

2.2 Kerangka Pemikiran

Diagram kerangka pemikiran yang dijalankan dalam penelitian ini dapat dilihat pada Gambar 2.1



Gambar 2.1 Diagram Kerangka Pemikiran

Dari diagram kerangka pemikiran di atas, latar belakang masalah adalah bagaimana mempromosikan CV. New Utara Travel secara luas dengan biaya yang rendah dan membuat *website* pesanan wisata *tour* dan *travel* agar memudahkan pegawai untuk mengatur jadwal perjalanan wisata. Berdasarkan latar belakang ,maka dibuatlah *website* untuk CV. New Utara agar dapat memperkenalkan secara luas dan menawarkan jasanya. Rumusan masalah dari kerangka pemikiran adalah bagaimana membangun *website* CV. New Utara Travel agar dapat dikenal luas serta dapat memperkenalkan jasanya. Dalam pembuatan *website* ada 3 hal yang perlu diperhatikan yaitu penguasaan dasar *PHP* & *SQL*, kemudian mengumpulkan data – data, dan yang terakhir adalah observasi. Dari 3 aspek di atas menghasilkan sebuah analisis dan perancangan sistem membangun website CV. New Utara Travel , setelah selesai melakukan perancangan kemudian mengimplementasikan perancangan ke dalam bahasa pemograman *PHP*. Pengujian sistem untuk mengetahui kesalahan yang mungkin tidak diketahui sbelumnya agar segera diperbaiki sebelum diimplementasikan.

2.3 Teori Pendukung

2.3.1 *Travel* & Produk wisata (Perjalanan Wisata)

mengungkapkan bahwa pariwisata adalah rangkaian aktivitas, dan penyediaan layanan baik untuk kebutuhan atraksi wisata, transportasi, akomodasi, dan layanan lain yang ditujukan untuk memenuhi kebutuhan perjalanan seseorang atau sekelompok orang. Perjalanan yang dilakukannya hanya untuk sementara waktu saja meninggalkan tempat tinggalnya dengan maksud beristirahat, berbisnis, atau untuk maksud lainnya (Sugiama, 2011) Produk wisata adalah semua bentuk pelayanan yang dinikmati wisatawan dari berangkat meninggalkan tempat tinggal hingga kembali pulang (Silalahi, 2011)

2.3.2 *Website*

Internet adalah jaringan fisik yang menghubungkan komputer di seluruh dunia. *Internet* sendiri terdiri dari infrastruktur jaringan server dan hubungan

komunikasi diantaranya digunakan untuk menyimpan dan memindahkan informasi antara komputer klien dan *web server* (Chaffey, 2011).

Menghubungkan beberapa komputer sehingga dapat menjadi sebuah jaringan membutuhkan suatu media penghubung yang bernama TCP/IP, yaitu sebuah protocol yang mengidentifikasi sebuah computer yang terhubung di dalam sebuah jaringan TCP/IP memiliki teknik mengidentifikasi dengan menggunakan penomoran yang dinamakan nomor IP/*IP address (Internet Protocol Address)*. Nomor IP sebuah komputer dapat terhubung dengan computer lain dalam sebuah jaringan atau dalam jaringan global yang disebut *internet*.

Website atau situs dapat diartikan sebagai kumpulan halaman yang menampilkan informasi data teks, data gambar, diam atau gerak, data animasi suara, video dan atau gabungan dari semuanya. Baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait dimana masing-masing dihubungkan dengan jaringan-jaringan halaman (Permana, 2012).

2.3.3 PHP

PHP adalah bahasa *server-side scripting* yang menyatu dengan *HTML* untuk membuat halaman *web* yang dinamis. Maksud dari *server-side scripting* adalah sintaks dan perintah-perintah yang diberikan akan sepenuhnya dijalankan di server tapi disertakan pada dokumen *HTML*. Pembuatan *web* ini merupakan kombinasi antara php sendiri sebagai bahasa pemrograman dan *HTML* sebagai pembangunan halaman *web* (Hendrianto, 2014).

2.3.4 Basis Data (Database)

Database sering didefinisikan sebagai kumpulan data yang terkait. Secara teknis yang berada dalam sebuah *database* adalah sekumpulan tabel atau objek lain indeks, *view*, dan lain-lain. Tujuan utama pembuatan *database* adalah untuk memudahkan mengakses data. Data dapat ditambahkan, dihapus, atau dibaca dengan relatif mudah dan cepat. Saat ini tersedia banyak perangkat lunak yang ditujukan untuk mengelola *database* (Permana, 2012).

2.3.5 MySQL

MySQL adalah *multi user database* yang menggunakan bahasa *Structured Query Language (SQL)*. *MySQL* dalam operasi *client server* melibatkan server *daemon MySQL* di sisi server dan berbagai macam program serta *library* yang berjalan di sisi *client*. *MySQL* mampu menangani data yang cukup besar. Perusahaan yang mengembangkan *MySQL* yaitu TEX., mengaku mampu menyimpan data lebih dari 40 *database*, 10.000 tabel, dan sekitar 7.000.000 baris totalnya, kurang lebih 100 *Gigabyte* data (Hendrianto, 2014).

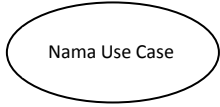
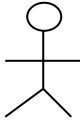
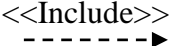
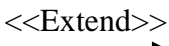

2.3.6 UML

Perkembangan teknologi perangkat lunak diperlukan adanya bahasa yang digunakan untuk memodelkan perangkat lunak yang akan dibuat dan perlu adanya standarisasi agar orang di berbagai negara dapat mengerti pemodelan perangkat lunak. Perkembangan teknik pemrograman berorientasi objek, muncul sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemrograman berorientasi objek, yaitu *Unified Modelling Language*. *Unified Modelling Language (UML)* merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. UML hanya berfungsi untuk melakukan pemodelan, jadi penggunaan UML tidak terbatas pada teknologi tertentu (Sukamto dan Shalahuddin, 2013).

2.3.6.1 Use Case Diagram

Use case diagram yang menggambarkan *actor*, *use case* dan relasinya sebagai suatu urutan tindakan yang memberikan nilai terukur untuk aktor. Sebuah *use case* digambarkan sebagai elips horizontal dalam suatu diagram UML *use case*. Simbol – simbol *Use Case Diagram* dapat dilihat pada Tabel 2.1.




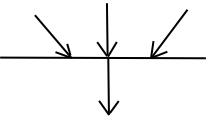

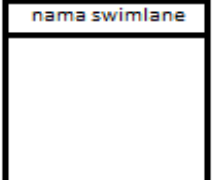
Tabel 2.1. Simbol – simbol *Use Case Diagram*

No	Gambar	Nama	Keterangan
1		<i>Use Case</i>	Fungsional yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.
2		<i>Actor</i>	Orang proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat itu sendiri.
3		<i>Include</i>	Menspesifikasikan bahwa pemanggilan <i>use case</i> oleh <i>use case</i> lain
4		<i>Extend</i>	Menspesifikasikan bahwa perluasan dari <i>use case</i> lain jika kondisi atau syarat terpenuhi
5		<i>Generalisasi/ Generalizati on</i>	Hubungan generalisai dan spesialis antara dua buah <i>use case</i> .

2.3.6.2 Activity Diagram

Diagram aktivitas atau *Activity diagram* menggambarkan *workflow* atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak, perlu diperhatikan juga disini diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor (Sukamto dan Shalahuddin, 2013). Simbol-simbol yang ada pada *activity diagram* dapat dilihat pada Tabel 2.2.


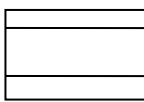
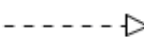
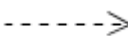

Tabel 2.2 Simbol-Simbol *Activity Diagram*

NO	NAMA	SIMBOL	KETERANGAN
1	Status awal		Status awal aktivitas sistem.
2	Aktivitas		Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3	Percabangan		Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
4	<i>Join</i>		Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
5	Status akhir		Status akhir yang dilakukan sistem.
6	<i>Swimlane</i>		Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

2.3.6.3 *Class Diagram*

Class Diagram menggambarkan struktur dan deskripsi *class*, *package* dan *object* beserta hubungan satu sama lain seperti *contaiment*, pewarisan, asosiasi, dan lain-lain. *Class diagram* membantu dalam visualisasi struktur kelas-kelas dari suatu sistem yang merupakan tipe *diagram* yang paling banyak dipakai. Simbol-simbol *Class Diagram* dapat dilihat pada Tabel 2.3.


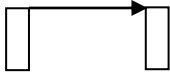




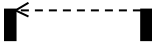
Tabel 2.3 Simbol-simbol *Class Diagram*

No	Gambar	Nama	Keterangan
1		<i>Generalization</i>	Hubungan dimana object anak (<i>descendent</i>) berbagi perilaku dan struktur data dari object yang ada di atasnya object induk (<i>ancestor</i>).
2		<i>Class</i>	Himpunan dari object-object yang berbagi atribut serta operasi yang sama.
3		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu object.
4		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan memengaruhi elemen yang bergantung padanya elemen yang tidak mandiri
5		<i>Association</i>	Apa yang menghubungkan antara object satu dengan object lainnya

2.3.6.4 *Sequence Diagram*

Sequence Diagram mendeskripsikan bagaimana entitas dalam sistem berinteraksi, termasuk pesan yang digunakan saat interaksi. *Diagram* ini juga menunjukkan serangkaian pesan yang diperlukan oleh objek-objek yang melakukan suatu tugas atau aksi tertentu. Simbol – simbol *Sequence Diagram* dapat dilihat pada Tabel 2.4

Tabel 2.4 Simbol – simbol *Sequence Diagram*

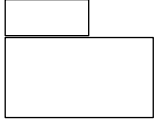
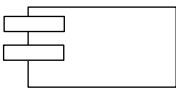
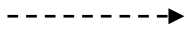

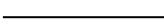
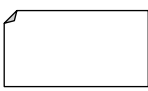
No	Gambar	Nama	Keterangan
1		<i>LifeLine</i>	Object <i>entity</i> , antarmuka yang saling berinteraksi.
2		<i>Message</i>	Spesifikasi dari komunikasi antar object yang memuat informasi-informasi tentang aktifitas yang terjadi.
3		<i>Actor</i>	Pengguna diluar sistem.
4		<i>Boundary</i>	Boundary biasanya berupa tepi dari system, seperti user interface, atau suatu alat yang berinteraksi dengan system lain.
5		<i>Control element</i>	Control element mengatur aliran dari informasi untuk sebuah scenario.
6		<i>Entity</i>	Entity biasanya elemen yang bertanggung jawab menyimpan data atau informasi.
7		<i>Return</i>	Spesifikasi dari komunikasi antar object yang memuat informasi-informasi tentang aktifitas yang terjadi

2.3.6.5 Component Diagram

Component diagram menggambarkan struktur dan hubungan antar komponen peranti lunak, termasuk ketergantungan (*dependency*). Diantaranya modul berisi kode, baik berisi *source code*, *binary*, *library*, *executable*. *User interface* adalah level terakhir yang bisa dilahit oleh pengguna, sedangkan sistem pendukung lain

seperti operasi atau *database* dan mesin *logic* program tidak akan terlihat oleh pengguna. *Component Diagram* disajikan pada Tabel 2.5.

Tabel 2.5 Simbol – simbol *Component Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Package</i>	Merupakan sebuah bungkus dari satu atau lebih komponen
2		<i>Komponen</i>	Komponen sistem
3		<i>Dependency</i>	Ketergantungan komponen anak panah mengarah pada komponen yang dipakai
4		<i>Nama_Interface</i>	Agar tidak langsung mengakses langsung komponen
5		<i>Link</i>	Relasi antar komponen
6		<i>Document</i>	Dokumen dapat berupa file <i>library</i>

2.3.7 Pengujian Sistem

1) Metode WebQual 4.0

Webqual adalah “webqual for assessing the usability, information, and service interaction quality of Internet web-sites, particularly those offering e-commerce facilities” Webqual adalah sebuah alat untuk menilai kegunaan, kualitas informasi dan kualitas interaksi pelayanan dari halaman web pada internet, khususnya yang menggunakan fasilitas e-commerce. Webqual merupakan salah satu metode atau teknik pengukuran kualitas website berdasarkan persepsi pengguna akhir (Sanjaya 2012:12). Berdasarkan beberapa definisi di atas webqual adalah alat untuk mengukur kegunaan, kualitas informasi

dan kualitas interaksi pelayanan pada web. Metode webqual telah dikembangkan sejak tahun 1998 oleh Barnes dan Vidgen pada situs web e-commerce maupun e-government.

Penentuan jumlah sampel dihitung dengan menggunakan rumus Slovin.

$$n = \frac{n}{1+N(e)^2} \quad n : \text{unit sampel}$$

N : populasi

e : nilai error.

2) ***BlackBox Testing***

Blackbox testing atau pengujian kotak hitam adalah pengujian pada perangkat lunak dari sisi spesifikasi fungsional tanpa melakukan pengujian desain dan kode program. Pengujian yang ada dimaksudkan untuk mengetahui apakah fungsi-fungsi yang ada pada aplikasi berfungsi dengan normal, masukan dan keluaran data yang ada dari perangkat lunak apakah sudah sesuai dengan spesifikasi yang dibutuhkan. Pengujian kotak hitam dapat bisa dijalankan dengan kasus uji yang bersifat mencoba semua fungsi dengan memakai perangkat lunak secara keseluruhan dan melihat apakah sesuai dengan spesifikasi yang dibutuhkan (Sukanto & Shalahuddin, 2013).

Pengujian kotak hitam diterapkan dengan melakukan pembuatan kasus uji yang bersifat menjalankan semua fungsi yang berada pada aplikasi dengan memakai perangkat lunak apakah sesuai dengan spesifikasi yang telah ada. Kasus uji aplikasi yang dilakukan kotak hitam adalah kasus uji yang dibuat untuk melakukan pengujian pada aplikasi jika perintah yang diberikan benar dan jika perintah yang diberikan salah. Dalam data yang telah di uji tadi maka akan terlihat kesimpulan yang ada dan terlihat bagaimana aplikasi berfungsi.