

BAB II

LANDASAN TEORI

II.1. Sistem

Menurut Churchman (2011) sistem adalah seperangkat bagian-bagian yang dikoordinasikan untuk melaksanakan seperangkat tujuan. Sehingga dapat disimpulkan bahwa sistem merupakan rangkaian bagian-bagian yang saling berhubung dan saling berpengaruh satu sama lain yang dikoordinasikan untuk melaksanakan seperangkat tujuan tertentu. (Rochmawati Daud ; 2014 : 18).

II.2. Informasi

Informasi merupakan data yang telah diproses sehingga mempunyai arti tertentu bagi penerimanya. Sumber dari informasi adalah data, sedangkan data itu merupakan suatu peristiwa yang terjadi pada waktu tertentu. Dalam hal ini informasi dan data saling berkaitan. Dengan adanya sistem yang baik diharapkan dapat menghasilkan suatu informasi yang berkualitas tinggi. (Rochmawati Daud; 2014 : 18).

II.3. Pengertian Sistem Informasi

Suatu sistem dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat

menyediakan kepada pihak luar tertentu dengan informasi yang diperlukan untuk pengambilan keputusan. (Rochmawati Daud; 2014 : 18).

II.4. Sistem Pendukung Keputusan (SPK)

Menurut Keen dan Scoot Morton Sistem Pendukung Keputusan merupakan penggabungan sumber - sumber kecerdasan individu dengan kemampuan komponen untuk memperbaiki kualitas keputusan. Sistem Pendukung Keputusan juga merupakan sistem informasi berbasis komputer untuk manajemen pengambilan keputusan yang menangani masalah - masalah semi struktur. (Nanda Abdurrahman Wahid ; 2014 : 93).

Sistem pendukung keputusan atau *Decision Support System (DSS)* adalah sistem informasi interaktif yang menyediakan informasi, pemodelan, dan pemanipulasian data yang digunakan untuk membantu pengambilan keputusan pada situasi yang semi terstruktur dan situasi yang tidak terstruktur dimana tak seorang pun tahu secara pasti bagaimana keputusan seharusnya dibuat. Konsep DSS dikemukakan pertama kali oleh Scoot Morton pada tahun 1971. (Nanda Abdurrahman Wahid ; 2014 : 93).

II.5. Analytical Hierarchy Process (AHP)

Analytical Hierarchy Process (AHP) merupakan suatu model pendukung keputusan yang dikembangkan oleh Thomas L. Saaty. Model pendukung keputusan ini akan menguraikan masalah multi faktor atau multi kriteria yang kompleks menjadi suatu hirarki, menurut Saaty, hirarki didefinisikan sebagai

suatu representasi dari sebuah permasalahan yang kompleks dalam suatu struktur multi level dimana level pertama adalah tujuan, yang diikuti level faktor, kriteria, sub kriteria, dan seterusnya ke bawah hingga level terakhir dari alternatif. Dengan hirarki, suatu masalah yang kompleks dapat diuraikan ke dalam kelompok-kelompoknya yang kemudian diatur menjadi suatu bentuk hirarki sehingga permasalahan akan tampak lebih terstruktur dan sistematis. (Sylvia Hartati Saragih ; 2013 : 83).

Analytical Hierarchy Process merupakan salah satu metode untuk membantu menyusun suatu prioritas dari berbagai pilihan dengan menggunakan berbagai kriteria. (Dita Monita ; 2013 : 31).

II.5.1. Tahapan *Analytical Hierarchy Process* (AHP)

Menurut (Kadarsyah Suryadi dan Ali Ramdhani, 1998) Dalam metode *Analytical Hierarchy Process* dilakukan langkah - langkah sebagai berikut : (Dita Monita ; 2013 : 31).

- a. Mendefinisikan masalah dan menentukan solusi yang diinginkan.

Dalam tahap ini kita berusaha menentukan masalah yang akan kita pecahkan secara jelas, detail dan mudah dipahami. Dari masalah yang ada kita coba tentukan solusi yang mungkin cocok bagi masalah tersebut. Solusi dari masalah mungkin berjumlah lebih dari satu. Solusi tersebut nantinya kita kembangkan lebih lanjut dalam tahap berikutnya.

- b. Membuat struktur hierarki yang diawali dengan tujuan utama.

Setelah menyusun tujuan utama sebagai level teratas akan disusun level

hirarki yang berada di bawahnya yaitu kriteria-kriteria yang cocok untuk mempertimbangkan atau menilai alternatif yang kita berikan dan menentukan alternatif tersebut. Tiap kriteria mempunyai intensitas yang berbeda-beda. Hirarki dilanjutkan dengan subkriteria (jika mungkin diperlukan).

- c. Membuat matrik perbandingan berpasangan yang menggambarkan kontribusi relatif atau pengaruh setiap elemen terhadap tujuan atau kriteria yang setingkat di atasnya. Matriks yang digunakan bersifat sederhana, memiliki kedudukan kuat untuk kerangka konsistensi, mendapatkan informasi lain yang mungkin dibutuhkan dengan semua perbandingan yang mungkin dan mampu menganalisis kepekaan prioritas secara keseluruhan untuk perubahan pertimbangan. Pendekatan dengan matriks mencerminkan aspek ganda dalam prioritas yaitu mendominasi dan didominasi. Perbandingan dilakukan berdasarkan judgment dari pengambil keputusan dengan menilai tingkat kepentingan suatu elemen dibandingkan elemen lainnya. Untuk memulai proses perbandingan berpasangan dipilih sebuah kriteria dari level paling atas hirarki misalnya K dan kemudian dari level di bawahnya diambil elemen yang akan dibandingkan misalnya E1,E2,E3,E4,E5.
- d. Melakukan Mendefinisikan perbandingan berpasangan sehingga diperoleh jumlah penilaian seluruhnya sebanyak $n \times [(n-1)/2]$ buah, dengan n adalah banyaknya elemen yang dibandingkan. Hasil perbandingan dari masing-masing elemen akan berupa angka dari 1 sampai 9 yang menunjukkan perbandingan tingkat kepentingan suatu elemen. Apabila suatu elemen

dalam matriks dibandingkan dengan dirinya sendiri maka hasil perbandingan diberi nilai 1. Skala 9 telah terbukti dapat diterima dan bisa membedakan intensitas antar elemen. Hasil perbandingan tersebut diisikan pada sel yang bersesuaian dengan elemen yang dibandingkan. Skala perbandingan perbandingan berpasangan dan maknanya yang diperkenalkan oleh Saaty bisa dilihat di bawah.

Intensitas Kepentingan

1 = Kedua elemen sama pentingnya, Dua elemen mempunyai pengaruh yang sama besar

3 = Elemen yang satu sedikit lebih penting daripada elemen yang lainnya, pengalaman dan penilaian sedikit menyokong satu elemen dibandingkan elemen yang lainnya

5 = Elemen yang satu lebih penting daripada yang lainnya, Pengalaman dan penilaian sangat kuat menyokong satu elemen dibandingkan elemen yang lainnya

7 = Satu elemen jelas lebih mutlak penting dari pada elemen lainnya, Satu elemen yang kuat disokong dan dominan terlihat dalam praktek.

9 = Satu elemen mutlak penting daripada elemen lainnya, Bukti yang mendukung elemen yang satu terhadap elemen lain memiliki tingkat penegasan tertinggi yang mungkin menguatkan.

2,4,6,8 = Nilai-nilai antara dua nilai pertimbangan-pertimbangan yang berdekatan, Nilai ini diberikan bila ada dua kompromi di antara 2 pilihan

Kebalikan = Jika untuk aktivitas i mendapat satu angka dibanding dengan aktivitas j , maka j mempunyai nilai kebalikannya dibanding dengan i .

- e. Menghitung nilai eigen dan menguji konsistensinya. Jika tidak konsisten maka pengambilan data di ulang.
- f. Mengulangi langkah 3,4, dan 5 untuk seluruh tingkat hirarki.
- g. Menghitung *vektor eigen* dari setiap matriks perbandingan berpasangan.

Yang merupakan bobot setiap elemen untuk penentuan prioritas elemen elemen pada tingkat hirarki terendah sampai mencapai tujuan. Penghitungan dilakukan lewat cara menjumlahkan nilai setiap kolom dari matriks, membagi setiap nilai dari kolom dengan total kolom yang bersangkutan untuk memperoleh normalisasi matriks, dan menjumlahkan nilai-nilai dari setiap baris dan membaginya dengan jumlah elemen untuk mendapatkan rata-rata.

- h. Memeriksa konsistensi hirarki.

Yang diukur dalam *Analytical Hierarchy Process* adalah rasio konsistensi dengan melihat index 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 %.

Indeks konsistensi dari matriks berordo n dapat diperoleh dengan rumus:
(Sylvia Hartati Saragih ; 2013 : 83).

$$CI = \frac{\lambda \text{ maksimum} - n}{n - 1}$$

dimana :

CI = Indek konsistensi (*Consistency Index*)

λ maksimum = Nilai eigen terbesar dari matrik berordo n

λ maksimum didapat dengan menjumlahkan hasil perkalian jumlah kolom dengan eigen vektor utama.

Apabila C.I = 0, berarti matriks konsisten.

Batas ketidakkonsistenan yang ditetapkan Saaty diukur dengan menggunakan rasio konsistensi (CR), yakni perbandingan indek konsistensi dengan nilai pembangkit random (RI). Nilai RI bergantung pada ordo matrik n . Nilai RI ditunjukkan pada tabel II.1 dibawah ini. (Sylvia Hartati Saragih ; 2013 : 83).

Tabel II.1. Nilai RI

| N | RI |
|----|------|
| 1 | 0.00 |
| 2 | 0.00 |
| 3 | 0.58 |
| 4 | 0.90 |
| 5 | 1.12 |
| 6 | 1.24 |
| 7 | 1.32 |
| 8 | 1.41 |
| 9 | 1.45 |
| 10 | 1.49 |
| 11 | 1.51 |
| 12 | 1.58 |

Sumber : (Sylvia Hartati Saragih ; 2013 : 83)

CR dirumuskan :

$$CR = \frac{CI}{RI}.$$

II.5.2. Langkah Dan Prosedur *Analytical hierarchy Process*

Untuk memecahkan suatu masalah dengan menggunakan metode *Analytical Hierarchy Process* diperlukan langkah-langkah sebagai berikut :
(Dita Monita ; 2013 : 32).

1. Mendefinisikan permasalahan dan menentukan tujuan.
2. Menyusun masalah ke dalam suatu struktur hierarki sehingga permasalahan yang kompleks dapat ditinjau dari sisi yang detail dan terstruktur.
3. Menyusun prioritas untuk tiap elemen masalah.
4. Melakukan pengujian konsistensi terhadap perbandingan antar elemen yang didapatkan pada tiap tingkat hierarki.

Dalam suatu kelompok yang besar, proses penetapan prioritas lebih mudah ditangani dengan membagi para anggota menjadi subkelompok yang lebih kecil dan terspesialisasi, yang masing-masing menangani suatu masalah dengan bidang tertentu dimana anggotanya mempunyai keahlian khusus. Apabila subkelompok ini digabungkan, maka nilai setiap matrik harus diperdebatkan dan diperbaiki. Akan tetapi perdebatan dapat dihindarkan dan pendapat perseorangan diambil melalui kuisioner dengan membuat nilai akhir dengan menggunakan rata-rata geometric seperti dibawah ini : (Dita Monita ; 2013 : 32).

$$a_w = \sqrt{a_1 \times a_2 \times \dots \times a_n}$$

Keterangan :

a_i = penilaian responden ke - i

a_w = penilaian gabungan

n = banyaknya responden

II.6. Microsoft Visual Studio 2010

Visual Basic diturunkan dari bahasa BASIC. *Visual Basic* terkenal sebagai bahasa pemrograman yang mudah untuk digunakan terutama untuk membuat aplikasi yang berjalan di atas *platform* Windows. Pada tahun 90an, *Visual Basic* menjadi bahasa pemrograman yang paling populer dan menjadi pilihan utama untuk mengembangkan program berbasis windows . Versi *Visual Basic* terakhir sebelum berjalan diatas *.NET Framework* adalah VB6 (*Visual Studio* 1998). (Erick Kurniawan ; 2011 : 1).

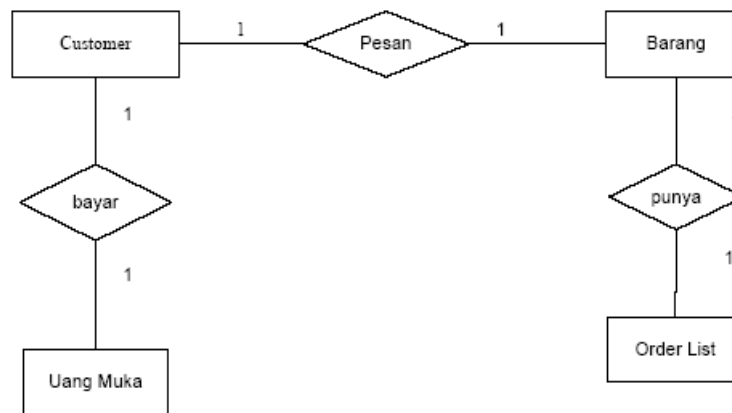
Visual Basic .NET dirilis pada bulan february tahun 2002 bersamaan dengan *platform .NET Framework* 1.0. Kini sudah ada beberapa versi dari *Visual Basic* yang berjalan pada *platform .NET* , yaitu VB 2002 (VB7), VB 2005 (VB8), VB 2008 (VB9), dan yang terakhir adalah VB 2010 (VB10) yang dirilis bersamaan dengan *Visual Studio* 2010. Selain *Visual Basic* 2010, *Visual Studio* 2010 juga mendukung beberapa bahasa lain, yaitu C#, C++, F# (bahasa baru untuk *functional programming*), *IronPhyton*, dan *IronRuby* (bahasa baru untuk *dynamic programming*). (Erick Kurniawan ; 2011 : 1).

II.7. Microsoft SQL Server 2008

SQL Server 2008 adalah sebuah terobosan baru dari *Microsoft* dalam bidang *database*. *SQL Server* adalah sebuah DBMS (*Database Management System*) yang dibuat oleh *Microsoft* untuk ikut berkecimpung dalam persaingan dunia pengolahan data menyusul pendahuluannya seperti *IBM* dan *Oracle*. *SQL Server 2008* dibuat pada saat kemajuan dalam bidang *hardware* sedemikian pesat. Oleh karena itu sudah dapat dipastikan bahwa *SQL Server 2008* membawa beberapa terobosan dalam bidang pengolahan dan penyimpanan data. (Wahana Komputer ; 2010 : 2).

II.8. Entity Relationship Diagram (ERD)

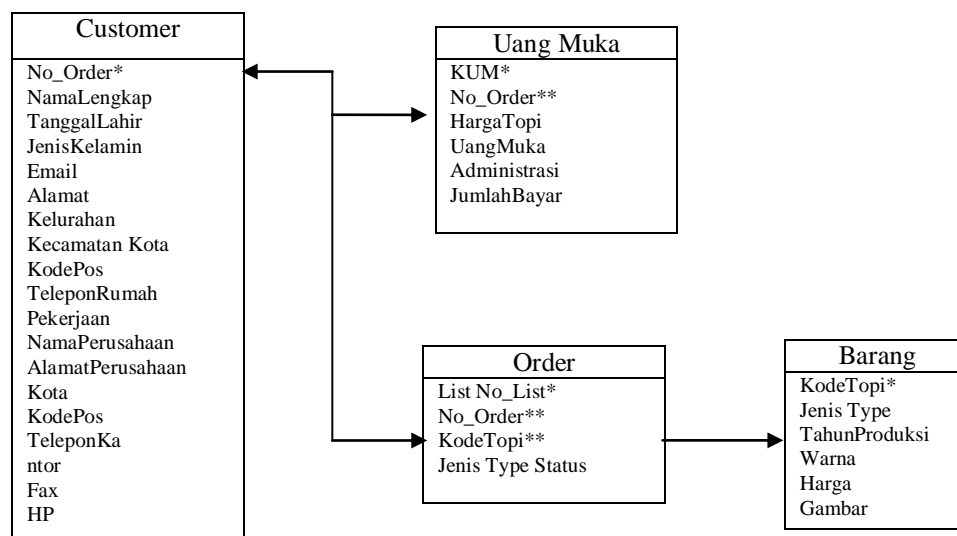
Entity Relationship Diagram atau diagram hubungan entitas dari sistem penjualan yang diusulkan berfungsi untuk menggambarkan model basis data yang akan dipakai. Model basis data yang digunakan adalah basis data relasional, dimana setiap entitas saling memiliki hubungan dengan entitas lain. (Iyan Gustiana ; 2010 : 8).



Gambar II.1. Contoh *Entity Relationship Diagram*
(Sumber : Iyan Gustiana ; 2010 : 9)

II.9. Normalisasi

Normalisasi dilakukan agar basis data yang akan diterapkan dapat digunakan dan dioperasikan dengan efisien, mudah dan tidak mengalami anomali atau keanehan. Normalisasi dimulai dengan menganalisa tabel dalam bentuk tidak normal. (Iyan Gustiana ; 2010 : 9).



Gambar II.2. Contoh Normalisasi
(Sumber : Iyan Gustiana ; 2010 : 9)

II.10. *Unified Modelling Language (UML)*

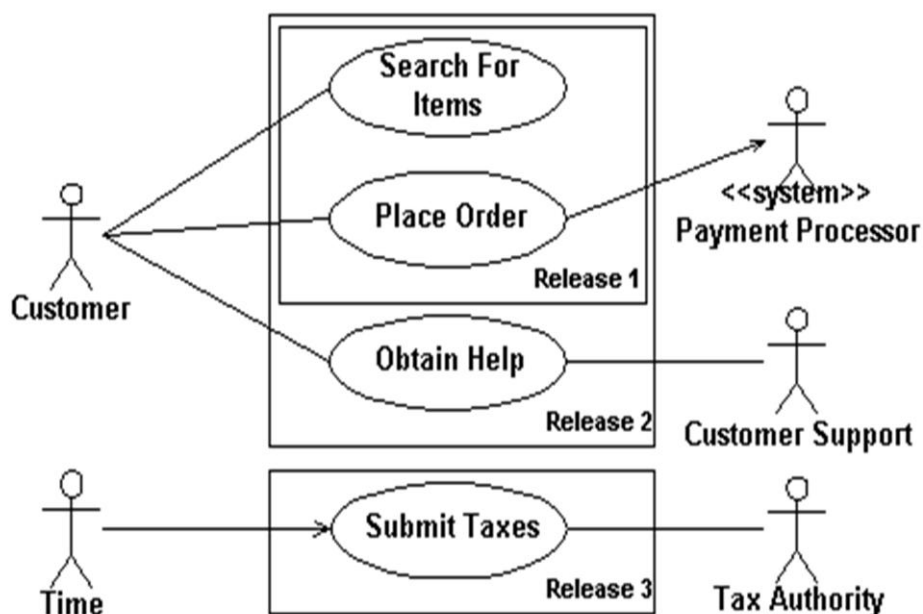
Unified Modelling Language adalah bahasa standar yang digunakan untuk menjelaskan dan memvisualisasikan artifak dari proses analisis dan desain berorientasi obyek. UML menyediakan standar pada notasi dan diagram yang bisa digunakan untuk memodelkan suatu sistem. UML dikembangkan oleh 3 pendekar “berorientasi obyek”, yaitu Grady Booch, Jim Rumbaugh, dan Ivar Jacobson. UML menjadi bahasa yang bisa digunakan untuk berkomunikasi dalam perspektif obyek antara user dengan, antara developer dengan develop, antara developer analisis dengan developer desain, dan antara developer desain dengan developer pemrograman. (Julius Hermawan ; 2010 : 7).

Unified Modelling Language (UML) adalah suatu alat untuk memvisualisasikan dan mendokumentasikan hasil analisa dan desain yang berisi sintak dalam memodelkan sistem secara *visual*. Juga merupakan satu kumpulan

konvensi pemodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem *software* yang terkait dengan objek. (Jurnal Informatika Mulawarman ; Haviluddin ; 2011 : 1).

II.10.1. Use Case Diagram

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*. (Jurnal Informatika Mulawarman ; Haviluddin ; 2011 : 4).

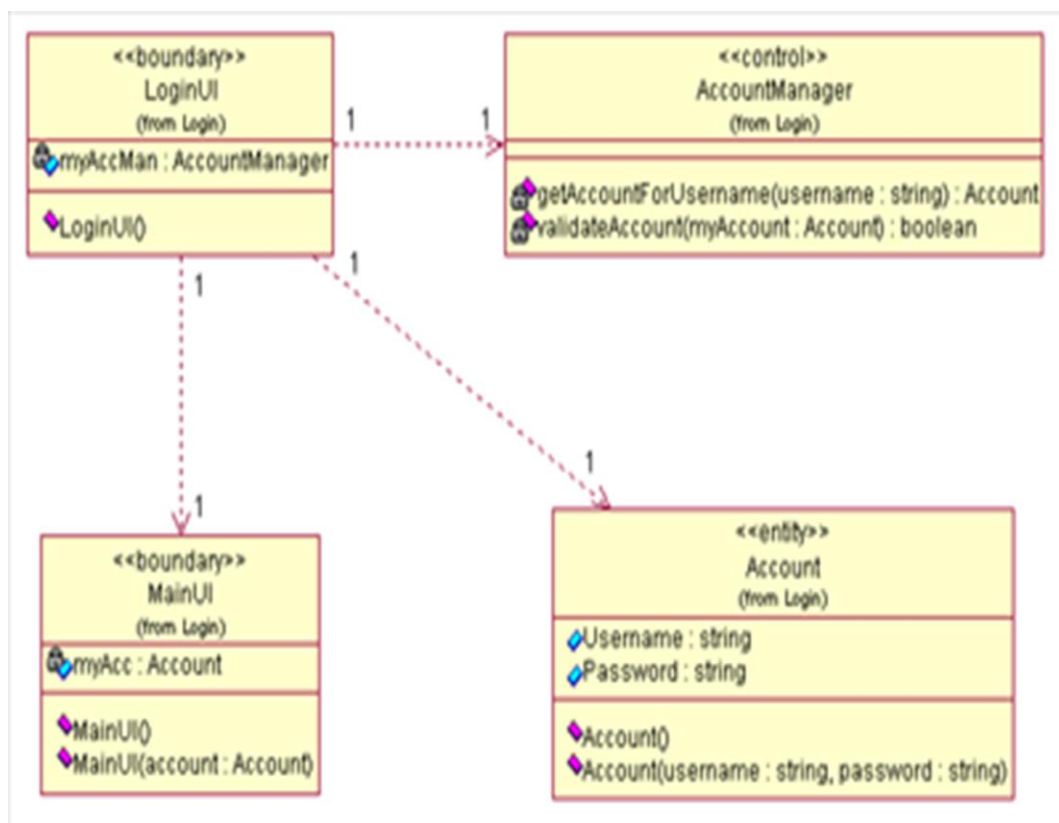


Gambar II.3. Contoh Use Case Diagram
(Sumber : Haviluddin ; 2011 : 4)

II.10.2. Class Diagram

Class diagram menggambarkan struktur statis dari kelas dalam sistem anda dan menggambarkan atribut, operasi dan hubungan antara kelas. *Class*

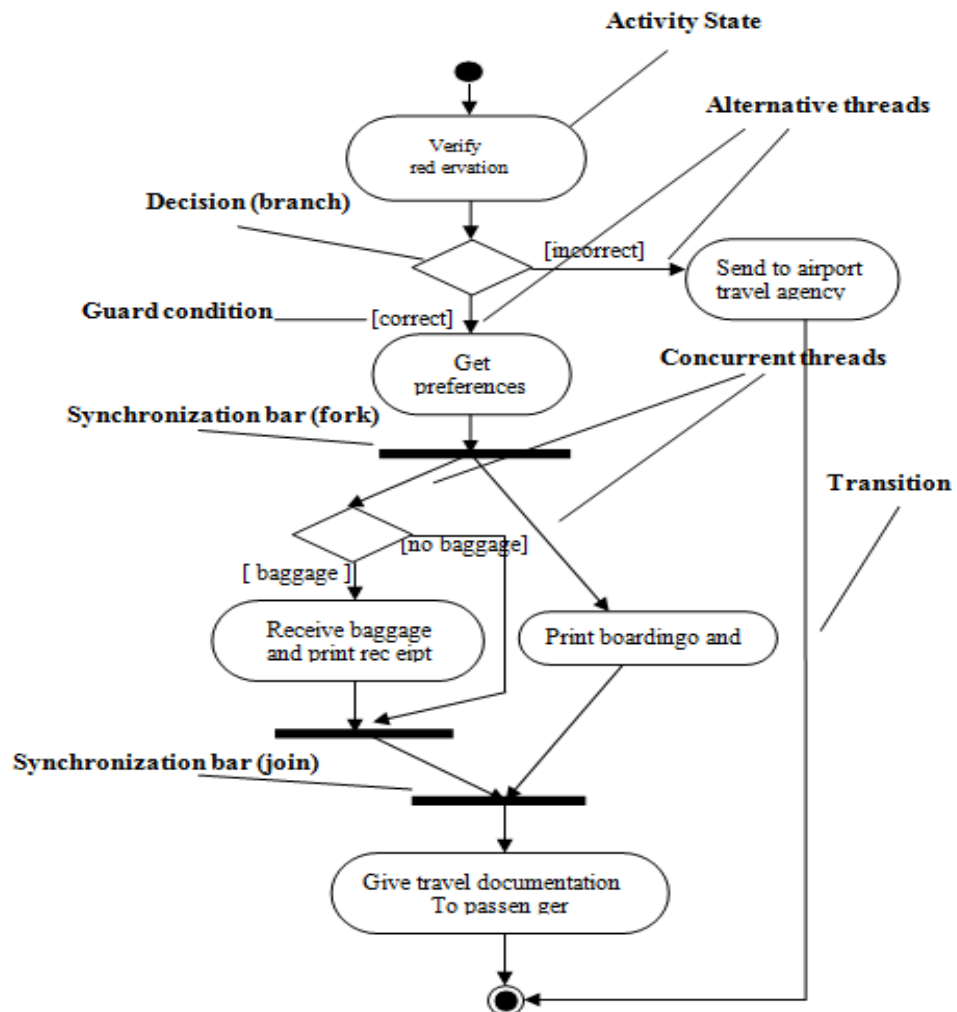
diagram membantu dalam memvisualisasikan struktur kelas-kelas dari suatu sistem dan merupakan tipe diagram yang paling banyak dipakai. Selama tahap desain, *class diagram* berperan dalam menangkap struktur dari semua kelas yang membentuk arsitektur sistem yang dibuat. (Jurnal Informatika Mulawarman ; Haviluddin ; 2011 : 3).



Gambar II.4. Contoh Class Diagram
(Sumber : Haviluddin ; 2011 : 3)

II.10.3. Activity Diagram

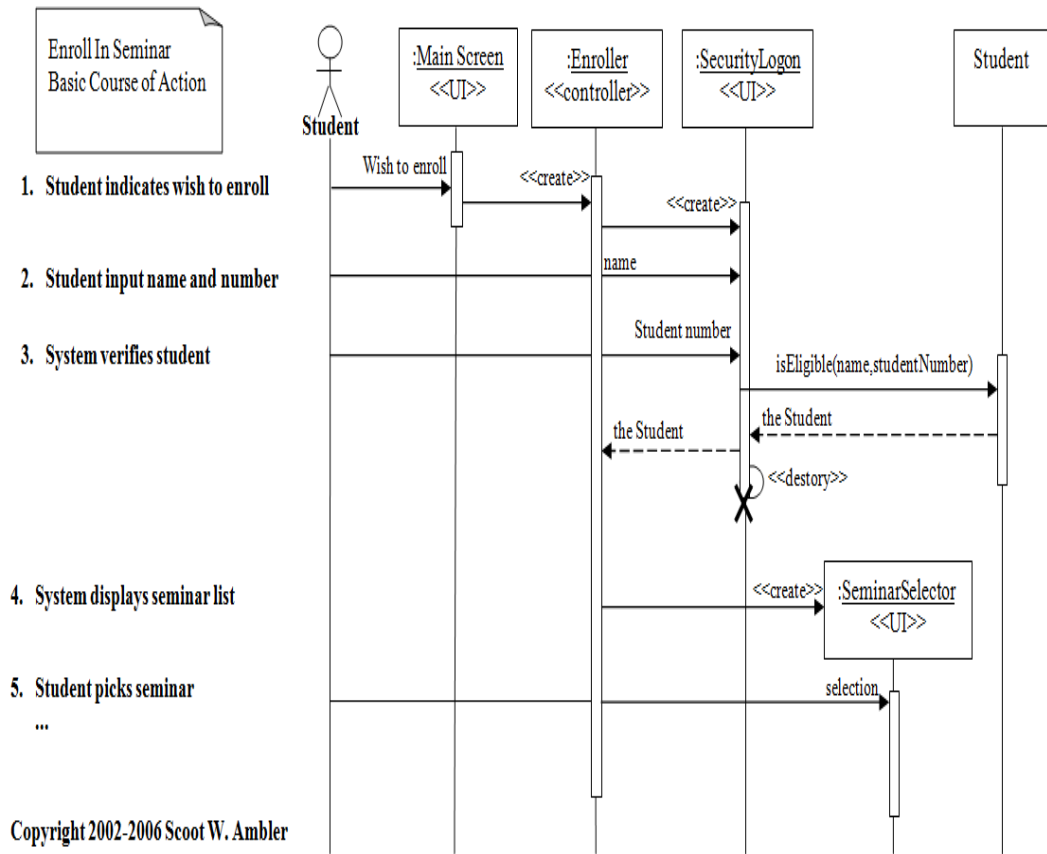
Menggambarkan aktifitas-aktifitas, objek, *state*, transisi *state* dan *event*. Dengan kata lain kegiatan diagram alur kerja menggambarkan perilaku sistem untuk aktifitas. (Haviludin ; 2011 : 4).



Gambar II.5. Contoh Activity Diagram
(Sumber : Havaluddin ; 2011 : 4)

II.10.4. Sequence Diagram

Sequence diagram menjelaskan interaksi objek yang disusun berdasarkan urutan waktu. Secara mudahnya *sequence diagram* adalah gambaran tahap demi tahap, termasuk kronologi (urutan) perubahan secara logis yang seharusnya dilakukan untuk menghasilkan sesuatu sesuai dengan *use case diagram*. (Havaluddin ; 2011 : 5).



Gambar II.6. Contoh Sequence Diagram
(Sumber : Havaluddin ; 2011 : 5)