



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

BAB II

LANDASAN TEORI

2.1 Teknik Pengumpulan Data

2.1.1 Wawancara

Menurut (Kamus Besar Bahasa Indonesia) wawancara umumnya dikenal sebagai komunikasi dengan seseorang seperti pejabat dan narasumber lainnya yang diperlukan untuk dimintai keterangan, pendapat, maupun informasi mengenai suatu hal, yang akan dimuat dalam surat kabar, keperluan tugas, lamaran kerja, dan keperluan lainnya.

Umumnya proses wawancara terjadi dengan adanya tanya jawab diantara kedua belah pihak yang sedang menjalani kegiatan wawancara tersebut. Wawancara dapat umumnya dilakukan secara *face-to-face* atau yang dikenal dengan wawancara secara langsung. Akan tetapi seiring dengan kemajuan teknologi, kini wawancara dapat dilakukan melalui pesawat telepon ataupun video-call dan teknologi lainnya, sehingga wawancara tidak perlu lagi dilakukan secara langsung, terutama bila *schedule* antara pihak narasumber dan pewawancara tidak sinkron, hal itu hanya akan menghalangi kegiatan wawancara secara langsung.

Kita dapat membedakan tiga pendekatan dasar dalam memperoleh data kualitatif melalui wawancara, yaitu (Francisca, 2005) :

a. Wawancara Informal, dimana kegiatan wawancara didasarkan sepenuhnya pada berkembangnya pertanyaan – pertanyaan secara spontan dalam interaksi alamiah.

b. Wawancara dengan pedoman wawancara, terjadi ketika peneliti dilengkapi pedoman wawancara yang sangat umum, yang mengandung isu – isu yang harus diliput tanpa menentukan urutan pertanyaan.

c. Wawancara dengan pedoman paling standar dan terbuka untuk wawancara ini, merupakan sebuah pedoman wawancara yang ditulis secara rinci, lengkap dengan kumpulan pertanyaan dan penjelasannya dalam kalimat.

2.1.2 Internet

Menurut (Kamus Besar Bahasa Indonesia) internet dikenal sebagai jaringan komunikasi elektronik yang menghubungkan jaringan komputer dengan fasilitas komputer lainnya yang terintegrasi di seluruh dunia melalui telepon maupun satelit internet.

Internet sendiri memberikan beberapa jasa umum seperti aplikasi *World Wide Web* (WWW), *e-mail*, *newsgroup*, dan lain-lain. Jasa internet yang paling sering digunakan saat ini adalah penggunaan aplikasi WWW untuk pencarian data maupun informasi yang dibutuhkan untuk tugas, pekerjaan,

maupun pengetahuan serta *e-mail* yang digunakan untuk pengiriman file maupun *chatting* antar penggunanya. Selain kedua hal tersebut, adapula *file sharing* dimana pihak-pihak pengguna internet saling mengirimkan data terhadap satu sama lain. Dalam hal pengiriman data perusahaan yang bersifat pribadi, umumnya memerlukan *encryption* sebagai kode - kode biner yang menyulitkan pihak tidak bertanggungjawab yang ingin menyalahgunakan data perusahaan tersebut.

2.1.3 Observasi

Menurut (Kamus Besar Bahasa Indonesia) observasi merupakan peninjauan secara cermat dengan menggunakan studi kasus maupun pengamatan langsung di lapangan berdasarkan pengalaman maupun uji coba. Observasi sendiri umumnya bisa dilakukan terhadap suatu kondisi, perilaku, maupun orang yang menjadi subyek dari suatu observasi tersebut berdasarkan.

Sedangkan menurut (Jogiyanto, 2005) observasi atau pengamatan (*observation*) merupakan salah satu teknik pengumpulan fakta/data yang cukup efektif untuk mempelajari suatu sistem. Observasi adalah pengamatan langsung suatu kejadian yang sedang dilakukan.

2.2 Teknik Pengolahan Data

2.2.1 *Rapid Application Development (RAD) Methods*

Berdasarkan (Gottesdiener, 1995) *Rapid Application Development* (RAD) merupakan kumpulan teknik, panduan, maupun alat bantu untuk memfasilitasi perangkat lunak bagi pihak pengguna dalam waktu yang singkat. RAD ini bersifat bekerja di 'belakang layar', dimana metode ini dapat berkembang berdasarkan *feedback* dari pengguna *software* tersebut, pada tahapan *RAD development process*.

Metode rapid ini bersifat *iterative* (berulang) sampai suatu proses selesai. Setelah proses selesai, maka metode ini tidak *auto-loop* atau terulang dengan sendirinya seperti pada metode *spiral*.

RAD memiliki 4 (empat) tahapan yang terdiri dari *requirement, user design, construction, serta cutover*.

Menurut (Noertjahyana, 2002) pada tahap *requirement*, pengguna dan seorang analis melakukan pertemuan untuk mengidentifikasi tujuan dari aplikasi serta kebutuhan apa saja yang dibutuhkan sebagai informasi maupun hal pendukung untuk pembuatan aplikasi tersebut. Hal terpenting dalam tahap ini adalah adanya keterlibatan dari kedua belah pihak yang melakukan persetujuan dibuatnya sebuah aplikasi.

Tahap *user design* dan *construction* merupakan 2 tahap yang sangat berhubungan dimana pada tahap ini pembuat aplikasi harus menemukan rancangan yang tepat bagi aplikasinya, sehingga rancangan tersebut bisa

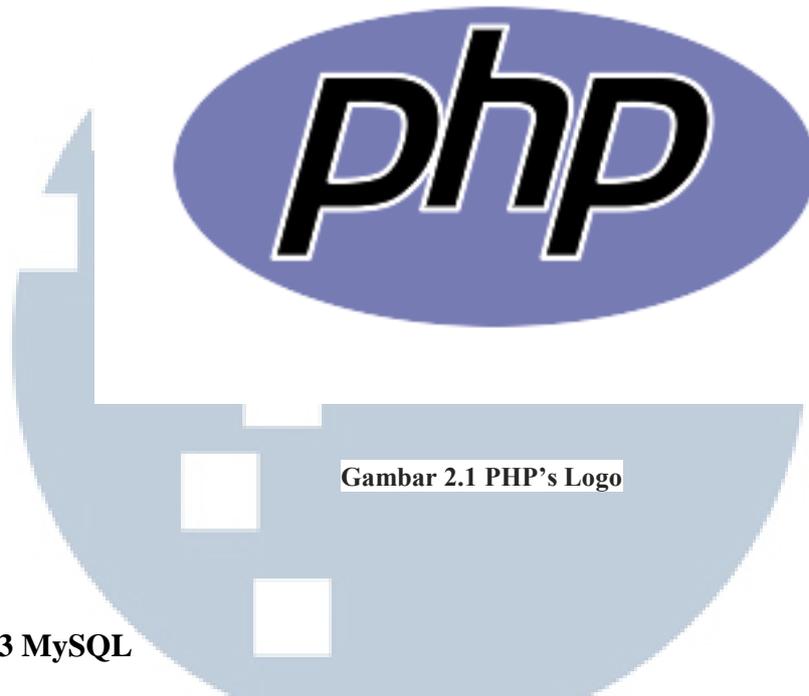
dituangkan secara nyata dalam bentuk aplikasi. Pada tahap ini juga pembuat aplikasi, dapat melakukan analisis terhadap aplikasi yang sudah dibuat, apakah sudah sesuai dengan tujuannya atau belum, apakah ada kekurangan pada aplikasi tersebut, dan lain – lain.

Tahap *cutover* merupakan tahap terakhir dari *Rapid Applications Development* (RAD), dimana aplikasi sudah *fix* dan berhasil dijalankan sebagai sebuah program.

2.2.2 PHP : Hypertext Preprocessor (PHP)

Berdasarkan (w3schools.com) PHP merupakan sebuah bahasa pemrograman dari sisi server yang digunakan untuk perancangan web yang dinamis serta interaktif, yang digunakan secara bebas, serta sebagai bahasa pemrograman yang efisien bila dihadapkan terhadap pesaing seperti Microsoft ASP.

Menurut (Oracle, 1997) PHP juga dapat digunakan untuk membuat halaman web yang dinamis, serta bersifat fleksibel untuk digunakan untuk rata-rata sistem operasi maupun *web server*. PHP dapat digunakan pula sebagai kunci pengakses *database*, dan salah satunya adalah MySQL.



Gambar 2.1 PHP's Logo

2.2.3 MySQL

Sistem manajemen *database* merupakan sebuah sistem yang membantu perusahaan untuk mengatur banyaknya informasi maupun data yang diproses hampir setiap hari didalam perusahaan tersebut.

Menurut (Tahaghoghi, Williams, 2007) dalam beberapa tahun terakhir, telah ditemukan MySQL sebagai sistem manajemen *database* serta PHP, dan bahasa pemrograman Perl sebagai platform yang tepat untuk digunakan dalam pembuatan aplikasi sungguhan.

MySQL awalnya dibuat pada tahun 1995 oleh sebuah perusahaan di Swedia, yaitu My SQL AB oleh David Axmark, Allan Larsson, dan Michael Widenius. Namun sejak tahun 2009, perusahaan Oracle mengakuisisi perusahaan tersebut. Bahasa pemrograman yang digunakan didalam MySQL cukup luas antara lain adalah C, C++, C#, Java, PHP, dan lain sebagainya.



Gambar 2.2 MySQL's Logo

Database MySQL umumnya dikenal dengan SQLite. SQLite merupakan paket perangkat lunak yang bersifat public domain yang menyediakan sistem manajemen basis data relasional atau RDBMS (Kreibich, 2010). Sistem basis data relasional digunakan sebagai tempat penyimpanan *record* yang didefinisikan oleh pengguna pada ukuran tabel yang besar dan memproses perintah *query* yang kompleks dan menggabungkan data dari berbagai tabel untuk menghasilkan laporan dan rangkuman data. Kata 'Lite' pada SQLite tidak merujuk pada kemampuannya, akan tetapi merujuk pada sifat dari SQLite, yaitu ringan ketika dihubungkan dengan kompleksitas pengaturan, *administrative overhead*, dan pemakaian dengan sumbernya. SQLite memiliki fitur-fitur sebagai berikut :

1. Tidak memerlukan server

Arsitektur SQLite tidak memiliki arsitektur *client* server. Kebanyakan sistem *database* skala besar memiliki paket server yang besar yang membentuk mesin *database*.

2. *Single File Database*

SQLite mengemas seluruh *database* ke dalam suatu single file. Single file tersebut berisi layout *database* dan data aktual yang berada pada tabel dan indeks yang berbeda. Format file dapat digunakan pada banyak platform dan dapat diakses pada mesin manapun tanpa memperhatikan native byte order ataupun ukuran kata. Pengemasan *database* kedalam suatu file tunggal memudahkan pengguna untuk membuat, menyalin, ataupun mem-*backup image database* yang berada di dalam media penyimpanan.

3. *Zero Configuration SQLite*

Tidak membutuhkan apapun untuk melakukan instalasi dan konfigurasi. Dengan mengeliminasi server dan menggabungkan *database* secara langsung ke dalam aplikasi, maka pengguna tidak perlu mengetahui bahwa mereka sedang menggunakan *database*.

4. *Embedded Device Support*

Ukuran *code* dari SQLite bersifat kecil dan penggunaan sumber daya yang konservatif membuatnya cocok digunakan untuk *embedded system* yang berjalan terbatas pada sistem operasi.

5. Fitur-fitur yang unik

SQLite menggunakan sistem dengan tipe dinamis untuk tabel-tabel. SQLite memungkinkan pengguna untuk memasukkan nilai ke dalam kolom tanpa memperhatikan tipe data. Pada beberapa cara pemakaiannya, sistem yang bertipe dinamis pada SQLite mirip dengan sistem yang ditemukan pada bahasa scripting yang populer, yang sering memiliki sebuah tipe skalar yang dapat menerima semua tipe data dari integer sampai string. Fitur lainnya adalah kemampuan untuk memanipulasi lebih dari satu basis data pada satu waktu. SQLite mempunyai kemampuan dalam menghubungkan sebuah koneksi *database* tunggal dengan banyak file basis data secara bersamaan. Hal ini memungkinkan SQLite untuk memproses SQL statement yang menjembatani beberapa basis data sekaligus.

6. *Compatible License*

SQLite dan SQLite *code* tidak memiliki lisensi pengguna dan tidak dilindungi oleh GNU's Not Unix (GNU) General Public

License (GPL) atau lisensi *open source* sejenisnya. Hal ini berarti pengguna dapat melakukan apapun dengan *source code* SQLite, sehingga *library code* dapat digunakan dengan berbagai cara, dimodifikasi dengan berbagai cara dan didistribusikan dengan berbagai cara.

7. *Highly reliable*

Sejumlah tes telah dilakukan sebelum *library* SQLite masing-masing dirilis. Hal ini dilakukan untuk mempertahankan tingkat kehandalan yang tinggi.

2.2.4 HyperText Markup Language 5 (HTML5)

Menurut (Johan, 2013) *HyperText Markup Language* (HTML) merupakan standar yang digunakan secara luas dalam menampilkan halaman web. Hingga saat ini, HTML digunakan sebagai standar internet yang diartikan maupun dikendalikan penggunaannya melalui *World Wide Web*. Tanpa adanya HTML, internet masih dalam keadaan berbasis *text*, dimana sebuah halaman web hanya berisikan format *text* datar tanpa adanya format dokumen berbasis visual secara menarik.

Pengertian HTML sendiri bila dijabarkan berdasarkan kata-kata penyusunannya, dapat dijabarkan lebih dalam lagi menjadi :

- *Hypertext*

Link hypertext merupakan kata yang dapat menunjukkan relasi suatu naskah dokumen dengan naskah lainnya. Bila kita mengklik pada kata untuk mengikuti link tersebut, maka sebuah *web browser* akan menampilkan bagian dari dokumen yang dikehendaki.

- *Markup*

Sesuai definisinya, *markup* menunjukkan bahwa sebuah file HTML memiliki suatu intruksi tertentu yang dapat memberikan suatu format pada dokumen yang akan ditampilkan pada *World Wide Web*.

- *Language*

HTML dapat berupa sebuah kumpulan dari beberapa intruksi yang dapat digunakan untuk mengubah-ubah format suatu dokumen.

HTML sendiri sudah memiliki banyak versi. Versi terbaru dari HTML saat ini adalah HTML5. HTML5 merupakan versi terbaru teknologi *hypertext* yang saat ini sedang dalam tahap *development*. Berikut adalah beberapa tujuan dibutanya HTML5:

- Fitur baru harus didasarkan pada HTML, CSS, DOM, dan JavaScript.
- Mengurangi kebutuhan plugin eksternal (contoh: Adobe Flash).
- Penanganan kesalahan yang lebih baik.
- Lebih banyak *markup* untuk menggantikan scripting.
- HTML5 merupakan perangkat mandiri.

Saat ini HTML5 sudah mulai diterapkan pada beberapa situs besar sebagai *test* kelayakan fitur serta teknologi baru yang masih terus

dikembangkan dari versi sebelumnya yaitu HTML 4.01. Fitur serta perbaikan yang ditambahkan dalam teknologi ini diantaranya adalah video dan audio, *database* lokal dan CSS3. Berikut ini adalah beberapa *browser* yang dapat mendukung HTML5 yaitu:

- Opera Web Browser
- Safari
- FireFox
- Google Chrome
- Internet Explorer

2.2.5 Google API

Menurut (Google Maps API) Google API merupakan sebuah aplikasi yang memberikan pengguna akses penuh ke *database* Google di seluruh dunia yang dapat memberikan rekomendasi tempat menarik, memberikan pengguna informasi mengenai rute terdekat dalam mencapai suatu tujuan, serta memberikan street view suatu lokasi yang akan dikunjungi oleh pengguna walaupun pengguna belum sampai di tempat tujuannya tersebut .

Menurut (Junus, 2012), Google Maps/ API diterapkan pada teknologi GPS yang digunakan untuk melakukan pelacakan kendaraan, dengan cara komunikasi GPRS serta membuat parsing data dari data yang diterima oleh jaringan dari perangkat GPS *tracker* tersebut. Hasil dari sinkronisasi informasi data hasil parsing serta Google Maps, dapat dijalankan menggunakan Google Maps dengan javascript.

2.3 Aplikasi

Menurut (Simarmata, 2006:112) pengertian tentang aplikasi berasal dari Bahasa Inggris yaitu “*to applicate*” yang artinya menerapkan atau terapan. Namun pengertian mengenai aplikasi secara umum adalah suatu paket program yang sudah jadi dan dapat digunakan. Sedangkan arti aplikasi adalah “program komputer yang dibuat untuk menolong manusia dalam melaksanakan tugas tertentu”.

Sedangkan menurut (Kamus Besar Bahasa Indonesia) aplikasi merupakan penerapan dari rancangan sistem untuk mengolah data yang menggunakan aturan atau ketentuan bahasa pemrograman tertentu, sesuai dengan jenis aplikasi yang akan dibuat tersebut.

2.4 UML

Menurut (Schmitt, 2002) *Unified Modelling Language* (UML) adalah bahasa untuk menggambarkan, menspesifikasikan, membangun, dan mendokumentasikan sistem perangkat lunak yang berorientasi pada objek. UML telah diterima secara luas sebagai standar untuk sistem perangkat lunak modeling dan didukung oleh banyak bantuan perangkat *Computer Aided Software Engineering Tools* (CASE).

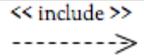
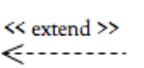
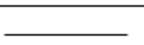
Menurut (Hermawan, 2014) *Unified Modelling Language* atau dikenal dengan UML, merupakan sebuah metode permodelan visual sebagai alat untuk merancang sistem berorientasi objek, atau dapat diartikan sebagai sebuah bahasa pemrograman yang sudah menjadi standar secara visual, perancangan serta

pendokumentasian sistem perangkat lunak. UML sendiri terdiri dari beberapa jenis, yang terdiri dari:

- *Use Case Diagram*

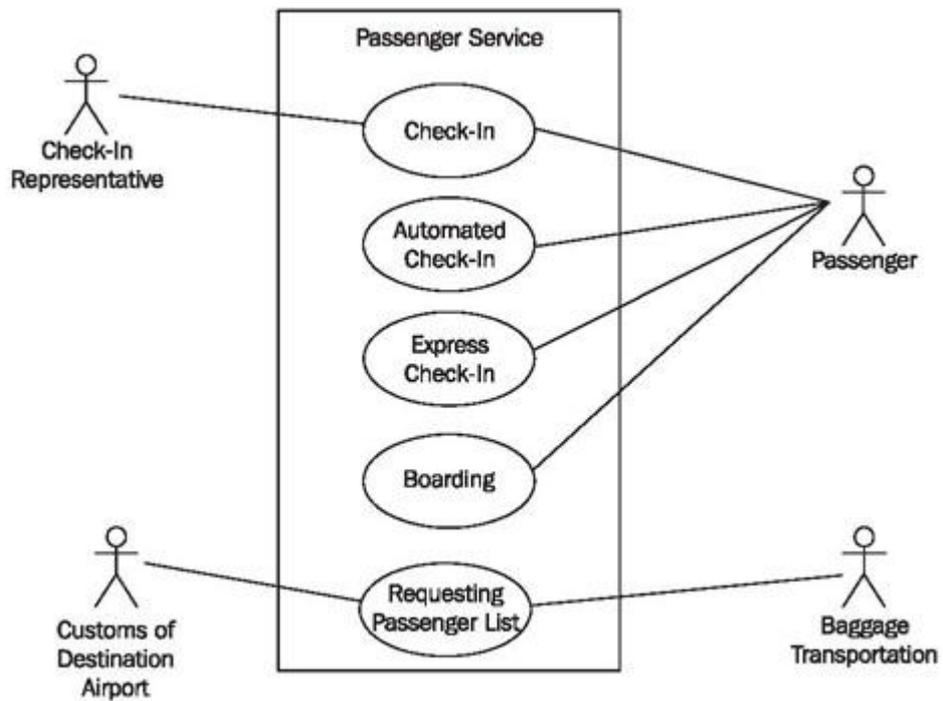
Use case diagram merupakan sebuah jenis diagram pada UML yang menggambarkan interaksi antara sistem dengan aktor atau penggunanya.

Tabel 2.1 Keterangan Simbol *Use Case Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara eksplisit.
3		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
4		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
5		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
6		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor

Use case diagram umumnya bersifat menjelaskan secara detail, tipe interaksi antara si *user* dengan sistemnya.

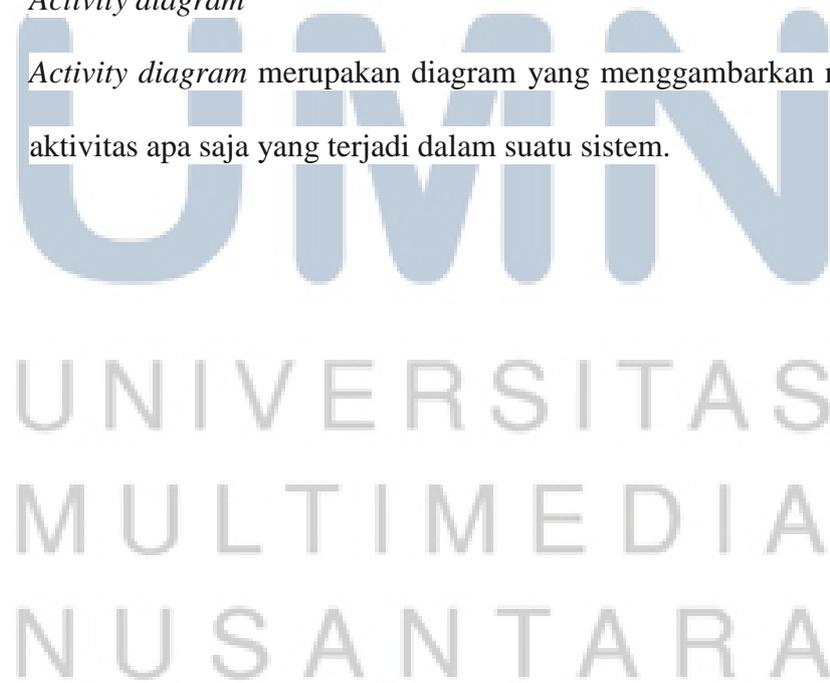
UNIVERSITAS
MULTIMEDIA
NUSANTARA



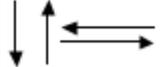
Gambar 2.3 Contoh Use Case Diagram

- *Activity diagram*

Activity diagram merupakan diagram yang menggambarkan model-model aktivitas apa saja yang terjadi dalam suatu sistem.

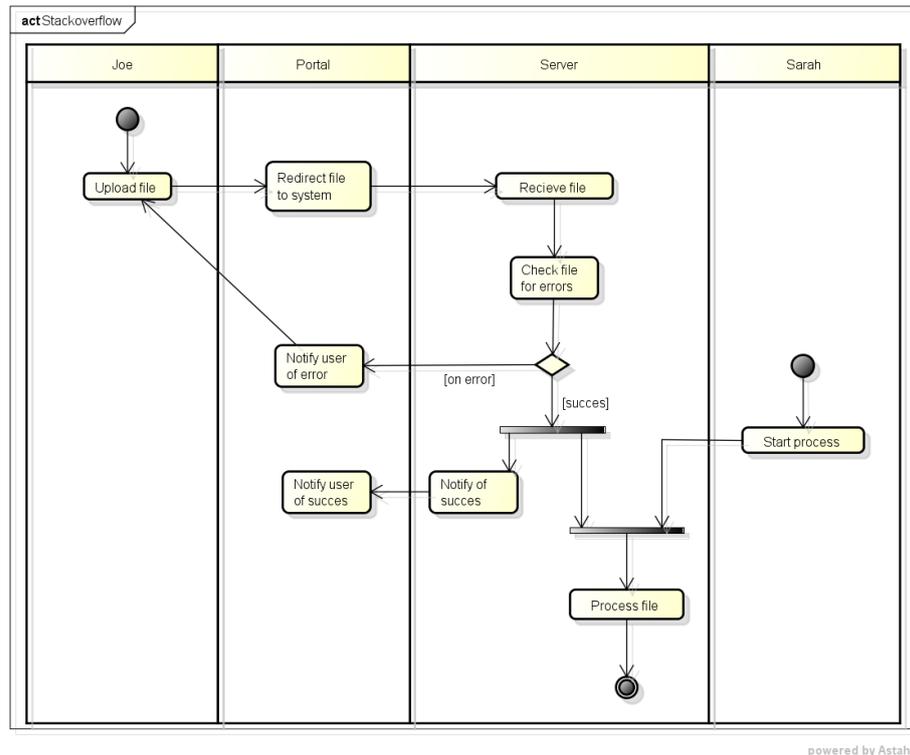


Tabel 2.2 Simbol *Activity Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		Activity	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		Action	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		Initial Node	Bagaimana objek dibentuk atau diawali.
4		Activity Final Node	Bagaimana objek dibentuk dan diakhiri
5		Decision	Digunakan untuk menggambarkan suatu keputusan / tindakan yang harus diambil pada kondisi tertentu
6		Line Connector	Digunakan untuk menghubungkan satu simbol dengan simbol lainnya

Tabel 2.2 bertujuan untuk menjelaskan masing-masing simbol yang terdapat pada *activity diagram* untuk memudahkan user dalam memahami alur atau proses yang terdapat dalam suatu *activity diagram* tersebut.

UMMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 2.4 Contoh Activity Diagram

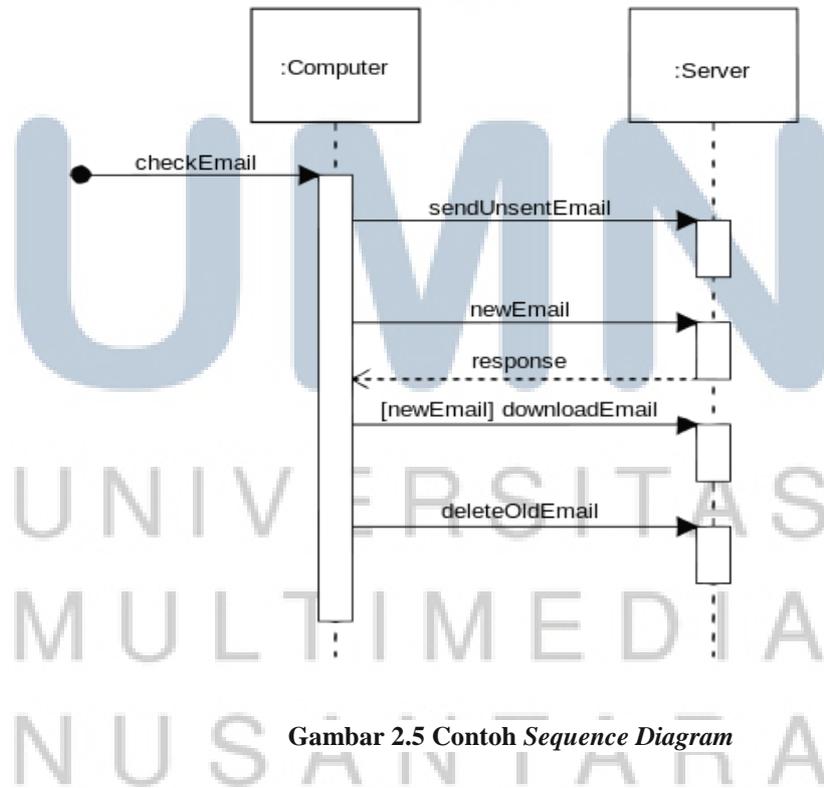
Gambar 2.4 merupakan contoh dari *activity diagram*, dimana diagram tersebut menjelaskan masing-masing proses beserta pihak-pihak yang terlibat dalam proses-proses tersebut.

- *Sequence Diagram*

Sequence diagram merupakan jenis diagram pada UML yang mendeskripsikan interaksi antar objek berdasarkan urutan waktu. *Sequence diagram* sendiri juga memiliki fungsi untuk menggambarkan langkah yang harus dilakukan untuk dapat menghasilkan sesuatu.

Tabel 2.3 Simbol *Sequence Diagram*

NO	GAMBAR	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

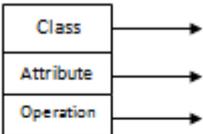
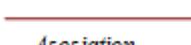
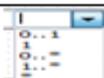


Gambar 2.5 Contoh *Sequence Diagram*

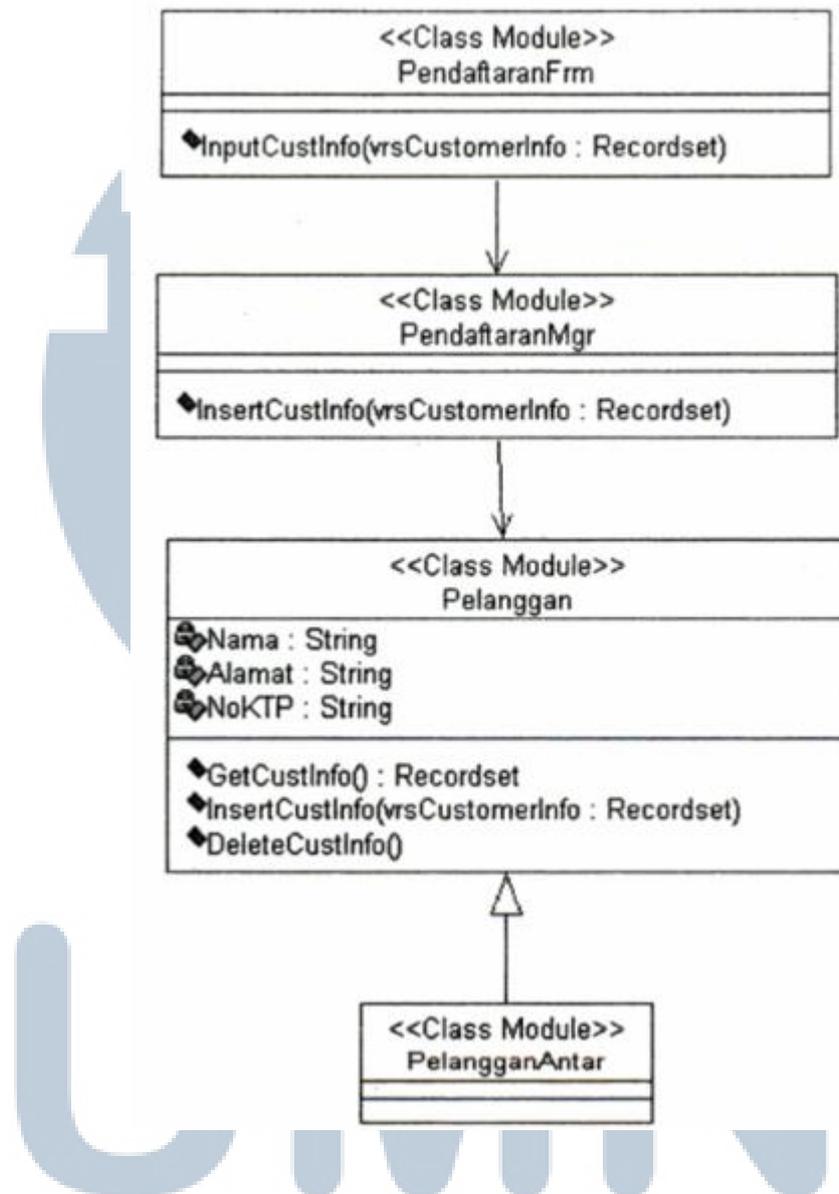
Gambar 2.5 menunjukkan sebuah proses yang dilakukan oleh kedua pihak yang terlibat dalam proses tersebut, sehingga pembaca *sequence diagram* ini dapat mengerti peran masing-masing pihak secara detail dalam suatu proses.

- **Class Diagram**

Tabel 2.4 Simbol Class Diagram

No.	Simbol	Fungsi
1		<p>1. <i>Class Name</i></p> <p>2. <i>Attribute</i> adalah properti dari sebuah class. <i>Attribute</i> ini melukiskan batas nilai yang mungkin ada pada objek dari class. Sebuah class mungkin mempunyai nol atau lebih <i>attribute</i>.</p> <p>3. <i>Operation</i> adalah sesuatu yang dapat dilakukan oleh sebuah class atau yang anda (class yang lain) dapat lakukan untuk sebuah class.</p>
2		<p>Dapat diartikan sebagai relasi. Digambarkan sebagai garis lurus antara dua kelas. Namun tidak berarti bahwa kelas satu memiliki kelas yang lain, tetapi kelas yang lain dapat berelasi juga dengan kelas yang sama.</p>
3		<p><i>Cardinality</i> atau <i>multiplicity</i> adalah simbol yang menunjukkan jumlah keterhubungan dari satu kelas dengan kelas lainnya.</p>

Class diagram merupakan sebuah diagram pada UML yang berfungsi untuk menampilkan kelas-kelas yang ada pada suatu sistem yang nantinya akan digunakan, sehingga *class diagram* memungkinkan untuk memberikan gambaran sistem maupun relasi-relasi yang terdapat dalam sistem tersebut



Gambar 2.6 Contoh *Class Diagram*

UNIVERSITAS
MULTIMEDIA
NUSANTARA

2.5 Interface

Menurut (*Oxford American Dictionaries*) *interface* adalah cara dari sebuah program komputer dalam menyajikan informasi kepada pengguna atau menerima informasi dari pengguna, terutama tata letak layar dan menu. Berdasarkan hal tersebut, maka *User Interface* adalah bentuk tampilan grafis yang berhubungan langsung dengan pengguna. *User Experience* buku “Ergonomics of Human System Interaction” dijelaskan UX adalah persepsi dan respon seseorang yang dihasilkan dari penggunaan dari sebuah produk, sistem, ataupun jasa. Kemudian *User Centered Design* merupakan paradigma baru dalam pengembangan sistem berbasis web. Perancangan berbasis pengguna UCD adalah istilah yang digunakan untuk menggambarkan filosofi perancangan. Konsep dari UCD adalah pengguna sebagai pusat dari proses pengembangan sistem, dan tujuan/sifat-sifat, konteks dan lingkungan sistem semua didasarkan dari pengalaman pengguna.

2.6 User Interface (UI)

Menurut (Fadeyev, 2009) *user interface* merupakan interaksi yang tercipta antara pengguna dengan aplikasi. Hal ini berarti, *user interface* bukan sekedar mengenai penampilan produk, tetapi lebih kepada produk tersebut bekerja. Fadeyev menjelaskan hal-hal yang perlu diperhatikan untuk

merancang sebuah user *interface* yang baik dan berkualitas adalah sebagai berikut :

- *Concision*
Tampilan yang ringkas, padat, dan mudah dimengerti sehingga tidak membingungkan pengguna
- *Familiarity*
Tampilan yang akrab dengan cara menggunakan elemen umum yang mudah dipahami pengguna.
- *Responsiveness*
Memiliki respon atau ketanggapan yang baik dalam memberikan umpan balik kepada pengguna.
- *Efficiency*
Menerapkan efisiensi yang tinggi dan menjunjung tinggi produktivitas.
- *Clarity*
Menghindari sifat ambigu, dengan memperjelas, bahasa, alur, serta hirarki tampilan tersebut.
- *Consistency*
Menggunakan elemen-elemen yang terpola dengan baik secara konsisten.
- *Aesthetics*
Tampilan yang indah, sehingga membuat pengguna tertarik dan nyaman dalam menggunakannya.
- *Forgiveness*

Menyediakan fasilitas untuk pengguna ketika melakukan kesalahan, misal ketika pengguna salah menekan tombol, user *interface* yang baik menyediakan tombol untuk undo atau back.

2.7 GPS

Menurut (Verma, 2013) GPS merupakan salah satu aplikasi yang digunakan untuk melakukan *tracking* pada kendaraan ataupun *handphone* seseorang, yang memungkinkan posisi dan informasi dari keberadaan seseorang tersebut diketahui melalui pengontrol lokasi lainnya. Selain itu GPS juga dapat digunakan sebagai aplikasi berbasis web, sehingga melalui aplikasi tersebut lokasi sesungguhnya dari sebuah target dapat diketahui.

Menurut (Moore, 1995) GPS telah menjadi alat bantu navigasi yang digunakan di seluruh dunia. GPS juga menjadi alat yang sangat berguna dalam pembuatan peta, survei lahan, aplikasi komersial, apalagi dalam penerapan ilmu sains. GPS dapat juga digunakan sebagai referensi waktu yang akurat yang dipakai dalam penelitian gempa bumi dan sinkronisasi jaringan telekomunikasi.

2.8 Tracking

Menurut (Verma, 2013) *tracking* merupakan sebuah sistem yang dapat melacak keberadaan sebuah target dengan tepat. *Tracking* system umumnya dilengkapi dengan GPS maupun teknologi GSM yang mampu memberikan

informasi maupun rute yang dilalui oleh target, seperti kendaraan maupun pengguna handphone.

2.9 Resi

Berdasarkan (Kamus Besar Bahasa Indonesia), resi merupakan surat tanda terima, yang diterima oleh konsumen sebagai bukti pembayaran maupun penggunaan jasa. Nomor resi pada ekspedisi umumnya digunakan sebagai nomor yang dapat dilacak oleh konsumen yang ingin mengetahui keberadaan barangnya tersebut. Hal ini berguna untuk menghindari akan adanya penipuan, maupun sebagai jaminan agar barang kirimannya dapat dijaga sehingga tidak rusak, serta aman sampai tempat tujuannya.

2.10 Ekspedisi

Menurut (Kamus Besar Bahasa Indonesia) ekspedisi merupakan proses pengiriman suatu barang, surat, dokumen berharga, dan lainnya. Ekspedisi sendiri di Indonesia sudah memiliki beberapa perusahaan terkenal seperti JNE, TIKI, Tikindo, dan perusahaan lainnya. Tentunya setiap ekspedisi memiliki tarif pengantaran yang berbeda setiap wilayahnya maupun antar ekspedisi, dikarenakan masing-masing ekspedisi memiliki rute antar maupun estimasi kedatangan barang yang berbeda pada masing-masing wilayah pengantaran.

2.11 Database

Menurut (Conolly, Begg, 2005: p14) *database* adalah kumpulan data yang mempunyai keterhubungan secara logis dan saling berinteraksi serta menghasilkan informasi yang dibutuhkan. Suatu *database* haruslah merupakan sebuah penyimpanan data besar yang dapat digunakan oleh berbagai pengguna atau bagian organisasi dalam waktu yang bersamaan.

Database adalah kumpulan elemen data yang saling terkait secara logika. Sedangkan menurut (Turban, Reiner, Potter, 2003: p19) *database* adalah kumpulan berkas atau arsip yang tersusun dan saling terkait membuat data dan lainnya yang tersimpan disuatu tempat.

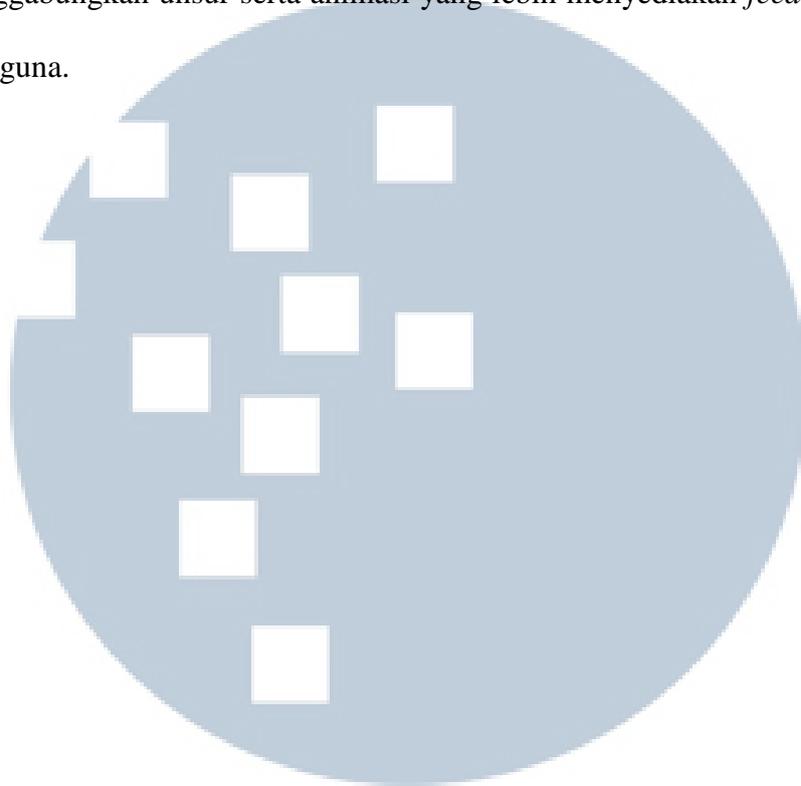
2.12 Website

Menurut (Arief, 2011) web adalah salah satu aplikasi yang berisikan dokumen-dokumen multimedia (teks, gambar, suara, animasi, video) di dalamnya yang menggunakan protokol HTTP (*Hypertext transfer protocol*) dan untuk mengaksesnya menggunakan perangkat lunak yang disebut *browser*.

2.13 Cascading Style Sheets (CSS)

Menurut (W3schools.com) CSS mendeskripsikan bagaimana unsur-unsur dalam HTML ditampilkan dalam layar, kertas, maupun media lainnya. Dalam CSS, terdapat sebuah tools bernama materialize CSS, dimana menurut (materializecss.com) tools ini berguna untuk menciptakan kerangka yang

menggabungkan unsur serta animasi yang lebih menyediakan *feedback* kepada pengguna.



UMMN

UNIVERSITAS
MULTIMEDIA
NUSANTARA