

BAB II

TINJAUAN PUSTAKA

2.1. Tinjauan Teori Umum

2.1.1. Sistem

Menurut Husda (2012:111), sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau menyelesaikan suatu sasaran tertentu.

Menurut Husda (2012:112), sistem merupakan seperangkat unsur yang saling terikat dalam suatu antar relasi diantara unsur-unsur tersebut dengan lingkungan.

Menurut Husda (2012:112), sistem adalah setiap kesatuan secara konseptual atau fisik yang terdiri dari bagian-bagian dalam keadaan saling tergantung satu sama lainnya.

Menurut Husda (2012:112), sistem adalah prosedur logis dan rasional untuk merancang suatu rangkaian komponen yang berhubungan satu dengan yang lainnya dengan maksud untuk berfungsi sebagai satu kesatuan dalam usaha mencapai suatu tujuan yang telah ditentukan.

Menurut Husda (2012:112), sistem adalah himpunan dari unsur-unsur yang saling berkaitan sehingga membentuk suatu kesatuan yang utuh dan terpadu.

Menurut Husda (2012:112), sebuah sistem terdiri dari bagian-bagian yang saling berkaitan yang beroperasi bersama untuk mencapai beberapa sasaran atau maksud.

2.1.2. Karakteristik Sistem

Menurut Fatta (2007: 5), untuk memahami atau mengembangkan suatu sistem, maka perlu membedakan unsur-unsur dari sistem yang membentuknya. Berikut adalah karakteristik sistem yang dapat membedakan suatu sistem dengan sistem lainnya:

1. Batasan (*Boundary*).

Penggambaran dari suatu elemen atau unsur mana yang termasuk di dalam sistem dan mana yang di luar sistem.

2. Lingkungan (*Environment*).

Segala sesuatu diluar sistem, lingkungan yang menyediakan asumsi, kendala, dan input terhadap suatu sistem.

3. Masukkan (*Input*).

Sumber daya (data, bahan baku, peralatan, energi) dari lingkungan yang dikonsumsi dan dimanipulasi oleh suatu sistem.

4. Keluaran (*Output*).

Sumber daya atau produk (informasi, laporan, dokumen, tampilan layar computer, barang jadi) yang disediakan untuk lingkungan sistem oleh kegiatan dalam suatu sistem.

5. Komponen (*Component*).

Kegiatan-kegiatan atau proses dalam suatu sistem yang mentransformasikan input menjadi bentuk setengah jadi (*output*), komponen ini bisa merupakan subsistem dari sebuah sistem.

6. Penghubung (*Interface*).

Tempat dimana komponen atau sistem dan lingkungannya bertemu atau berinteraksi.

7. Penyimpanan (*storage*).

Area yang dikuasai dan digunakan untuk penyimpanan sementara dan tetap dari informasi, energi, bahan baku, dan sebagainya. Penyimpanan merupakan suatu media penyangga diantara komponen tersebut bekerja dengan tinkleatan yang ada dan memungkinkan komponen yang berbeda dari berbagai data yang sama.

2.1.3. Informasi

Menurut Husda (2012:117), data yang telah diolah menjadi suatu bentuk yang penting bagi si penerima dan mempunyai nilai yang nyata yang dapat dirasakan dalam keputusan-keputusan yang sekarang atau keputusan-keputusan yang akan datang.

Menurut Husda (2012:117), Informasi adalah data yang telah diolah menjadi bentuk yang memiliki arti bagi si penerima dan bermanfaat bagi pengambil keputusan saat ini atau mendatang.

Secara umum informasi dapat didefinisikan sebagai hasil dari pengolahan data dalam suatu bentuk yang lebih berguna dan lebih berarti bagi penerimanya yang menggambarkan suatu kejadian-kejadian yang nyata yang digunakan untuk pengambilan keputusan (Husda 2012:117).

2.1.3.1. Kualitas Informasi

Sebuah sistem informasi harus mampu mengumpulkan data dan mentransformasikan data tersebut kedalam informasi yang memiliki kualitas-kualitas tertentu, berikut karakteristik informasi yang berkualitas (Kusrini, 2007:4):

1. Relevan. Informasi yang disajikan sebaiknya terkait dengan keputusan yang akan diambil oleh pengguna informasi tersebut.
2. Akurat. Kecocokan antara informasi dengan kejadian-kejadian atau objek-objek yang diwakilinya.
3. Lengkap. Merupakan derajat sampai seberapa jauh informasi menyertakan kejadian-kejadian atau objek-objek yang berhubungan.
4. Tepat waktu. Informasi yang tidak tepat waktu akan menjadi informasi yang tidak berguna atau tidak dapat digunakan untuk membantu mengambil keputusan.
5. Dapat dipahami. Hal tersebut terkait dengan bahasa dan cara penyajian informasi agar pengguna lebih mudah mengambil keputusan.

6. Dapat dibandingkan. Sebuah informasi yang memungkinkan seorang pemakai untuk mengidentifikasi persamaan dan perbedaan antara dua objek atau kejadian yang mirip.

2.1.4. Sistem Informasi

Menurut Husda (2012:119), Sistem informasi merupakan sistem yang mempunyai kemampuan untuk mengumpulkan berbagai media untuk menampilkan informasi.

Sistem informasi dapat didefinisikan sebagai suatu sistem di dalam suatu organisasi yang merupakan kombinasi dari orang-orang, fasilitas, teknologi, media prosedur-prosedur dan pengendalian yang ditujukan untuk mendapatkan jalur komunikasi penting, memproses tipe transaksi rutin tertentu, memberi sinyal kepada manajemen dan yang lainnya terhadap kejadian-kejadian internal dan eksternal yang penting dan menyediakan suatu dasar informasi untuk pengambilan keputusan (Husda, 2012:120).

Definisi Sistem Informasi menurut Azhar Susanto (2008), adalah kumpulan dari subsistem apapun baik fisik ataupun non fisik yang saling berhubungan satu sama lain dan bekerja sama secara harmonis untuk mencapai satu tujuan yaitu mengolah data menjadi informasi yang berarti dan berguna.

2.1.4.1. Komponen Sistem Informasi.

Fatta (2007: 9), menjelaskan bahwa sistem informasi berbasis komputer dalam suatu organisasi terdiri dari komponen-komponen sebagai berikut :

1. Perangkat Keras

Komponen untuk melengkapi kegiatan masukan data, memproses data, dan keluaran data.

2. Perangkat lunak

Yaitu program dan instruksi yang diberikan ke komputer.

3. Database

Yaitu kumpulan data dan informasi yang diorganisasikan sedemikian rupa sehingga mudah diakses pengguna sistem informasi.

4. Telekomunikasi

Yaitu komunikasi yang menghubungkan antara pengguna sistem dengan sistem komputer secara bersama-sama kedalam suatu jaringan kerja yang efektif.

5. Manusia

Yaitu personel dari sistem informasi, meliputi manajer, analisis, programmer, dan operator, serta bertanggung jawab terhadap sistem.

Sementara Burch dan Grudnitski berpendapat dalam Fatta (2007: 10), Sistem informasi terdiri dari komponen-komponen antara lain sebagai berikut:

1. Blok masukan

Input mewakili data yang masuk ke dalam sistem informasi. Input disini termasuk metode-metode dan media untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen-dokumen dasar.

2. Blok model

Blok ini terdiri dari kombinasi prosedur, logika dan model matematik yang akan memanipulasi data input dan data yang tersimpan di basis data dengan cara yang sudah tertentu untuk menghasilkan keluaran yang diinginkan.

3. Blok keluaran

Produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

4. Blok teknologi

Teknologi merupakan *tool box* dalam sistem informasi. Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu mengendalikan dari sistem secara keseluruhan.

5. Blok *database*

Merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan di perangkat keras *computer* dan digunakan perangkat lunak untuk memanipulasinya.

6. Blok kendali

Pengendalian perlu dirancang dan diterapkan untuk menyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah ataupun bila terlanjur terjadi kesalahan-kesalahan dapat langsung cepat diatasi.

2.2. Tinjauan Teori Khusus

2.2.1. Sistem Pendukung Keputusan

Menurut Kusri (2007), *Decision support system (DSS)* merupakan sistem informasi interaktif yang menyediakan informasi, pemodelan, dan manipulasi data. Sistem itu digunakan untuk membantu pengambilan keputusan dalam situasi yang semiterstruktur dan situasi yang tidak terstruktur, dimana tidak seorang pun tahu secara pasti bagaimana keputusan seharusnya dibuat.

Tujuan dari *Decision support system (DSS)* adalah, (Kusri 2007) :

1. Membantu manajer dalam pengambilan keputusan atau masalah semiterstruktur.
2. Memberikan dukungan atas pertimbangan manajer dan bukannya dimaksud untuk menggantikan fungsi manajer.
3. Meningkatkan efektifitas keputusan yang diambil manajer lebih dari pada perbaikan efisiensi.
4. Kecepatan komputasi. Computer memungkinkan para pengambil keputusan untuk melakukan banyak komputasi secara cepat dengan biaya yang rendah.
5. Peningkatan produktifitas. Membangun suatu kelompok pengambilan keputusan, terutama para pakar. Bisa sangat mahal. Pendukung terkomputerisasi bisa mengurangi ukuran kelompok dan memungkinkan para anggotanya untuk berada di berbagai lokasi yang berbeda-beda (menghemat biaya perjalanan).

6. Dukungan kualitas. *Computer* bisa meningkatkan kualitas keputusan yang dibuat. Sebagai contoh, semakin banyak data yang diakses, maka semakin banyak juga alternatif yang bisa di evaluasi.
7. Berdaya saing, Teknologi pengambilan keputusan bisa menciptakan pemberdayaan yang signifikan dengan cara memperoleh seseorang untuk membuat keputusan yang baik secara cepat, bahkan jika mereka memiliki pengetahuan yang kurang.
8. Mengatasi keterbatasan kognitif dalam pemrosesan dan penyimpanan. Menurut Simon (1997) dalam Kusri (2007), otak manusia memiliki kemampuan yang terbatas untuk memproses dan menyimpan informasi. Orang-orang kadang sulit mengingat dan menggunakan sebuah informasi dengan cara yang bebas dari kesalahan.

2.2.2. Pariwisata

Wisata menurut Yoeti (2006:12), kepariwisataan didefinisikan sebagai kegiatan perjalanan yang dilakukan sukarela dan bersifat sementara, serta perjalanan itu sebagian atau seluruhnya bertujuan untuk menikmati objek dan daya tarik wisata.

Sedangkan pariwisata adalah segala sesuatu yang berhubungan dengan wisata, termasuk pengusaha objek dan daya tarik wisata serta usaha-usaha yang terkait dibidang tersebut. Seringkali, pariwisata hanya dilihat dalam bingkai ekonomi, padahal ia merupakan rangkaian dari kekuatan ekonomi, lingkungan

dan social budaya yang bersifat global. Namun pariwisata dapat juga memberikan manfaat dan menyumbang antara lain kepada:

1. Pelestarian budaya dan adat istiadat;
2. Peningkatan kecerdasan masyarakat;
3. Peningkatan kesehatan dan kesegaran;
4. Terjaganya sumber daya alam dan lingkungan lestari;
5. Terpeliharanya penginggalan kuno dan warisan masa lalu; dll.

Kegiatan pariwisata sendiri adalah salah satu ungkapan budaya. Mungkin tidak semua kebudayaan mempunyai konsep yang sama mengenai kegiatan berwisata, bahkan mungkin ada yang sama sekali tidak mengenal pariwisata sebagai sebuah upaya yang sengaja, yang bertujuan untuk mendapatkan suatu pengalaman khusus ditempat lain, (Yoeti, 2006:22).

2.2.2.1. Potensi Pariwisata Indonesia

Bangsa Indonesia adalah bangsa yang majemuk, karena terdiri lebih dari empat ratus suku bangsa, dan juga tempat berkembangnya lima agama besar dunia, disamping kepercayaan lain yang sudah mengakar ke dalam kultur social masyarakat Indonesia (Yoeti, 2006:114). Hal ini wajar saja mengingat wilayah Nusantara yang strategis yaitu terletak di jalur silang antara dua samudra dan dua benua, dan terdiri lebih kurang tiga ribu pulau yang berserakan di wilayah khatulistiwa sepanjang tiga ribu mil dari timur ke barat dan seribu mil dari utara ke selatan.

2.2.2.2. Pariwisata Sebagai Suatu Alternatif

Pemerintah harus mencari alternatif sektor ekonomi yang dianggap pas untuk mempercepat penanggulangan kemiskinan di Indonesia. Salah satu sektor ekonomi yang dianggap cukup prospektif adalah sektor pariwisata.

Sektor pariwisata diyakini tidak hanya sekedar mampu menjadi sektor andalan dalam usaha meningkatkan perolehan devisa untuk pembangunan yang sekarang sedang giat-giatnya dilakukan pemerintah, akan tetapi juga mengentaskan kemiskinan (Yoeti, 2006:14). Dalam GBHN tahun 1993 dikatakan:

Pembangunan pariwisata diarahkan pada peningkatan pariwisata menjadi sektor andalan yang mampu menggalakkan kegiatan ekonomi, termasuk sektor-sektor lainnya yang terkait, sehingga lapangan kerja, pendapatan masyarakat, pendapatan daerah, pendapatan Negara serta penerimaan devisa meningkat melalui pengembangan dan pendayagunaan potensi kepariwisataan nasional.

2.2.2.3. Dampak Positif Pariwisata

Dilihat dari kacamata ekonomi makro, jelas pariwisata memberikan dampak positif, karena sebagai suatu industri:

1. Dapat menciptakan peluang usaha. Dengan datangnya wisatawan, perlu pelayanan untuk menyediakan kebutuhan, keinginan, dan harapan wisatawan yang terdiri dari berbagai kebangsaan dan tingkah lakunya.
2. Dapat meningkatkan kesempatan kerja

3. Dapat meningkatkan pendapatan sekaligus mempercepat pemerataan pendapatan masyarakat, sebagai akibat *multiplier effect* yang terjadi dari pengeluaran wisatawan yang relatif cukup besar.
4. Dapat meningkatkan penerimaan pajak pemerintah dan retribusi daerah.
5. Dapat meningkatkan pendapatan nasional atau *Gross Domestic Bruto*.
6. Dapat mendorong peningkatan investasi dari sektor industri pariwisata dan sektor ekonomi lainnya
7. Dapat memperkuat neraca pembayaran, bila neraca pariwisata mengalami surplus, dengan sendirinya akan memperkuat neraca pembayaran Indonesia, dan sebaliknya.

2.2.2.4. Dampak Negatif Pariwisata

Sebegitu jauh pengembangan pariwisata di Indonesia yang menerima kedatangan wisatawan silih berganti, dari sudut sosiologi belum banyak dilakukan penelitian tentang dampak negatif sebagai akibat pengembangan pariwisata secara tidak terkendali, beberapa contoh kejadian sehari-hari dampak negatif itu antara lain:

1. Harga tanah menjadi mahal, pantai-pantai di kaveling, sehingga sering terjadi spekulasi harga yang pada akhirnya meningkatkan harga tanahnya di sekitarnya.
2. Di pusat-pusat konsentrasi kegiatan pariwisata harga-harga bahan makanan menjadi mahal yang dapat meningkatkan inflasi tiap tahunnya.

3. Sumber-sumber hayati menjadi rusak, yang menyebabkan Indonesia kehilangan daya tariknya untuk jangka panjang.
4. Terjadi urbanisasi, pencari kerja mengalir dari desa ke kota-kota besar.
5. Ramainya lalu-lintas wisatawan, ternyata ditumpukin oleh penyeludupan obat bius dan narkotika.

2.2.3. AHP (*Analytical Hierarchy Process*)

Menurut Hariman dan Rusmana (2014), *Analytical Hierarchy Process* adalah salah satu teknik pengambilan keputusan / optimasi *multivariate* yang digunakan dalam analisis kebijaksanaan. Pada hakekatnya *Analytical Hierarchy Process* merupakan suatu model pengambil keputusan yang komprehensif dengan memperhitungkan hal-hal yang bersifat kualitatif dan kuantitatif. Dalam model pengambilan keputusan dengan *Analytical Hierarchy Process* pada dasarnya berusaha menutup semua kekurangan dari model-model sebelumnya. *Analytical Hierarchy Process* juga memungkinkan struktur suatu sistem dan lingkungan ke dalam komponen yang saling berinteraksi dan kemudian menyatukan mereka dengan mengukur dampak pada komponen kesalahan sistem.

2.2.3.1. Prosedur AHP

1. Mendefinisikan masalah dan menentukan solusi yang diinginkan.
2. Membuat struktur hierarki yang diawali dengan tujuan utama.
3. Membuat matriks perbandingan, menggunakan skala rasio, dimana jika nilai matriks berada simetris dengan diagonalnya maka akan bernilai kebalikannya, atau dapat dirumuskan :

$$\{A_{ij} = 1/A_{ji}\}$$

Rumus 2.1. Rumus Matriks Perbandingan

4. Menjumlahkan setiap kolom (\sum kolom) pada matriks perbandingan
5. Normalisasi matriks, dengan membagi setiap kolom (\sum kolom), kemudian di jumlahkan setiap barisnya (\sum baris).
6. Menghitung *Total Priority Value* (TPV) untuk mendapatkan bobot subkriteria.
7. Menghitung uji konsistensi

Untuk model AHP, matriks perbandingan dapat diterima jika nilai ratio *consistency* (CR) ≤ 0.1 (10%). Tahapan dalam melakukan uji konsistensi adalah sebagai berikut :

1. Mengalikan nilai TPV dengan nilai kolom matriks pada nilai matriks perbandingan kemudian jumlahkan tiap barisnya.
2. Mencari *consistency index* (CI) dengan rumus :

$$\{CI = (\sum \text{maks} - n) / (n - 1)\}$$

Rumus 2.2. Rumus *Consistency index*

Dimana :

CR = *Consistency Index*

N = Banyaknya elemen yang dibandingkan

$\sum \text{maks}$ = *Eigen value maksimum*

3. Mencari *Consistency Ratio* (CR) dengan rumus :

$$\{CR=CI/RI\}$$

Rumus 2.3. Rumus *Consistency Ratio*

Dimana :

CR = *Consistency Ratio*

CI = *Consistency Index*

RI = *Random Index*

2.2.3.2. Nilai Skala Perbandingan

Menurut Hariman dan Rusmana, 2014. Kriteria dan alternatif dinilai melalui perbandingan berpasangan. Untuk berbagai persoalan, skala 1 sampai 9 adalah skala terbaik dalam mengekspresikan pendapat. Nilai dan definisi pendapat kualitatif dari skala perbandingan Saaty dapat dilihat pada tabel berikut :

Intensitas kepentingan	Defenisi
1	Kedua elemen sama pentingnya
2	Elemen yang satu sedikit lebih pentng dari pada elemen yang lainnya
5	Elemen yang satu lebih penting dari pada yang lainnya
7	Satu elemen jelas lebih mutlak penting dari pada elemen lainnya
9	Satu elemen mutlak penting dari pada elemen lainnya
2,4,6,8	Nilai-nilai antara dua nilai pertimbangan yang berdekatan

Tabel 2.1. Nilai Skala Perbandingan

Perbandingan dilakukan berdasarkan kebijakan pembuat keputusan dengan menilai tingkat kepentingan satu elemen terhadap elemen lainnya. Proses perbandingan berpasangan, dimulai dari level hirarki paling atas yang ditujukan untuk memilih kriteria, misalnya A, kemudian diambil elemen yang akan dibandingkan, misal A1, A2, dan A3. Maka susunan elemen-elemen yang dibandingkan tersebut akan tampak seperti pada gambar matriks di bawah ini :

Kriteria	A	B	C
A	1		
B		1	
C			1

Tabel 2.2. Tabel Hirarki

Untuk menentukan nilai kepentingan relatif antar elemen digunakan skala bilangan dari 1 sampai 9 seperti pada di atas. Penilaian ini dilakukan oleh seorang pembuat keputusan yang ahli dalam bidang persoalan yang sedang dianalisa dan mempunyai kepentingan terhadapnya. Jika pengambil keputusan memiliki pengalaman atau pemahaman yang besar mengenai masalah keputusan yang dihadapi, maka dapat langsung memasukkan pembobotan dari setiap alternatif. Jika $0 \geq \text{rasio konsistensi} \leq 0.1$, hasil perhitungan data dapat dibenarkan.

RI	0.00	0.00	0.58	0.90	1.12	1.24	1.32	1.41	1.45	1.49
N	1	2	3	4	5	6	7	8	9	10

Tabel 2.3. Pembobotan alternatif

Yang diukur dalam AHP adalah rasio konsistensi dengan melihat indeks konsistensi. Konsistensi yang diharapkan adalah yang mendekati sempurna agar

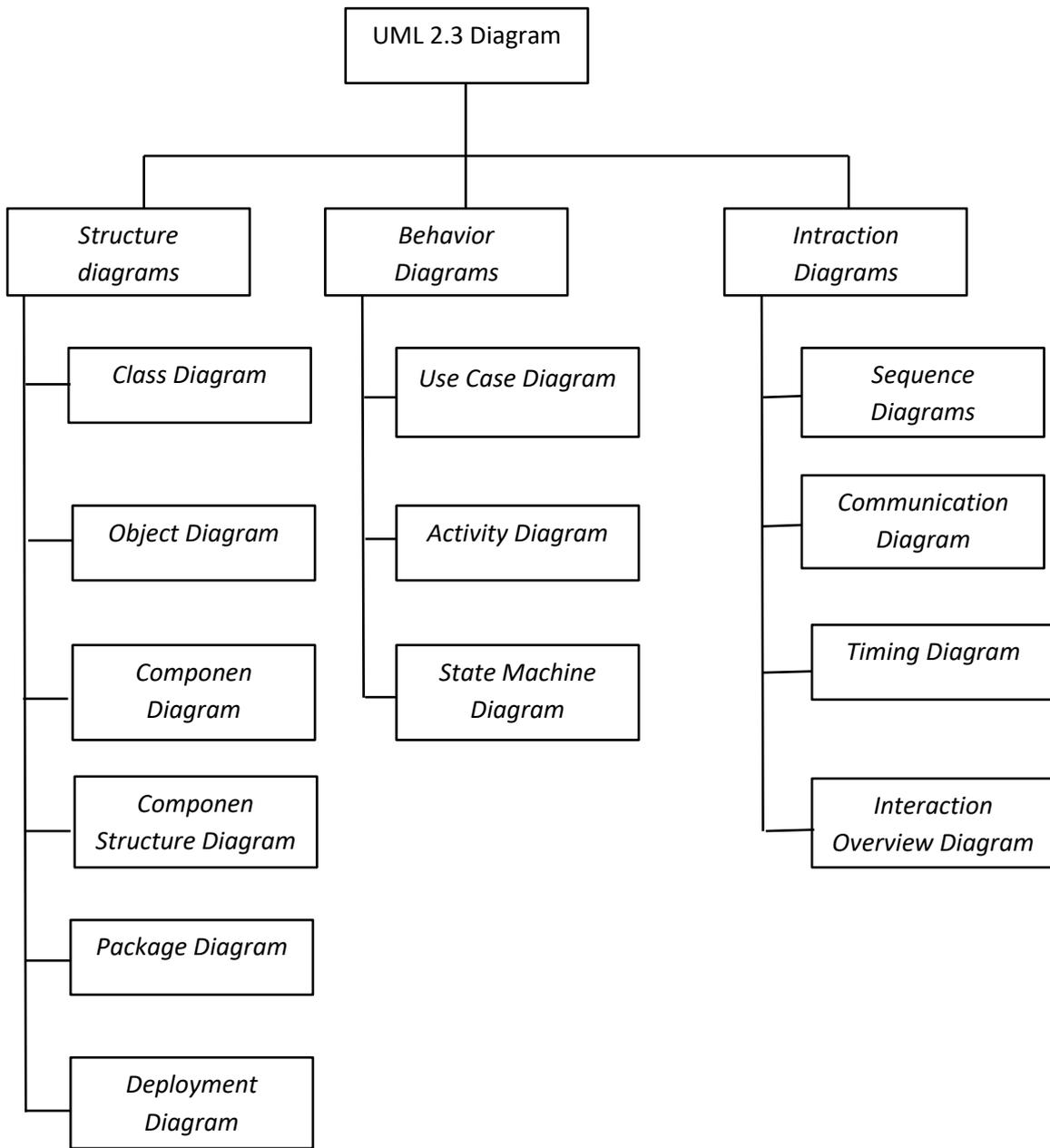
menghasilkan keputusan yang mendekati *valid*. Walaupun sulit untuk mencapai yang sempurna, rasio konsistensi diharapkan kurang dari atau sama dengan 10%.

2.2.4. *Unified Modeling Language (UML)*

UML adalah sebuah “bahasa” yang telah menjadi standar dalam industry untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. UML menawarkan sebuah standart untuk merancang model sebuah sistem.

Dengan menggunakan UML kita dapat membuat model untuk seua jenis aplikasi piranti lunak, dimana aplikasi tersebut dapat berjalan pada piranti keras, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa pemrograman apapun. Tetapi karena uml juga menggunakan *class* dan *operation* dalam konsep dasarnya, maka ia lebih cocok untuk penulisan piranti lunak dalam bahasa-bahasa berorientasi objek seperti C++, Java, C# atau VB.NET (Yasin, 2012:194).

Pada diagram dibawah ini terdiri dari 13 macam diagram yang dikelompokan dalma 3 kategori dan macam-macam diagram tersebut dapat dilihat pada gambar dibawah ini.



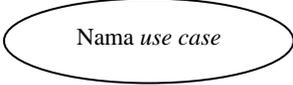
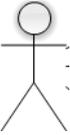
Gambar 2.1 Macam-macam UML

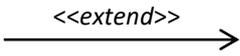
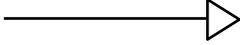
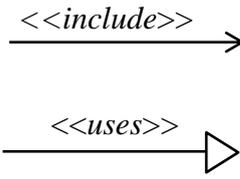
2.2.4.1. Use Case Diagram

Use case merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada didalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu.

Syarat penamaan pada *use case* adalah nama didefinisikan sesimple mungkin dan dapat dipahami. Ada dua hal utama pada *use case* yaitu pendefenisian apa yang disebut aktor dan *use case* (Rosa dan Shalahuddin, 2013:155).

1. Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor itu orang, tapi aktor belum tentu merupakan orang.
2. *Use case* merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukaran pesan antara unit atau aktor.

Simbol	Deskripsi
<i>Use case</i> 	Fungsionalias 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>  Nama aktor	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri.

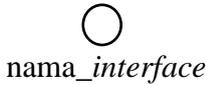
<p>Asosiasi / <i>association</i></p> 	<p>Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor</p>
	<p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu; mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek</p>
<p>Generalisasi / <i>generalization</i></p> 	<p>Hubungan generalisasi dan spesialisasi (umum – khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.</p>
<p>Menggunakan / <i>include</i> / <i>uses</i></p> 	<p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini</p> <p>Ada dua sudut pandang yang cukup besar mengenai <i>include</i> di <i>use case</i>:</p> <ol style="list-style-type: none"> 1. <i>Include</i> berarti <i>use case</i> yang ditambahkan akan selalu dipanggil saat <i>use case</i> tambahan dijalankan 2. <i>Include</i> berarti <i>use case</i> yang tambahan akan selalu melakukan pengecekan apakah <i>use case</i> yang ditambahkan telah dijalankan sebelum <i>use case</i> tambahan dijalankan. <p>Kedua interpretasi diatas dapat dianut salah satu atau keduanya tergantung pada pertimbangan dan interpretasi yang dibutuhkan</p>

Tabel 2.4. Penjelasan *Use Case* Diagram

2.2.4.2. Class Diagram

Diagram kelas 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 dan Shalahuddin, 2013:141).

1. Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas
2. Operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas

Simbol	Deskripsi
<p>Kelas</p> <div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 10px auto;"> <p>Nama_kelas</p> <p>+atribut</p> <p>+operasi()</p> </div>	Kelas pada struktur sistem
<p>Antarmuka / <i>interface</i></p> <div style="text-align: center; margin: 10px auto;">  </div>	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek
<p>Asosiasi / <i>association</i></p> <div style="text-align: center; margin: 10px auto;">  </div>	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
<p>Asosiasi beralih / <i>directed association</i></p> <div style="text-align: center; margin: 10px auto;">  </div>	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
<p>generalisasi</p> <div style="text-align: center; margin: 10px auto;">  </div>	Relasi antar kelas dengan makna generalisasi – spesialisasi (umum – khusus)
<p>Kebergantungan / <i>dependency</i></p> <div style="text-align: center; margin: 10px auto;">  </div>	Relasi antar kelas dengan makna kebergantungan antar kelas
<div style="text-align: center; margin: 10px auto;">  </div>	Relasi antar kelas dengan makna semua-bagian (<i>whole-part</i>)

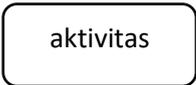
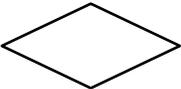
Tabel 2.5. Penjelasan *Class Diagram*

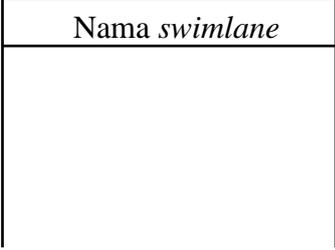
2.2.4.3. Activity Diagram

Diagram aktivitas menggambarkan *workflow* (aliran kerja) atau aktifitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem (Rosa dan Shalahuddin, 2013:161).

Diagram aktivitas juga banyak digunakan untuk mendefinisikan hal-hal berikut :

1. Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan
2. Urutan atau pengelompokan tampilan dari sistem / *user interface* dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan
3. Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya
4. Rancangan menu yang ditampilkan pada perangkat lunak.

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

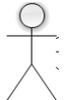
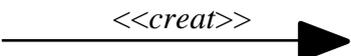
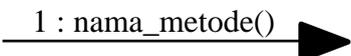
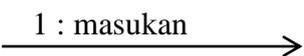
Penggabungan / <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
<i>Swimlane</i> 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

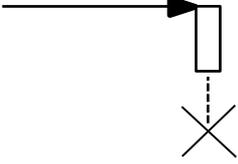
Tabel 2.6. Penjelasan *Activitie* Diagram

2.2.4.4. *Sequence* Diagram

Diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek.

Banyaknya diagram sekuen yang harus digambar adalah minimal sebanyak pendefinisian *use case* yang memiliki proses sendiri atau yang penting semua *use case* yang telah didefinisikan interaksinya pesan sudah dicakup pada diagram sekuen sehingga semakin banyak *use case* yang didefinisikan maka diagram sekuen yang harus dibuat juga semakin banyak (Rosa dan Shalahuddin, 2013:165).

Simbol	Deskripsi
<p data-bbox="288 338 383 371">Aktor</p>  <p data-bbox="427 488 584 521">Nama aktor</p>	<p data-bbox="834 338 1402 483">Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri</p>
<p data-bbox="288 562 568 595">Garis hidup / <i>lifeline</i></p> 	<p data-bbox="834 562 1310 595">Menyatakan kehidupan suatu objek</p>
<p data-bbox="288 853 384 887">Objek</p> <div data-bbox="339 925 711 1025" style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;"> <p data-bbox="379 943 671 976"><u>nama objek : nama kelas</u></p> </div>	<p data-bbox="834 853 1402 931">Menyatakan objek yang berinteraksi pesan</p>
<p data-bbox="288 1077 456 1111">Waktu aktif</p> 	<p data-bbox="834 1077 1402 1155">Menyatakan objek dalam keadaan aktif dan berinteraksi pesan</p>
<p data-bbox="288 1312 504 1346">pesan tipe <i>creat</i></p> 	<p data-bbox="834 1312 1402 1424">Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat</p>
<p data-bbox="288 1458 488 1491">pesan tipe <i>call</i></p> 	<p data-bbox="834 1458 1402 1570">Menyatakan suatu objek operasi memanggil / metode yang ada pada objek lain atau dirinya sendiri.</p>
<p data-bbox="288 1603 504 1637">Pesan tipe <i>send</i></p> 	<p data-bbox="834 1603 1402 1749">Menyatakan bahwa suatu objek mengirim data / masukan / informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim</p>
<p data-bbox="288 1749 520 1783">Pesan tipe <i>return</i></p> 	<p data-bbox="834 1749 1402 1939">Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian</p>

<p>Pesan tipe <i>destroy</i></p> 	<p>Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada <i>creat</i> maka ada <i>destroy</i></p>
--	---

Tabel 2.7. Penjelasan *Sequence Diagram*

2.2.5. Eclipse

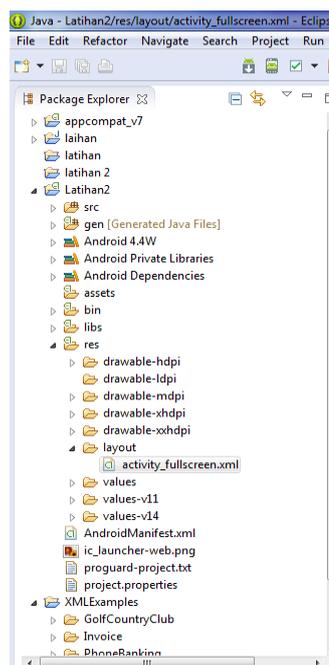
Merupakan sebuah IDE yang gratis dan *open source* atau yang dapat dikembangkan dan digunakan untuk membangun sebuah program komputer dan dapat dijalankan disemua *platform*. Eclipse memiliki banyak keunggulan, diantaranya (Satyaputra dan Alfa, 2012:12) :

1. Dapat dijalankan pada sistem informasi apa saja (*multi platform*).
2. Mendukung pengembangan aplikasi dengan bahasa pemrograman lain, seperti C/C++, COBOL, Python, Perl, PHP, dan lain sebagainya.
3. Dapat digunakan dalam siklus pengembangan perangkat lunak, seperti dokumentasi, tes perangkat lunak, pengembangan web, dan lainnya.
4. Relatif lebih ringan dan memiliki kemampuan untuk dapat dikembangkan oleh pengguna dengan banyak komponen. Salah satunya adalah JDT yang membuat eclipse kompatibel dalam pengembangan program. Eclipse dilengkapi juga dengan adanya PDE (*Plug-in Development Environment*) untuk mengembangkan plug-in baru.

Selain keunggulan tersebut, eclipse juga unggul dalam konsepnya yang mengusung IDE yang open source dan mudah diperluas untuk apa saja. Bahkan eclipse tidak saja untuk mengembangkan program java, namun juga keperluan lainnya. Dan itu semua dapat dilakukan dengan cukup hanya menginstal plugin yang dibutuhkan (Satyaputra dan Alfa, 2012:12).

Berikut adalah beberapa fitur yang terdapat pada aplikasi eclipse (Satyasaputra dan Alfa, 2012:44) :

2.2.5.1. *Package explorer*



Gambar 2.2. *Package explorer*

Package explorer view bertugas menampilkan hierarki elemen-elemen yang menyusun proyek atau program yang anda buat, *package explorer* akan membantu anda dalam melakukan pekerjaan dengan menyediakan perintah-perintah sebagai berikut :

- a. *Back* : berfungsi menavigasi lalu menampilkan paket program paling baru yang anda lihat dalam *package explorer*.
- b. *Forward* : berfungsi menavigasi dan menampilkan paket program lain yang letaknya setelah paket program yang sedang dibuka.
- c. *Up* : berfungsi menavigasi induk dari paket yang sedang ditampilkan pada level atas tampilan.
- d. *Collapse all* : menutup seluruh pohon paket program.
- e. *Link with editor* : menghubungkan *package explorer* ke editor aktif.
- f. *Top level elements project* : menampilkan projek yang sedang aktif sebagai elemen tingkat atas pada *package explorer*.
- g. *Top level elemen working sets*: menampilkan *working sets* sebagai elemen tingkat atas pada *package explorer*.
- h. *Filters* : membuka dialog penyaringan elemen pada java.

2.2.5.2. Console

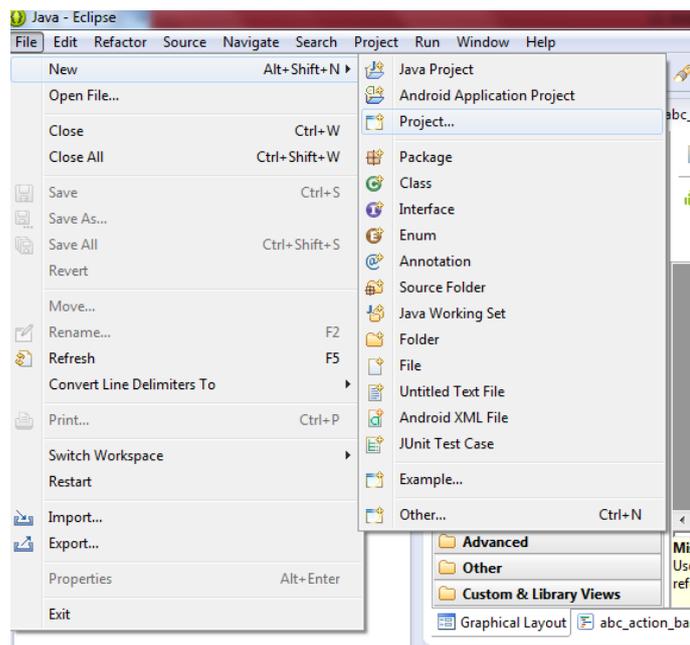


Gambar 2.3. Console

Ketika menggunakan Eclipse, terdapat tiga *console* yang *default* dengan eclipse yaitu : *process console*, *stacktrace console*, dan *cvs console*. Berikut adalah perintah yang terdapat pada *console* tersebut :

- a. *Clear console* : berfungsi untuk menghapus *console* yang sedang aktif
- b. *Display slected console* : membantu anda membuka daftar *console*, dan memungkinkan anda untuk melihat *console* yang anda inginkan.
- c. *Open console* : membuka *console* baru.
- d. *Pins* : menandai *console* untuk berada di atas *console* lainnya.
- e. *Scroll lock* : digunakan untuk mengaktifkan maupun menonaktifkan *scroll* pada sebuah *console*.

2.2.5.3. Membuat Projek Baru



Gambar 2.4. Projek Baru

Membuat proyek baru dilakukan untuk mengarahkan lalu menyatukan dan menyimpan seluruh elemen program atau proyek yang akan anda buat satu kesatuan didalam sebuah folder pada *workspace* yang telah di tentukan.

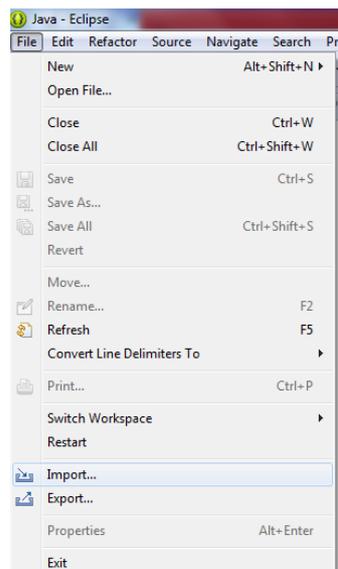
Membuat proyek baru dapat dilakukan dengan cara memilih *New > project* (gambar 2.3.) ketika anda telah membuka eclipse lalu memilih *workspace* sesuai dengan keinginan.

2.2.5.4. *Import* Proyek

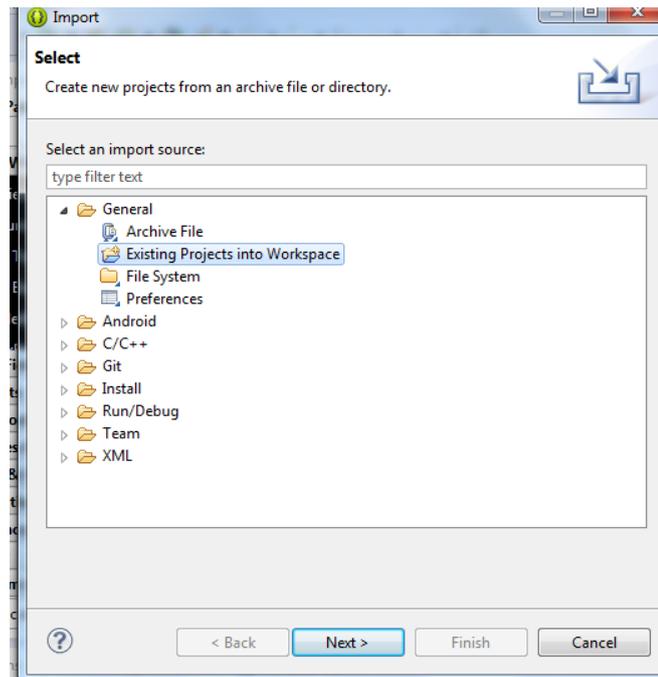
Java dibuat dengan motto “*write once, run anywhere*” yang artinya java dapat berdiri sendiri dan dapat beradaptasi dengan lingkungan *Operating system* (OS). Dengan demikian java mempermudah anda dalam pembuatan program, proses *running*, maupun pengaplikasian program.

Import proyek adalah salah satu fasilitas yang memwadahi hal tersebut, anda dapat membaca dan menjalankan proyek yang dibuat dengan *Operating System* (OS) yang berbeda dengan yang digunakan.

Berikut adalah langkah-langkah yang digunakan dalam melakukan *import project*:



Gambar 2.5. *Import Project*



Gambar 2.6. *import project tahap 2*

- a. *Copy folder project yang akan diimport, lalu paste pada project eclipse pada komputer anda.*
- b. *Buka eclipse anda lalu klik File > Import seperti gambar 2.4 di atas*
- c. *Ketika anda memilih import maka akan muncul pilihan import, drop down pilihan general seperti gambar 2.5. diatas Untuk memilih project*

2.3. Penelitian Terdahulu

Pada tinjauan penelitian terdahulu akan dibahas secara lengkap jurnal dan artikel yang mendukung sebagai dasar pembahasan interpretasi penelitian pada bahan sebelumnya.

1. **Irman Hariman, S.T., M.T dan Komar Rusmana (2014)** dalam jurnal LPKIA, Vol 1 No 1, September 2014, STMIK LPKIA Bandung yang berjudul “*APLIKASI PENDUKUNG KEPUTUSAN DALAM PENENTUAN OBJEK WISATA ALAM MENGGUNAKAN METODE AHP BERBASIS ANDROID*”. Dalam penelitian ini ditemukan bahwa aplikasi pendukung keputusan dapat memberikan solusi mengenai pemilihan objek wisata alam yang sesuai dengan kriteria yang di inginkan.
2. **Iwan Rijayana dan Lirien Okirindho (2013)** dalam jurnal Pelita Informatika, Vol IV No 2 ISSN 2301-9425, Agustus 2013, STMIK Budi Darma Medan yang berjudul “*PENERAPAN METODE ANALITYCAL HIERARCHY PROCESS PADA SISTEM PENDUKUNG KEPUTUSAN PEMILIHAN LEAPTOP*”. Dalam penelitian ini ditemukan bahwa aplikasi pendukung keputusan yang dilakukan pada pemilihan leaptop dengan menggunakan metode Ahp dalam menentukan kriteria dan bobot untuk di hitung secara.
3. **Iwan Rijayana dan Lirien Okirindho (2012)** dalam jurnal PROSIDING SEMNASIF 2012, ISSN 1979-2328, Fakultas Tenknik Universitas Widyatama Bandung yang berjudul “*SISTEM PENDUKUNG KEPUTUSAN PEMILIHAN KARYAWAN BERPRESTASI BERDASARKAN KINERJA MENGGUNAKAN METODE ANALITYC HIERARCY PROCESS* ”. Dalam penelitian ini ditemukan bahwa sistem pendukung keputusan dapat digunakan sebagai alat untuk mengevaluasi atas kinerja karyawan dengan menggunakan metode Ahp