

## BAB II

### LANDASAN TEORI

#### 2.1 Persamaan Differensial

##### 2.1.1 Pengertian Persamaan Differensial

Persamaan differensial adalah persamaan matematika untuk suatu fungsi tak diketahui dari satu atau beberapa peubah yang menghubungkan nilai dari fungsi tersebut dengan turunannya sendiri pada berbagai derajat turunan (Ledder, 2005, p16). Persamaan differensial muncul dalam berbagai bidang sains dan teknologi: apabila suatu relasi deterministik melibatkan beberapa besaran yang berubah secara kontinu (dimodelkan dengan fungsi) dan laju perubahan besaran itu dalam ruang atau dalam waktu (dimodelkan dengan turunannya) diketahui atau diandaikan. Dalam mekanika klasik, persamaan differensial dipakai dalam penggambaran gerak tubuh dalam kaitannya dengan posisi dan kecepatannya berdasarkan perubahan waktu.

Suatu persamaan differensial disebut persamaan differensial biasa, jika semua turunannya berkaitan dengan satu peubah saja, dan disebut persamaan differensial parsial, jika turunannya berkaitan dengan dua atau lebih peubah. Orde dari persamaan differensial adalah derajat tertinggi dari turunan dalam persamaan yang bersangkutan. Himpunan dari  $n$  persamaan differensial orde-satu dengan  $n$  menyatakan banyaknya persamaan yang tidak diketahui disebut sistem persamaan differensial orde-satu;  $n$  adalah dimensi dari sistem yang bersangkutan. Satu

pengertian lain yang perlu diketahui adalah persamaan differensial otonom. Suatu persamaan differensial biasa atau suatu sistem persamaan differensial biasa disebut otonom jika peubah bebasnya tidak tampak secara eksplisit dalam persamaannya (Ledder, 2005, p16).

Secara matematis, persamaan differensial dipelajari dari beberapa sudut pandang yang berbeda, sebagian besar dari sudut pandang yang beragam itu berminat dengan hasil dari persamaan differensial yang dipelajari, yaitu serangkaian fungsi yang memenuhi persamaan differensial yang diberikan. Hanya persamaan differensial yang paling sederhana memungkinkan penyelesaian berdasarkan rumus eksplisit; akan tetapi, beberapa sifat penyelesaian dari suatu persamaan differensial yang diberikan dapat ditentukan tanpa menemukan bentuknya yang tepat atau eksak. Jika suatu rumus yang dapat ditentukan penyelesaiannya tidak tersedia, hampiran terhadap penyelesaiannya dapat ditentukan secara numerik dengan bantuan komputer.

### **2.1.2 Metode Newton-Raphson**

Salah satu metode penghitungan secara numerik yang akan dipakai dalam program aplikasi yang dirancang yaitu metode Newton-Raphson. Metode Newton-Raphson (umumnya disebut dengan metode Newton), yang mendapat nama dari Isaac Newton dan Joseph Raphson, merupakan metode penyelesaian persamaan non-linear yang sering digunakan di antara metode lainnya, karena metode ini memberikan konvergensi yang lebih cepat dibandingkan dengan metode lainnya. Metode ini merupakan metode yang paling dikenal untuk mencari hampiran

terhadap akar fungsi riil. Metode Newton ini dapat dijabarkan dengan persamaan sebagai berikut :

$$x_{i+1} = x_i - \frac{f_i}{f'_i}, \text{ dimana :}$$

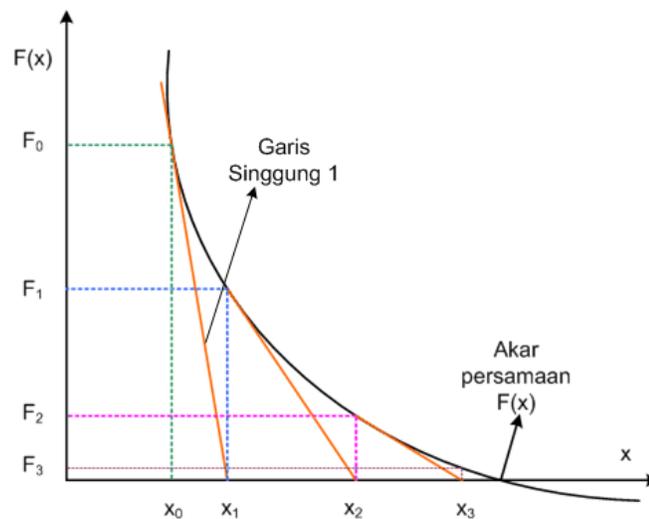
$x_{i+1}$  = nilai x pada iterasi ke  $i + 1$

$x_i$  = nilai x pada iterasi ke  $i$

$f_i$  = nilai fungsi  $F(x)$

$f'_i$  = nilai fungsi turunan pertama dari  $F(x)$

Cara kerja metode newton ini adalah dengan menggunakan garis singgung untuk menentukan nilai x selanjutnya atau  $x_{i+1}$  , garis singgung ini tentunya menyinggung grafik persamaan fungsi  $F(x)$  yang ada, seperti yang terlihat pada gambar dibawah ini :



**Gambar 2.1 Cara Kerja Metode Newton-Raphson**

Misalkan kita memiliki sebuah fungsi  $F(x)$ , dengan akar persamaan seperti yang ditunjukkan pada gambar diatas. Karena metode ini merupakan metode Terbuka, maka tetap diperlukan nilai tebakan awal untuk  $X_0$ . Jika nilai awal tebakan adalah  $x_0$ , maka nilai fungsinya adalah  $F_0$ , sebuah garis singgung ( disebut juga garis tangen atau gradien) dibuat pada titik  $(x_0, F_0)$  yang diteruskan memotong sumbu  $x$  pada  $x_1$ , dengan menggunakan nilai  $x_1$  ini dihitung nilai fungsi  $F(x)$  yaitu pada  $F_1$ , garis singgung berikutnya dibuat pada titik  $(x_1, F_1)$  memotong sumbu  $x$  pada  $x_2$ , begitu seterusnya ( secara berurutan , sequence ) hingga mendekati akar persamaan yang diinginkan. Melalui persamaan garis singgung ( tangen atau gradien ) kita dapat menurunkan metode newton ini :

Lihat garis singgung 1, garis tersebut dapat dibuat dari dua titik yaitu  $(x_0, F_0)$  dan  $(x_1, 0)$ , gradien garis tersebut adalah :

$$F'_0 = \frac{\Delta y}{\Delta x} = \frac{0 - F_0}{x_1 - x_0} = \frac{-F_0}{x_1 - x_0}$$

Persamaan ini disusun kembali menjadi :

$$x_1 = x_0 - \frac{F_0}{F'_0}$$

Atau dapat juga ditulis menjadi :

$$x_{i+1} = x_i - \frac{f_i}{f'_i}$$

dengan  $f'_i \neq 0$

Perhitungan iterasi ini akan dihentikan apabila

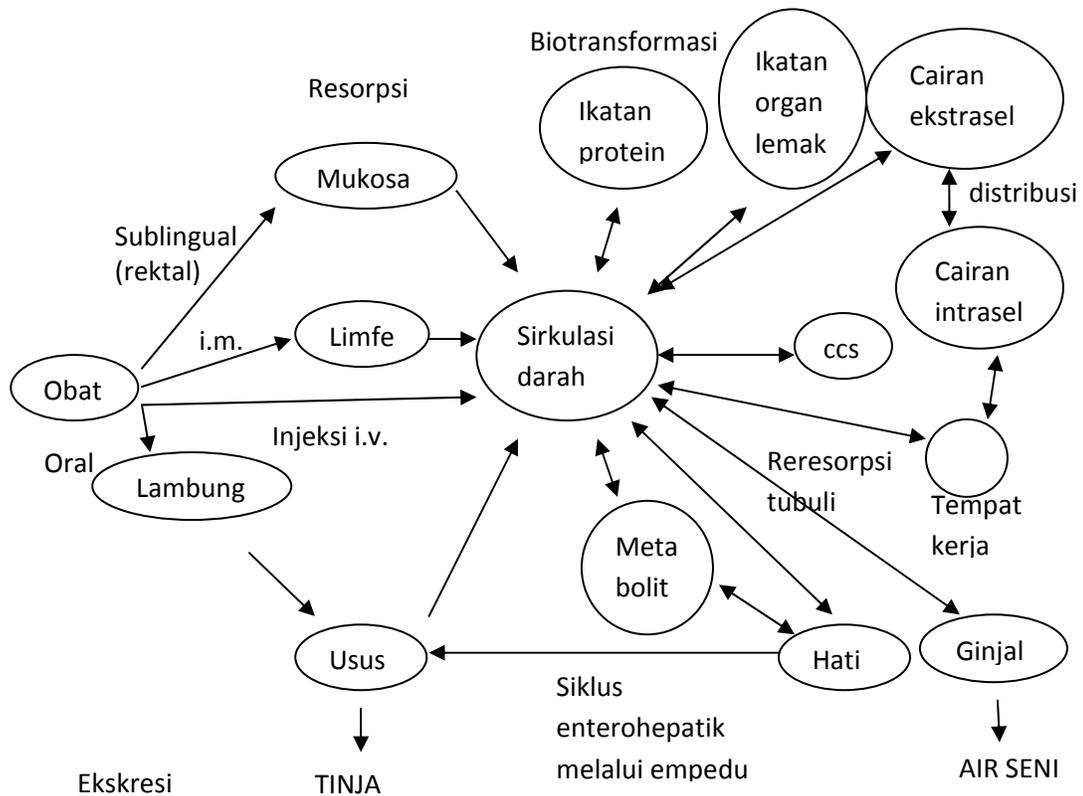
$$\left| \frac{x_{i+1} - x_i}{x_{i+1}} \right| = \left| \frac{x_{\text{daru}} - x_{\text{lama}}}{x_{\text{daru}}} \right| < \delta \text{ atau } |x_{i+1} - x_i| < \varepsilon$$

## 2.2 Farmakokinetika

### 2.2.1 Pengertian Dasar Farmakokinetika

Farmakokinetika adalah ilmu yang khusus mempelajari perubahan – perubahan konsentrasi dari obat dan metabolitnya di dalam darah dan jaringan sebagai fungsi dari waktu sebagai hasil dari proses yang dilakukan tubuh terhadap obat, yaitu resorpsi, transpor, biotransformasi (metabolisme), distribusi dan ekskresi (Hoan Tjay, 2008, p22). Farmakokinetika menggunakan model matematika untuk menguraikan proses – proses resorpsi, distribusi, biotransformasi, dan ekskresi, dan memperkirakan besarnya kadar obat dalam plasma sebagai fungsi dari besarnya dosis, interval pemberian dan waktu.

Pada skema dibawah ini digambarkan proses – proses farmakokinetik yang dapat dialami obat selama perjalanannya di dalam tubuh:



**Gambar 2.2 Perjalanan obat di dalam tubuh dengan proses – proses farmakokinetika yang dialaminya**

### 2.2.2 Sistem Transpor

Setelah obat masuk dalam tubuh, kita harus mempertimbangkan bagaimana obat akan diedarkan keseluruh tubuh. Misalnya apakah partikel obat akan berikatan dengan serum protein ataukah beredar bebas. Sistem transportasi meliputi pergerakan obat dari peredaran darah ke jaringan, organ dan bagian lain dari tubuh dimana obat akan berpengaruh.

Untuk mentransfer obat ke tempat yang tepat di dalam tubuh, zat aktif diolah menjadi suatu bentuk khusus. Molekul zat kimia obat dapat melintasi membran semipermeabel berdasarkan adanya perbedaan konsentrasi. Pada proses ini beberapa mekanisme transpor memegang peranan, yaitu secara pasif (dengan cara filtrasi dan atau difusi) atau secara aktif (tidak tergantung konsentrasi obat).

### **2.2.3 Resorpsi**

Umumnya penyerapan obat dari usus ke dalam sirkulasi berlangsung melalui filtrasi, difusi atau transpor aktif. Zat hidrofil yang melarut dalam cairan ekstra-sel diserap dengan mudah, sedangkan zat – zat yang sukar melarut lebih lambat diresorpsi.

Kecepatan resorpsi terutama tergantung pada bentuk pemberian obat, cara pemberiannya dan sifat fisiko-kimiawinya. Dengan cara apa obat akan diberikan kepada pasien, apakah dengan diminum langsung, disuntik, intramuscularly, dihirup, ataukah lewat kulit akan mempengaruhi dalam menentukan proses absorpsi. Selain itu juga harus dipertimbangkan apakah obat yang diberikan akan mengalami proses perubahan fisik atau kimiawi ketika dimasukkan dalam tubuh. Sebagai contoh, apakah obat berbentuk padat sehingga harus dihancurkan terlebih dahulu ataukah berbentuk cairan.

Cara pemberian berdasarkan efek yang diinginkan, yaitu efek sistemis (di seluruh tubuh) atau efek lokal (setempat). Untuk mendapatkan efek

sistemik dapat diberikan dengan cara, yaitu oral, sublingual (obat ditempatkan dibawah lidah), injeksi, implantasi subkutan (memasukkan obat ke bawah kulit), dan rektal. Sedangkan untuk mendapatkan efek lokal dapat diberikan dengan cara, yaitu intrasal, intra-okuler dan intra-aurikuler, inhalasi, intravaginal, dan kulit (berupa salep, krem, atau lotion).

Resorpsi dari usus ke dalam sirkulasi berlangsung cepat bila obat diberikan dalam bentuk terlarut (obat cairan, sirop atau obat tetes). Sedangkan obat padat (tablet, kapsul atau serbuk) akan lebih lambat diresorpsi, hal ini dikarenakan obat padat harus dipecah dahulu dan zat aktifnya perlu dilarutkan dalam cairan lambung-usus. Dalam proses tersebut, kecepatan larut partikel obat berperan penting; semakin halus semakin cepat larutnya dan resorpsi obat.

Oleh karena itu, pemberian secara injeksi intravena menghasilkan efek yang tercepat, karena obat langsung masuk ke dalam sirkulasi. Efek lebih lambat lagi diperoleh dengan injeksi intramuskuler (i.m.) karena obat harus melewati banyak membran sel sebelum tiba dalam peredaran darah besar.

Untuk obat yang diberikan secara oral akan diresorpsi dari saluran lambung-usus. Kebanyakan obat bersifat asam atau basa organik lemah mengalami disosiasi menjadi ion dalam larutan. Besarnya ionisasi untuk setiap zat berbeda dan tergantung dari konstanta disosiasinya dan derajat asam lingkungan sekitarnya. Lebih sedikit obat terdisosiasi, lebih lancar pula penyerapannya. Untuk obat yang bersifat asam lemah, hanya akan sedikit

terurai menjadi ion dalam lingkungan asam kuat di dalam lambung, sehingga resorpsinya sangat baik di lambung. Sebaliknya, basa lemah terionisasi dengan baik pada pH lambung sehingga hanya sedikit diresorpsi. Sedangkan pada usus halus zat yang bersifat basa lemah paling mudah diserap. Pada usus besar terkandung sangat sedikit air untuk melarutkan obat yang belum terlarut dalam usus halus. Selain itu juga tidak terdapat jonjot mukosa dan transpor aktif yang menyebabkan obat yang diserap secara aktif sebaiknya tidak diberikan secara rektal dan suppositoria sebaiknya digunakan ketika rektum dalam keadaan kosong.

#### **2.2.4 Biotransformasi**

Pada dasarnya setiap obat merupakan zat asing yang tidak diinginkan bagi tubuh, karena obat dapat merusak sel dan mengganggu fungsinya. Oleh karena itu, tubuh akan berusaha untuk merombak zat asing ini menjadi metabolit yang tidak aktif lagi dan sekaligus bersifat lebih hidrofil agar memudahkan proses ekskresinya oleh ginjal. Dengan demikian reaksi - reaksi metabolisme dalam hati dan beberapa organ lain (paru – paru, ginjal, dinding usus dan juga di dalam darah) disebut biotransformasi. Persentase obat yang secara utuh mencapai sirkulasi umum untuk melakukan kerjanya disebut bio-availability.

Metabolisme berlangsung melalui kerja yang dilakukan katalis protein yang disebut enzim. Metabolisme obat seringkali berlangsung pada liver,

walaupun ada juga beberapa enzim pada darah. Enzim liver yang terlibat dalam metabolisme obat termasuk dalam keluarga epoxide hidratase dan cytochrome P450 enzim liver mengkatalisasi modifikasi kimiawi pada obat, lalu mengoksidasinya dan membuatnya lebih mudah dikeluarkan.

Kecepatan biotransformasi umumnya bertambah bila konsentrasi obat meningkat. Hal ini akan terus berlaku hingga seluruh molekul enzim yang melakukan perubahan ditempati terus menerus oleh molekul obat sehingga kecepatan biotransformasi menjadi konstan. Faktor lain yang mempengaruhi adalah fungsi hati, usia, faktor genetis dan juga interaksi dengan penggunaan bersama obat lain.

### **2.2.5 Distribusi**

Obat yang telah melalui hati bersamaan dengan metabolitnya disebarkan secara merata ke seluruh jaringan tubuh, khususnya melalui peredaran darah. Seringkali distribusi obat tidak merata akibat beberapa gangguan, yaitu adanya rintangan, terikatnya obat pada protein darah atau jaringan dan lemak. Bagian obat yang mengalami pengikatan protein darah akan hilang aktivitas farmakologinya dan menjadi inaktif, tetapi tidak mengalami proses biotransformasi dan ekskresi. Pengikatan protein ini dapat dianggap suatu cara untuk menyimpan obat, karena bagian yang terikat tidak dirombak atau diekskresi. Pada umumnya, ketika konsentrasi obat bebas

menurun, ikatan obat-protein akan pecah dan obat terlepas kembali, hingga kadar obat bebas hampir tidak berubah.

### **2.2.6 Ekskresi**

Ekskresi adalah proses mengeluarkan obat atau metabolitnya dari tubuh, terutama dilakukan oleh ginjal melalui air seni. Ginjal akan menyaring darah dan membuang obat dan sisa metabolisme. Mekanisme ekskresi obat yang dilakukan oleh ginjal yakni filtrasi glomeruli (pasif) dan transpor aktif untuk zat – zat tertentu.

Walaupun sebagian obat tidak masuk ke peredaran darah untuk bisa memberikan efek pada penyakit yang diderita, tapi sebagian sisanya akan tetap dikeluarkan secara bertahap dari peredaran darah oleh proses yang dinamakan elimination. Proses eliminasi sebagian besar disebabkan oleh ginjal, tetapi metabolisme juga mempengaruhi.

Obat bisa juga dikeluarkan dari liver di bile. Bile adalah suatu substansi yang dibutuhkan dalam pengolahan lemak, yang dihasilkan oleh liver, disimpan di gallblader, dan dikirimkan ke intestin kecil. Cara lain, yaitu melalui kulit bersama keringat, paru – paru melalui pernafasan, empedu oleh hati, air susu ibu pada ibu menyusui dan juga usus. Contoh ekskresi melalui pernafasan adalah pada tes kadar alkohol pada pengemudi kendaraan bermotor. Pada usus, zat – zat yang tidak atau tak lengkap diresorpsi dikeluarkan melalui tinja.

### 2.2.7 Konsentrasi Plasma

Untuk dapat menilai suatu obat secara klinis, menetapkan dosis, dan skema penakarannya yang tepat, perlu adanya sejumlah data farmakokinetika. Khususnya mengenai kadar obat di tempat tujuan dan dalam darah, serta perubahan kadar ini dalam waktu tertentu. Pada umumnya besarnya efek obat tergantung pada konsentrasinya di target site itu dan ini berhubungan erat dengan konsentrasi plasma. Pada obat yang resorpsinya baik, kadar plasma meningkat bila dosisnya diperbesar. Kadar obat dalam plasma, yang nilainya kurang lebih sama dengan konsentrasinya dalam darah, dapat diukur dengan alat – alat modern dengan ketelitian dari satu per seribu mg (0,001 mg).

Dengan mengambil contoh darah dari seorang pasien yang telah diberikan suatu dosis obat tertentu pada beberapa titik waktu, kemudian mengukur kadarnya dalam contoh – contoh darah tersebut dan menggambarkan nilai – nilai kadar ini sebagai fungsi dari waktu, maka dapat diperoleh grafik konsentrasi-waktu dari obat tersebut.

Gambar 2.3 memperlihatkan grafik lengkung yang lazim pada kebanyakan obat. Di sini dapat dilihat bahwa obat mencapai konsentrasi puncak dalam waktu 2 jam, lalu menurun. Penurunan ini mula – mula agak cepat dan kemudian berkurangnya konsentrasi akan berlangsung secara berangsur – angsur yaitu secara eksponensiil. Hal ini disebabkan eliminasi obat setiap menit menjadi semakin sedikit.

### **Gambar 2.3 Grafik eksponensial**

#### **2.2.8 Plasma Half-life**

Telah kita lihat, bahwa turunnya kadar-plasma obat dan lama efeknya bergantung pada kecepatan metabolisme dan ekskresi. Kedua faktor ini menentukan kecepatan eliminasi obat, yang dinyatakan dengan pengertian masa-paruh ( $t_{1/2}$ ), yaitu rentang waktu di mana kadar obat dalam plasma pada fase eliminasi menurun sampai separuhnya. Pada gambar 2-2, masa paruh obat adalah 4 jam, kadar menurun dari 8 sampai 4 dalam waktu 4 jam, begitu pula dari 4 sampai 2 dan seterusnya.

Setiap obat memiliki masa paruh yang berlainan dan dapat bervariasi dari 23 detik (adrenalin) hingga 2 tahun (obat kontras-iod organik). Waktu paruh juga berbeda secara perorangan berhubung variasi individual. Waktu paruh juga dipengaruhi oleh faktor – faktor lain, yaitu fungsi organ – organ eliminasi. Faktor tersebut sangatlah penting, karena pada kerusakan hati atau ginjal, maka waktu paruh dapat meningkat sampai 20 kali atau lebih.

### **2.3 Efek Terapeutis Obat**

Tidak semua obat bersifat betul – betul menyembuhkan penyakit; banyak diantaranya yang hanya meniadakan atau meringankan gejalanya. Oleh karena itu, dapat dibedakan tiga jenis pengobatan, yaitu:

- a. Terapi kausal, menghilangkan penyebab penyakit, dengan memusnahkan kuman, virus, atau parasit. Contoh: antibiotik, obat – obat malaria.
- b. Terapi simptomatis, hanya mengobati gejala penyakit yang timbul dan meringankan penyebabnya sedangkan yang lebih mendalam tidak dipengaruhi, misalnya kerusakan pada suatu organ atau saraf. Contoh: analgetika pada rematik atau sakit kepala, obat hipertensi dan obat jantung.
- c. Terapi substitusi, menggantikan zat yang lazimnya dibuat oleh organ yang sakit. Misalnya insulin pada diabetes, karena produksinya oleh pankreas kurang atau berhenti, tiroksin pada hipotirostis dan estrogen pada hipofungi ovarium di masa klimakterium wanita.

Efek terapeutis obat tergantung dari banyak faktor, antara lain dari cara dan bentuk pemberian, sifat fisiko-kimiawinya yang menentukan resorpsi, biotransformasi dan ekskresinya dari tubuh. Selain itu juga dari kondisi fisiologis penderita (fungsi hati, ginjal, usus, dan peredaran darah). Faktor – faktor individual lainnya, misalnya etnis, kelamin, luas permukaan badan dan kebiasaan makan juga dapat memegang peranan penting.

Efek terapeutis akan berlangsung ketika konsentrasi obat dalam tubuh berada di antara minimum effect concentration (MEC) dan minimum toxic concentration (MTC). MEC adalah konsentrasi minimal obat dalam tubuh agar obat bisa bekerja dan menunjukkan hasil yang diharapkan. Sedangkan MTC adalah konsentrasi minimal obat dalam tubuh sehingga obat memberikan efek racun bukannya menyembuhkan. Oleh karena itulah diperlukan perhitungan untuk mengukur konsentrasi obat dalam tubuh terhadap waktu secara dinamis agar pengobatan yang dilakukan bisa memberikan hasil yang efektif dan aman. Salah satu contoh pentingnya pemberian obat agar konsentrasinya lebih besar dari MEC adalah dalam kemoterapeutika. Jika obat yang diberikan berada dibawah batas MEC, maka besar kemungkinan kuman akan berkembang lagi dan bahkan bisa timbul sifat resistensi kuman terhadap obat yang diberikan.

### **2.3.1 Kesetiaan Terapi (*patient compliance*)**

Banyak penelitian yang menunjukkan bahwa sejumlah besar pasien tidak minum obatnya dengan taat dan teratur, atau tidak menghabiskan obat yang

diberikan padanya sesuai dengan resep dokter. Hal ini menyebabkan obat tidak memberikan efek optimal yang diinginkan, bahkan dapat menimbulkan resistensi, khususnya pada antibiotika. Kesetiaan dan kerelaan pasien untuk meminum obatnya dipengaruhi oleh sejumlah faktor dan yang utama adalah (Hoan Tjay, 2008, p38):

- a. Sifat individual: misalnya watak, tingkat pendidikan, dan kepekaan untuk nyeri.
- b. Relasi dokter-pasien: bila pasien tidak senang dengan perlakuan dokter atau tidak menerima informasi yang cukup mengenai penyakitnya compliance akan turun. Begitu pula jika dokter tidak memberikan instruksi yang lengkap atau cukup jelas mengenai penggunaan obat. Misalnya, antibiotika harus diselesaikan pengobatannya, walaupun gejala penyakit infeksi sudah lenyap.
- c. Jenis penyakit: semakin berat penyakit, semakin baik compliance-nya, juga bila pasien merasa nyeri. Sebaliknya, compliance berkurang bila obat harus diminum dalam waktu yang lama atau bertahun – tahun, sedangkan penyakit tidak memperlihatkan gejala tidak nyaman (radang, nyeri) seperti diabetes dan hipertensi.
- d. Jumlah obat dan frekuensi takarannya: semakin banyak frekuensi pemberian obat, akan semakin turun compliance. Bila obat harus digunakan lebih dari dua kali sehari, compliance menurun dengan nyata, begitu pula bila obat tidak diberikan sebagai tablet atau kapsul, melainkan cairan atau suppositoria.

### **2.3.2 Dosis**

Dosis obat yang diberikan harus diberikan pada pasien untuk menghasilkan efek yang diharapkan tergantung dari banyak faktor, antara lain usia, bobot badan, jenis kelamin, besarnya luas permukaan tubuh, beratnya penyakit dan keadaan daya-tangkis penderita. Berikut akan dibahas faktor usia, jenis kelamin, dan besarnya luas permukaan tubuh.

#### **2.3.2.1 Umur**

Pada orang yang berusia di atas 65 tahun, lazimnya lebih peka terhadap obat dan efek sampingnya, karena perubahan – perubahan fisiologis, seperti menurunnya fungsi ginjal dan metabolisme hati, meningkatnya rasio lemak-air dan berkurangnya sirkulasi darah. Karena fungsi ginjal dan hati sudah menurun, maka eliminasi obat pun berlangsung lebih lambat. Selain itu, beberapa faktor seperti bakteri serta kekurangan nutrisi makanan banyak terjadi pada manula yang akhirnya menurunkan fungsi kerja hati. Lagi pula jumlah albumin darahnya lebih sedikit, oleh karena itu pengikatan obat pun berkurang, terutama obat – obat dengan persentase ikatan protein besar, seperti anti-koagulansia dan fenilutazon. Hal ini berarti bahwa bentuk bebas dan aktif dari obat – obat ini menjadi lebih besar dan bahaya keracunan semakin meningkat.

Selanjutnya, pada manula tak jarang terjadi kerusakan umum pada sel – sel otak, yang mengakibatkan peningkatan kepekaan bagi obat dengan kerja

pusat, seperti obat tidur. Obat ini pada dosis biasa dapat mengakibatkan reaksi keracunan yang hebat pada manula, juga obat jantung digoksin, hormon insulin, dan adrenalin.

Oleh karena faktor – faktor tersebut, bagi lansia dianjurkan menggunakan dosis yang lebih rendah, yakni (Hoan Tjay, 2008, p45):

- 65 – 74 tahun: dosis biasa – 10%
- 75 – 84 tahun: dosis biasa – 20%
- 85 tahun dan seterusnya: dosis biasa – 30%

Sedangkan untuk anak – anak berusia dibawah 18 tahun, dosis yang dianjurkan berdasarkan rumus Augsberger adalah (Hoan Tjay, 2008, p46):

- 2 – 12 bulan:  $(m + 13)\%$  dari Dosis dewasa
- 1 – 12 tahun:  $(4n + 20)\%$  dari Dosis dewasa
- 12 – 17 tahun:  $(5n + 10)\%$  dari Dosis dewasa

### **2.3.2.2 Jenis Kelamin**

Perbedaan massa otot, aliran darah organ dan banyak cairan tubuh pada pria dan wanita mempengaruhi parameter farmakokinetik dari banyak obat. Beberapa penelitian juga menunjukkan perbedaan kecepatan metabolisme untuk beberapa jenis obat karena perbedaan kelamin. Banyak penelitian yang mengurangi dosis obat untuk wanita sebesar 10 – 20% dari dosis pria.

### 2.3.2.3 Luas Permukaan Tubuh

Luas permukaan tubuh berhubungan langsung dengan kecepatan metabolisme obat. Misalnya, parameter eliminasi, seperti filtrasi glomeruler, volume darah dan arusnya di ginjal. Semakin bertambah usia, maka perbandingan antara permukaan badan dan bobotnya akan menjadi lebih kecil. Obat dengan luas terapi sempit, dosisnya selalu ditentukan berdasarkan luas permukaan tubuh, karena lebih eksak. Formula Dubois untuk menghitung luas permukaan tubuh:

$$BSA = 0.007184 \times \text{Weight (kg)}^{0.425} \times \text{Height (cm)}^{0.725}$$

Formula Crawford – Erry Rourke digunakan untuk menghitung besar dosis yang diberikan:

$$Dosis = \frac{BSA}{adult\ BSA} \times \text{dosis dewasa}$$

BSA dewasa : 1,73 m<sup>2</sup>

## 2.4 Kombinasi Obat

Dua obat yang digunakan pada waktu bersamaan dapat saling memengaruhi khasiatnya masing – masing, yakni dapat memperlihatkan kerja berlawanan (antagonisme) atau kerja sama (sinergisme).

- a. Antagonisme terjadi jika kegiatan obat pertama dikurangi atau ditiadakan sama sekali oleh obat kedua yang memiliki khasiat farmakologi berlawanan, misalnya

barbital dan strychnin, adrenalin dan histamin. Pada antagonisme kompetitif, dua obat bersaing secara reversibel untuk reseptor yang sama, misalnya antihistamin dan histamin. Ada juga obat – obat yang bersaing secara tak reversibel untuk molekul yang sama, misalnya zat – zat chelasi pada keracunan logam.

- b. Sinergisme adalah kerja sama antara dua obat atau dikenal dua jenis:
- i. Adisi. Efek kombinasi adalah sama dengan jumlah kegiatan dari masing – masing obat, misalnya kombinasi asetosal dan parasetamol, juga trisulfa.
  - ii. Potensiasi. Kedua obat saling memperkuat khasiatnya, sehingga terjadi efek yang melebihi jumlah matematis dari  $a + b$ . Kedua obat kombinasi dapat memiliki kegiatan yang sama, seperti estrogen dan progesteron, asetosal dan kodein. Atau, satu obat dari kombinasi memiliki efek yang berlainan, misalkan analgetika dan klorpromazin.

Sering kali kombinasi obat diberikan dalam perbandingan tetap dengan maksud mengadidaya kerja terapeutisnya tanpa mengadisi efek buruknya, seperti pada trisulfa. Atau untuk mencegah timbulnya resistensi kuman, misalnya kombinasi INH dan PAS. Kadang – kadang, ditambahkan obat pembantu untuk meniadakan efek samping obat pertama, seperti kalium pada diuretik thiazid, dan ranitidin pada penggunaan NSAID.

Tersedianya kombinasi tetap dari dua atau lebih obat adalah praktis, karena pasien hanya harus minum satu tablet atau kapsul. Kesulitan yang muncul adalah dosis obat tidak dapat diubah tanpa mengubah pula dosis obat kedua, sedangkan skema pentakaran untuk kedua obat tidak selalu sama berhubung dengan masa paruh obat yang berlainan.

## 2.5 Interaksi Obat

### 2.5.1 Interaksi antar obat

Bila seorang pasien diberikan dua atau lebih obat, kemungkinan besar akan terjadi interaksi antara obat – obatan tersebut di dalam tubuhnya. Efek masing – masing obat dapat saling mengganggu atau efek samping yang tidak diinginkan mungkin akan timbul. Efek – efek yang mungkin timbul antaranya menurunnya efisiensi salah satu obat, meningkatnya efek atau konsentrasi obat dalam tubuh sehingga cenderung menjadi toksik dan meningkatnya kemungkinan timbul efek samping.

Secara rata – rata, interaksi antara 3 jenis obat dapat menimbulkan 3 efek interaksi, sedangkan 4 jenis obat dapat menimbulkan 6 efek interaksi, interaksi 5 jenis obat dapat menimbulkan 10 efek interaksi (resiko tinggi), 6 jenis obat menyebabkan 15 efek interaksi (bahaya tinggi), dan 7 jenis obat bahkan dapat menyebabkan kematian ( slide FARMAKOLOGI & TERAPEUTIK 3 FK UNDIP, tanpa tahun, slide 80).

Ada beberapa cara berlangsungnya interaksi obat, yang terpenting di antaranya adalah:

- a. Interaksi kimiawi. Obat beraksi dengan obat lain secara kimiawi, misalnya penisilamin oleh Cu, Pb, atau Au.
- b. Kompetisi untuk protein plasma: analgetika, klobrifat dan kinidin mendesak obat lain dari ikatannya pada protein dan dengan demikian memperkuat khasiatnya.

- c. Induksi enzim. Obat yang menstimulir pembentukan enzim hati, tidak hanya mempercepat eliminasinya, tetapi juga mempercepat perombakan obat lain.
- d. Inhibisi enzim. Zat yang mengganggu fungsi hati dan enzimnya, seperti alkohol, dapat memperkuat daya kerja obat lain yang efek dan lama kerjanya tergantung pada enzim tersebut.

### **2.5.2 Interaksi Obat Dengan Makanan**

Adakalanya terjadi interaksi dari obat dengan bahan makanan, yang dapat mempengaruhi farmakokinetika obat. Interaksi obat terutama harus diperhatikan bila obat diberikan bersamaan dengan obat lain yang indeks terapinya kecil, sehingga sedikit peningkatan kadar plasmanya sudah dapat menimbulkan gejala toksis.

- a. Absorpsi. Obat dapat diikat oleh makanan, sehingga absorpsinya di usus dapat diperlambat atau dikurangi dan efeknya akan menurun. Misalnya, dengan mengonsumsi makanan yang banyak serat dapat mengabsorpsi obat, seperti lovastatin, sehingga bioavailabilitasnya menurun, sedangkan serat sendiri berdaya menurunkan kolesterol. Efek sama terjadi pada digoksin dan garam litium. Contoh lain adalah interaksi dari antikoagulan dengan sayuran yang mengandung vitamin K, seperti bayam, brokoli dan kol kecil. Bila dimakan terlalu banyak, vitamin K dapat mengurangi efek antikoagulan.

- b. Metabolisme. Perombakan obat dapat dirintangi, sehingga kadarnya meningkat dan timbul efek toksis. Contoh yang terkenal adalah interaksi MAO-blockers dengan keju dan coklat. Enzim MAO bertanggung jawab atas penguraian semua katecholamin di dalam tubuh, misalnya adrenalin, serotonin dan dopamin. Bila pasien diberi perintang MAO sebagai antidpresivum dan makan sesuatu yang mengandung tiramin atau amin lain, maka zat ini tidak akan diuraikan lagi karena enzim MAO sudah diblokir. Sebagai akibatnya dapat terjadi hipertensi hebat dengan efek buruknya. Makanan yang mengandung amin antara lain keju, alpukat, anggur, produk – produk ragi dan hati ayam. Coklat mengandung feniletilamin.
- c. Ekskresi. Suatu diet vegetatis ketat meningkatkan pH urin dan memperlancar ekskresi obat yang bersifat asam lemah, seperti vitamin C dan NSAIDs, juga makanan dengan buah – buahan (kecuali prune kering), semua sayuran (kecuali jagung dan “lentils”), kentang dan susu. Diet yang kaya protein (daging, ikan, kerang, keju, telur), mentega kacang, roti dan cake menurunkan pH urin. Urin asam ini mengurangi reabsorpsi tubuler obat yang bersifat basa lemah dan dengan demikian memperbesar ekskresinya, misalnya alkaloida (morfin).

## 2.6 Parameter Farmakokinetik

### 2.6.1 Bioavailabilitas (=F)

Parameter ini menunjukkan fraksi dari dosis obat yang mencapai peredaran darah sistemik dalam bentuk aktif. Jika obat dalam bentuk aktif diberikan secara injeksi intravena (IV) maka  $F=1$ , karena obat tidak perlu melewati proses resorpsi melainkan langsung masuk dalam sistem peredaran darah. Sedangkan jika obat diberikan per oral maka  $F$  biasanya kurang dari 1 dan besarnya bergantung pada jumlah obat yang dapat menembus dinding saluran cerna (jumlah obat yang diresorpsi) dan jumlah obat yang mengalami eliminasi presistemik (metabolisme lintas pertama) di mukosa usus dan dalam hati.

Besarnya bioavailabilitas suatu obat oral di gambarkan oleh AUC (area under curve atau luas area di bawah kurva kadar obat  $C_p$  dalam plasma terhadap waktu) obat oral tersebut dibandingkan dengan AUC-nya pada pemberian IV. Ini disebut bioavailabilitas oral. AUC dapat di formulasikan sebagai berikut

$$AUC = \int C_p(t) dt$$

$$F = \frac{AUC_{oral}}{AUC_{IV}}$$

### 2.6.2 Volume Distribusi (Vd)

Parameter ini menunjukkan volume penyebaran obat dalam tubuh dengan kadar plasma atau serum.  $V_d$  bukanlah volume tubuh yang sebenarnya,

tetapi hanya volume semu yang menggambarkan luasnya distribusi obat dalam tubuh. Besarnya  $V_d$  ditentukan oleh ukuran dan komposisi tubuh, kemampuan molekul obat memasuki berbagai kompartemen tubuh, dan derajat ikatan obat dengan protein plasma dan dengan berbagai jaringan. Secara sederhana,  $V_d$  adalah volume yang dibutuhkan untuk obat menyebar secara homogen pada darah, plasma dan cairan plasma.

$$V_d = \frac{\text{jumlah obat dalam tubuh}}{\text{kadar obat dalam plasma atau serum}}$$

Obat dengan volume distribusi yang sangat besar memiliki konsentrasi yang lebih tinggi di jaringan extravascular, hingga cenderung terjadi penimbunan obat di jaringan tersebut dibandingkan di kompartemen vascular (plasma). Sedangkan obat yang memiliki ikatan yang kuat dengan protein plasma sehingga memiliki konsentrasi dalam plasma yang cukup tinggi, maka volume distribusinya kecil.

### **2.6.3 Waktu Paruh Eliminasi ( $=t^{1/2}$ )**

Ini adalah waktu yang diperlukan untuk turunnya kadar obat dalam plasma atau serum pada fase eliminasi (setelah fase resorpsi dan distribusi) menjadi separuhnya. Untuk obat – obatan yang mengalami eliminasi presistemik,  $t^{1/2}$  ini merupakan bilangan konstan yang tidak tergantung dari besarnya dosis, interval pemberian, maupun cara pemberian. Akan tetapi, parameter ini adalah bentuk turunan dan tergantung pada bersihan dan volume distribusi dari obat.

Oleh karena itu, jika bersihan dan volume distribusi berubah karena penyakit, interaksi obat dan umur, maka perubahan waktu paruh juga bisa terjadi. Waktu paruh biasanya dihitung dari persamaan berikut:

$$t^{1/2} = \frac{0.693}{k} = \frac{\ln(2)}{k}$$

dimana, k = ratio eliminasi konstan

karena  $k = \text{bersihan (Cl)} / V_d$ , maka hubungan antar parameter terbukti dengan jelas.

Secara medis, k dan Cl adalah dua parameter farmakokinetik yang penting dalam menentukan penjadwalan dosis pasien yang spesifik. Waktu paruh adalah parameter farmakokinetik yang penting dalam menentukan interval antar dosis, sedangkan besarnya dosis ditentukan dari dua parameter farmakokinetik lainnya, yaitu volume distribusi dan bersihan.

#### **2.6.4 Bersihan (Clearance = Cl)**

Cl adalah volume darah/ plasma yang dibersihkan dari obat per satuan waktu (mL/ menit).

$$Cl = \frac{k}{C}$$

dengan Cl = bersihan, k = ratio eliminasi, dan C = konsentrasi obat pada kondisi stabil.

Eliminasi obat dari tubuh melibatkan proses yang terjadi pada ginjal, hati, paru – paru dan organ lain. Dengan membagi ratio eliminasi dari setiap organ dengan konsentrasi obat, kita akan dapatkan bersihan (Cl) pada setiap organ tersebut. Jika nilai tersebut digabungkan menjadi satu, akan menjadi bersihan total seluruh tubuh.

$$Cl_{renal} = \frac{k_{ginjal}}{C}$$

$$Cl_{hati} = \frac{k_{hati}}{C}$$

$$Cl_{paru-paru} = \frac{k_{paru-paru}}{C}$$

$$Cl_{lainnya} = \frac{k_{lainnya}}{C}$$

$$Cl_{total} = Cl_{ginjal} + Cl_{hati} + Cl_{paru-paru} + Cl_{lainnya}$$

## 2.7 Model Farmakokinetik Tubuh Manusia

Model digunakan untuk mendeskripsikan dan menginterpretasikan suatu data yang diperoleh dari hasil percobaan. Model farmakokinetik adalah struktur hipotetikal yang bisa digunakan untuk menjelaskan proses yang dialami dan nasib obat dalam sistem biologis tubuh ketika diberikan dengan cara dan dosis tertentu.

Ada beberapa cara untuk menggambarkan proses kinetik obat dalam tubuh. Tiga kelas model farmakokinetik yang banyak digunakan adalah kompartemen, non-

kompartemen, dan model fisiologis. Walaupun model fisiologis memberikan gambaran yang paling akurat mengenai proses kinetik yang terjadi, tetapi membutuhkan beberapa percobaan dan data medis. Model non-kompartemen berdasarkan teori momen statistik dan membutuhkan lebih sedikit asumsi mengenai fisiologis distribusi obat dan mekanisme eliminasi obat. Pada model kompartemen, dilakukan penggabungan jaringan dan organ yang memiliki efek kinetik terhadap obat yang sama untuk membentuk satu kompartemen. Biasanya proses kinetik dalam sistem biologis bisa digambarkan dengan model satu kompartemen atau dua kompartemen.

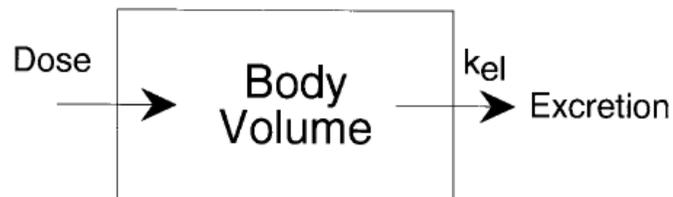
Sesungguhnya, tubuh manusia terdiri dari berjuta – juta model kompartemen berdasarkan konsentrasi obat yang berbeda pada sel atau jaringan. Akan tetapi, pada tubuh yang hidup kita hanya dapat mengakses dua tipe cairan tubuh, darah (atau plasma atau serum) dan urin. Model kompartemen digunakan untuk menggambarkan kinetika proses sistem biologis sesuai data eksperimen dari konsentrasi obat dalam darah terhadap waktu. Model kompartemen adalah model yang banyak digunakan oleh para peneliti di Indonesia dan para peneliti farmakokinetika lainnya.

Model kompartemen mana yang cocok untuk suatu obat tergantung pada jenis obatnya dan dapat diperkirakan dari profil kurva kadar obat dalam plasma terhadap waktu. Dalam penelitian farmakokinetik tentu saja harus digunakan model yang paling cocok untuk obat yang bersangkutan. Tetapi untuk perhitungan regimen dosis obat, yang harus cepat dan tidak perlu terlalu tepat karena selalu harus disesuaikan kembali menurut respon pasien, cukup digunakan model satu

kompartemen untuk pemberian oral dan kalau perlu model dua kompartemen untuk pemberian IV.

### 2.7.1 Model 1 Kompartemen

Menurut model ini, tubuh dianggap sebagai satu kompartemen yang memiliki kinetika yang sama dengan darah/ plasma, tempat obat menyebar dengan seketika dan merata ke seluruh cairan dan jaringan tubuh. Pada model satu kompartemen obat tidak harus masuk dalam sistem peredaran. Obat bisa ada pada seluruh cairan ekstraselular, jaringan lunak, atau seluruh tubuh, tetapi tidak terkumpul di satu tempat tertentu.



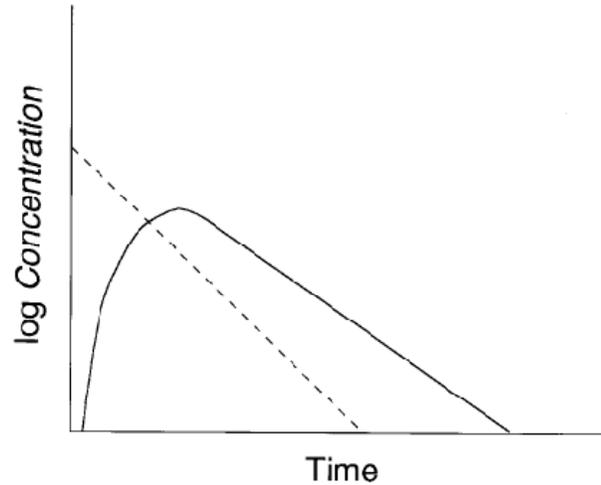
**Gambar 2.4 Diagram kotak model satu kompartemen**

Persamaan differensial dan solusinya yang menggambarkan model diatas adalah (Xiaoling Li, 2006)

$$\frac{d(C_p)}{dt} = -K(C_p)$$

$$C_p = C_p^0 e^{-Kt} = \frac{\text{dosis}}{Vd} e^{-Kt}$$

dimana  $K$  adalah ratio eliminasi presistemik, dan  $C_p^0$  adalah  $C_p$  ketika waktu awal.



**Gambar 2.5 Grafik log konsentrasi plasma terhadap waktu setelah pemberian obat secara intravenous (---) and oral (-) pada model satu kompartemen**

Grafik diatas menunjukkan perubahan konsentrasi obat terhadap waktu secara dinamis pada model satu kompartemen. Garis putus – putus menunjukkan perubahan konsentrasi setelah pemberian obat dengan injeksi intravena dan garis sambung menunjukkan perubahan konsentrasi setelah pemberian obat dengan oral. Karena pemberian obat dengan injeksi intravena tidak memiliki tahap resorpsi, maka grafik yang ditunjukkan linear. Sedangkan untuk pemberian obat dengan cara oral, konsentrasi obat pada darah secara perlahan mencapai konsentrasi puncak karena proses resorpsi oleh tubuh.

Persamaan deferensial dan solusinya dari pemodelan di atas adalah sebagai berikut(Xiaoling Li, 2006)

$$\frac{d(C_p)}{dt} = K_a(C_{absorb}) - K(C_p)$$

$$C_p = \frac{(F)(S)(dosis)(K_a)}{Vd(K_a - K)} (e^{-K(t)} - e^{-K_a(t)})$$

dimana  $K_a$  adalah ratio absorpsi per satuan waktu,  $K$  adalah ratio eliminasi per satuan waktu,  $V_d$  adalah volume distribusi,  $F$  adalah banyak bagian dari dosis yang diberikan yang masuk ke dalam sistem sirkulasi, dan  $S$  adalah formulasi faktor salt.  $V_d$  dapat dihitung dengan persamaan(Xiaoling Li, 2006):

$$V_d = \frac{(F)(S)(dosis)}{(AUC_{0 \rightarrow \infty})(K)}$$

yang merupakan persamaan yang sama dengan  $V_{d(area)}$  pada model dua kompartemen. Pada model satu kompartemen,  $V_{d(area)}$  diturunkan menjadi  $V_d$ . Dua parameter,  $C_{p,max}$  dan  $t_{max}$ , yang menunjukkan konsentrasi maksimal obat yang dapat dicapai dan waktu dimana konsentrasi maksimal obat mencapai titik maksimal, dapat dihitung dengan persamaan berikut(Xiaoling Li, 2006):

$$t_{max} = \frac{\ln\left(\frac{k_a}{K}\right)}{k_a - K}$$

$$C_{p,max} = \frac{(F)(S)(dosis)(K_a)}{Vd(K_a - K)} (e^{-K(t_{max})} - e^{-K_a(t_{max})})$$

## 2.8 Persamaan Matematika Untuk Pengobatan Dengan Dosis Berulang

### 2.8.1 Persamaan Matematika Konsentrasi Obat Dan Waktu Paruh

Secara umum dan sederhana, kecepatan dari eliminasi obat dalam peredaran darah proporsional dengan jumlah yang ada dalam peredaran darah saat itu. Oleh karena itu, jika  $C(t)$  adalah konsentrasi obat pada waktu  $t$ , maka fakta bahwa obat dieliminasi dari peredaran darah pada kecepatan yang proporsional dengan jumlah yang ada saat itu bisa dirumuskan sebagai berikut (Raina Robeva, 2008):

$$\frac{dC(t)}{dt} = -rC(t), \text{ dimana } r > 0.$$

dan solusi dari persamaan differensial diatas adalah

$$C(t) = C(0)e^{-rt}$$

Tanda negative pada persamaan diatas mengindikasikan konsentrasi obat dalam darah berkurang. Nilai konstan  $r$ , disebut kecepatan eliminasi konstan, mengontrol kecepatan obat akan dikeluarkan dari dalam darah. Semakin besar nilai  $r$ , maka semakin cepat proses eliminasinya.

Hal ini berhubungan dekat dengan waktu-paruh dari obat, yang didefinisikan sebagai waktu yang diperlukan untuk mengurangi konsentrasi obat dalam darah menjadi setengahnya. Dalam konsep matematika dengan menggunakan solusi persamaan differensial untuk konsentrasi obat di atas, maka akan didapat waktu-paruh ( $t^{1/2}$ ) obat adalah:

$$t^{1/2} = \frac{\ln(2)}{r}$$

### 2.8.2 Persamaan Matematika Untuk Dosis Berulang

Telah dijelaskan sebelumnya bahwa konsentrasi obat dalam darah setelah setiap pemberian dosis obat  $C$ , maka konsentrasi obat tersebut dalam darah dapat dihitung dengan persamaan  $C(t) = C(0)e^{-rt}$ . Oleh karena itu setelah pemberian dosis kedua, maka konsentrasi obat akan bertambah dan terakumulasi dan begitu seterusnya. Konsentrasi obat yang terakumulasi dari pemberian obat dengan dosis  $C$  dan interval waktu antar dosis  $T$  tersebut dapat dituliskan dalam bentuk

$$C + Ce^{-Tr} + C(e^{-Tr})^2 + C(e^{-Tr})^3 + \dots$$

Jika obat diberikan dalam dosis  $C$  dan interval waktu  $T$  yang sama, maka pada akhir periode ke- $n$  konsentrasi obat dalam darah menjadi (Raina Robeva, 2008):

$$R_n = C[(e^{-Tr})^n + (e^{-Tr})^{n-1} + (e^{-Tr})^{n-2} + \dots + e^{-Tr}]$$

dan ketika dosis obat selanjutnya diminum, maka persamaan tersebut menjadi:

$$R_n + C = C[(e^{-Tr})^n + (e^{-Tr})^{n-1} + (e^{-Tr})^{n-2} + \dots + e^{-Tr} + 1]$$

Kedua persamaan diatas dapat dibuat sederhana dengan menggunakan konsep deret geometri.

Sebuah deret dalam bentuk,

$$a + ab + ab^2 + ab^3 + \dots$$

adalah sebuah deret geometri. Jika kita menghitung jumlah dari  $n$  suku pertama dari deret tersebut  $S_n$ , maka:

$$S_n = a + ab + ab^2 + ab^3 + \dots + ab^{n-1}$$

Dan jika dikalikan dengan  $b$ ,

$$bS_n = ab + ab^2 + ab^3 + \dots + ab^{n-1} + ab^n$$

Sehingga

$$S_n - bS_n = (1 - b)S_n = a - ab^n$$

Oleh karena itu, untuk  $b \neq 1$ , maka jumlah  $n$  suku pertama dari deret geometri di atas adalah

$$S_n = \frac{a - ab^n}{1 - b} = a \frac{1 - b^n}{1 - b}$$

dan jika  $|b| < 1$ , maka limit dari persamaan di atas adalah

$$\lim_{n \rightarrow \infty} S_n = \frac{a}{1 - b}$$

Karena ketika  $|b| < 1$ ,  $\lim_{n \rightarrow \infty} b^n = 0$ .

Persamaan di atas jika disesuaikan dengan deret geometri yang dijelaskan di atas dengan  $a = Ce^{-Tr}$  dan  $b = e^{-Tr}$ , maka akan diperoleh (Raina Robeva, 2008):

$$R_n = Ce^{-Tr} \frac{1 - (e^{-Tr})^n}{1 - e^{-Tr}}$$

Dan untuk mengetahui apa yang akan terjadi ketika pemberian obat sudah berlangsung beberapa saat, maka kita harus menghitung  $\lim R_n$  ketika  $n \rightarrow \infty$ .

$$R = \lim_{n \rightarrow \infty} R_n = \frac{C e^{-Tr}}{1 - e^{-Tr}} = \frac{C}{e^{Tr} - 1}$$

dimana pada dosis berulang yang cukup banyak, konsentrasi obat akan stabil di sekitar nilai R, yang tergantung pada dosis C, interval waktu yang sama T, dan kecepatan eliminasi konstan r.

### 2.8.3 Regimen Dosis

Dengan mengetahui MEC, MTC dan waktu-paruh dari obat, kita dapat membuat suatu design pengobatan dengan manfaat maksimal dan aman. Dosis C yang sama harus diberikan pada interval T yang sama. Secara umum, setelah beberapa dosis, konsentrasi obat akan selalu berada di antara R dan R+C. Karena tujuan utamanya adalah untuk mempertahankan konsentrasi diantara MEC dan MTC, maka kita bisa menentukan R dan C dari kondisi (Raina Robeva, 2008):

$$R = \text{MEC} \quad \text{dan} \quad R + C = \text{MTC}$$

Karena MEC dan MTC setiap obat bisa diketahui, maka dosis C adalah:

$$C = \text{MTC} - \text{MEC}$$

Oleh karena itu, dengan nilai R dan C, kita dapatkan (Raina Robeva, 2008):

$$\text{MEC} = R = \frac{C}{e^{Tr} - 1} = \frac{\text{MTC} - \text{MEC}}{e^{Tr} - 1} \quad \text{dan} \quad T = \frac{1}{r} \ln \frac{\text{MTC}}{\text{MEC}}$$

Membuat semua dosis dalam jumlah yang sama akan mengakibatkan suatu kerugian yaitu diperlukan suatu periode tertentu sebelum konsentrasi obat mencapai MEC. Untuk beberapa obat, seperti antidepresan, periode tersebut sangat dibutuhkan untuk meminimalisir efek sampingnya. Sedangkan untuk banyak obat umum lainnya, dosis yang ditentukan memberikan toleransi untuk dosis pertama yang lebih besar untuk mencapai konsentrasi efektif maksimal secepat mungkin.

## **2.9 Microsoft C#**

Bahasa pemrograman C# (dibaca *C sharp*) adalah sebuah bahasa pemrograman modern yang bersifat *general purpose*, berorientasi objek yang dikembangkan oleh *Microsoft* sebagai inisiatif kerangka *.NET Framework*. Bahasa pemrograman ini diciptakan berbasiskan bahasa C++ yang telah dipengaruhi oleh aspek – aspek maupun fitur bahasa yang terdapat pada bahasa-bahasa pemrograman lainnya seperti *Java*, *Delphi*, dan *Visual Basic* dengan beberapa penyederhanaan. Bahasa pemrograman ini dikembangkan oleh sebuah tim pengembang di *Microsoft* yang dipimpin oleh Anders Hejlsberg, seseorang yang telah lama berpengalaman di dunia pengembangan bahasa pemrograman karena memang ialah yang membuat *Borland Turbo Pascal*, *Borland Delphi*, dan juga *Microsoft J++*.

Menurut standar *ECMA-334 C# Language Specification*, nama C# terdiri atas sebuah huruf Latin C (U+0043) yang diikuti oleh tanda pagar yang menandakan # (U+0023). Tanda pagar # yang digunakan memang bukan tanda

kres dalam seni musik (U+266F), dan tanda pagar # (U+0023) tersebut digunakan karena karakter kres dalam seni musik tidak terdapat di dalam *keyboard* standar.

### 2.9.1 Sejarah C#

Pada akhir dekade 1990-an, *Microsoft* membuat program *Microsoft J++* sebagai sebuah langkah percobaan untuk menggunakan *Java* di dalam sistem operasi *Windows* untuk meningkatkan antarmuka dari *Microsoft Component Object Model (COM)*. Akan tetapi, akibat masalah dengan pemegang hak cipta bahasa pemrograman *Java*, *Sun Microsystems*, *Microsoft* pun menghentikan pengembangan *J++*, dan beralih untuk membuat pengganti *J++*, kompilernya dan mesin virtualnya sendiri dengan menggunakan sebuah bahasa pemrograman yang bersifat *general purpose*. Untuk menangani proyek ini, *Microsoft* merekrut Anders Helsberg, yang merupakan mantan karyawan *Borland* yang membuat bahasa *Turbo Pascal*, dan *Borland Delphi*, yang juga mendesain *Windows Foundation Classes (WFC)* yang digunakan di dalam *J++*. Sebagai hasil dari usaha tersebut, *C#* pun pertama kali diperkenalkan pada bulan Juli 2000 sebagai sebuah bahasa pemrograman modern berorientasi objek yang menjadi sebuah bahasa pemrograman utama di dalam pengembangan di dalam *platform Microsoft .NET Framework*.

*C#* didesain untuk memenuhi kebutuhan akan sintaksis *C++* yang lebih ringkas dan *Rapid Application Development* yang 'tanpa batas' (dibandingkan dengan *RAD* yang 'terbatas' seperti yang terdapat pada *Delphi* dan *Visual Basic*). Seperti halnya bahasa *Java*, bahasa *C#* telah membuang beberapa fitur berbahaya

dari bahasa C. Memang, *pointer* belum sepenuhnya "dicabut" dari C#, tetapi sebagian besar pemrograman dengan menggunakan bahasa C# tidak membutuhkan *pointer* secara ekstensif, seperti halnya C dan C++. Persamaan lainnya antara Java dan C# mencakup peran dari kompiler. Pengalaman Helsberg sebelumnya dalam pendesain bahasa pemrograman seperti *Visual J++*, *Delphi*, *Turbo Pascal*) dengan mudah dilihat dalam sintaksis bahasa C#, begitu pula halnya pada inti *Common Language Runtime (CLR)*. Untuk menjalankan suatu program C#, di komputer atau alat elektronik lain yang bersangkutan harus tersedia CLR (*Common Language Runtime*).

Jalannya sebuah program C# akan dikelola sepenuhnya oleh CLR, oleh karena itu program C# dikatakan sebagai program yang *managed*. Sedangkan program - program klasik yang langsung berinteraksi dengan operating system maupun hardware tertentu disebut program *unmanaged*. Biasanya, kompiler menterjemahkan kode sumber (berkas teks yang berisi bahasa pemrograman tingkat tinggi) ke dalam kode mesin. Kode mesin tersebut membentuk sebuah berkas yang dapat dieksekusi (*executable* atau EXE), yang berupa sebuah berkas yang siap untuk dijalankan kapan saja secara langsung oleh komputer. Tetapi, karena kode mesin hanya diasosiasikan dengan sebuah jenis mesin tertentu saja, berkas yang dapat dieksekusi tersebut hanya dapat berfungsi di atas satu jenis komputer saja. Inilah sebabnya mengapa program yang sama tidak berfungsi di atas sistem operasi *Windows* di atas sistem operasi *GNU* atau *Linux*, *Apple Macintosh* atau sistem operasi lainnya, dan begitu pula sebaliknya.

Alat bantu kompiler yang digunakan oleh C# tidak menterjemahkan kode sumber ke dalam kode mesin, tetapi hanya menterjemahkan ke dalam sebuah

bahasa perantara atau *Intermediate Language* (disingkat menjadi *IL*), yang merupakan sebuah jenis kode mesin yang telah digeneralisasikan. Ketika program dijalankan di atas sebuah mesin, maka *IL* akan diterjemahkan ke dalam kode mesin secara keseluruhan. Dilihat dari perspektif pengguna, proses translasi ini tidak terlihat. Tetapi, dalam teorinya, ternyata di balik itu terdapat proses dua langkah rumit yang mengizinkan program dengan bahasa *IL* yang sama untuk berjalan di atas mesin yang berbeda. Selain itu, sebuah program dalam bentuk *IL* dapat diuji lebih mudah oleh sistem operasi dari keberadaan kode yang merusak atau kode yang mencurigakan. Kemampuan ini telah menjadi lebih penting saat program tersebut dipertukarkan melalui jaringan publik, seperti halnya Internet.

Bahasa *C*, *C++*, *Java* dan *C#* kini dikenal dengan sebutan "keluarga besar bahasa pemrograman C" atau "bahasa pemrograman berbasis bahasa C". *C++* mengandung semua hal yang dimiliki oleh *C* tetapi memiliki fitur yang tidak dimiliki oleh *C*. *Java* dan *C#* meskipun masih berbasis bahasa *C*, keduanya tidaklah serta merta merupakan pengganti dari bahasa *C*. Bahasa *Java* dan *C#* memiliki kesamaan dalam berbagai bidang, walaupun mirip dengan bahasa *C++*. Meskipun demikian, semuanya menggunakan banyak sintaksis yang mirip, seperti *void*, *int*, *struct* dan lain sebagainya.

### **2.9.2 Tujuan Desain C#**

Standar *European Computer Manufacturer Association* (*ECMA*) mendaftarkan beberapa tujuan desain dari bahasa pemrograman *C#*, sebagai berikut.

- a. Bahasa pemrograman *C#* dibuat sebagai bahasa pemrograman yang bersifat *general-purpose* (untuk tujuan jamak), berorientasi objek, modern, dan sederhana.
- b. Bahasa pemrograman *C#* ditujukan untuk digunakan dalam mengembangkan komponen perangkat lunak yang mampu mengambil keuntungan dari lingkungan terdistribusi.
- c. Kemudahan *programmer* sangatlah penting, khususnya bagi *programmer* yang telah lama menggunakan bahasa pemrograman *C* dan *C++*.
- d. *C#* ditujukan agar cocok digunakan untuk menulis program aplikasi baik mulai dari program aplikasi yang sangat besar yang menggunakan sistem operasi yang canggih hingga kepada program aplikasi yang sangat kecil yang memiliki fungsi-fungsi tertentu.
- e. Meskipun aplikasi *C#* ditujukan agar bersifat 'ekonomis' dalam hal kebutuhan pemrosesan dan memori komputer, bahasa *C#* tidak ditujukan untuk bersaing secara langsung dengan kinerja dan ukuran program aplikasi yang dibuat dengan menggunakan bahasa pemrograman *C* dan bahasa rakitan.
- f. Bahasa *C#* harus mencakup pengecekan jenis (*type checking*) yang kuat, pengecekan larik (*array*), pendeteksian percobaan terhadap penggunaan variabel-variabel yang belum diinisialisasikan dan kemudahan kode sumber.

## 2.10 Perancangan Perangkat Lunak

Perangkat lunak adalah (1) perintah (program komputer) yang bila dieksekusi memberikan fungsi dan unjuk kerja seperti yang diinginkan, (2) struktur data yang

memungkinkan program memanipulasi informasi secara proporsional, dan (3) dokumen yang menggambarkan operasi dan kegunaan program. Menurut Ian Sommerville (2007), perancangan perangkat lunak adalah disiplin perancangan yang berhubungan dengan semua aspek dari produksi perangkat lunak dari tahap awal spesifikasi sistem sampai dengan pemeliharaan setelah sistem dalam tahap berjalan.

Menurut Pressman (2005, p24), rekayasa piranti lunak mencakup 3 elemen yang mampu mengontrol proses pengembangan piranti lunak, yaitu:

- a. *Methods*, berfungsi untuk menyediakan cara-cara teknis untuk membangun piranti lunak.
- b. *Tools*, berfungsi untuk mengadakan dukungan otomatis atau semi-otomatis untuk metode-metode seperti *Computer Aided Software Engineering (CASE)* yang memadukan *software*, *hardware*, dan *software engineering database*.
- c. *Procedures*, merupakan pengembangan dari *methods* dan *tools*.

### **2.10.1 Software Development Life Cycle**

Dalam perancangan software dikenal istilah *software life cycle* yaitu serangkaian kegiatan yang dilakukan selama masa perancangan software. Pemakaian jenis *software life cycle* yang cocok salah satunya ditentukan oleh jenis bahasa pemrograman yang cocok. Contohnya, *Waterfall Model* merupakan model yang paling umum dan paling dasar pada *software life cycle*. *Rapid Application Development (RAD)* dan *Joint Application Development (JAD)* cocok untuk *software* berbasis objek (*OOP*), sedangkan *Sync+Stabilize* dan *Spiral Model* yang merupakan pengembangan model *waterfall* dengan komponen *prototyping* cocok untuk sebuah aplikasi yang rumit dan cenderung mahal pembuatannya.

Menurut Dix (1997, p180), visualisasi dari kegiatan pada *software life cycle* model *waterfall* adalah sebagai berikut.

a. Spesifikasi kebutuhan (*Requirement specification*)

Pada tahap ini, pihak pengembang dan konsumen mengidentifikasi apa saja fungsi-fungsi yang diharapkan dari sistem dan bagaimana sistem memberikan layanan yang diminta. Pengembang berusaha mengumpulkan berbagai informasi dari konsumen.

b. Perancangan arsitektur (*Architectural design*)

Pada tahap ini, terjadi pemisahan komponen-komponen system sesuai dengan fungsinya masing-masing.

c. *Detailed design*

Setelah memasuki tahap ini, pengembang memperbaiki deskripsi dari komponen-komponen dari sistem yang telah dipisah – pisah pada tahap sebelumnya.

d. *Coding and unit testing*

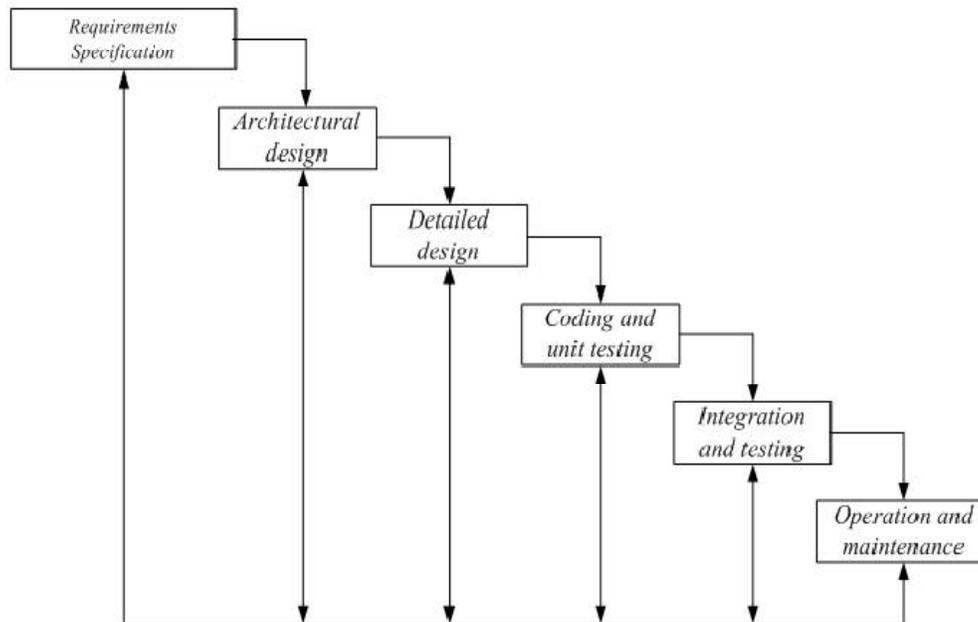
Pada tahap ini, disain diterjemahkan ke dalam bahasa pemrograman untuk dieksekusi. Setelah itu komponen – komponen dites apakah sesuai dengan fungsinya masing-masing.

e. *Integration and testing*

Setelah tiap – tiap komponen dites dan telah sesuai dengan fungsinya, komponen-komponen tersebut disatukan lagi. Lalu sistem dites untuk memastikan sistem telah sesuai dengan kriteria yang diminta konsumen.

*f. Pemeliharaan (maintenance)*

Setelah sistem diimplementasikan, maka perlu dilakukannya perawatan terhadap sistem itu sendiri. Perawatan yang dimaksud adalah perbaikan *error* yang ditemukan setelah sistem diimplementasikan.



**Gambar 2.6 Waterfall Model untuk Sistem Software Life-cycle**

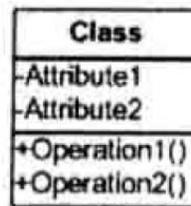
### 2.10.2 Unified Modelling Language

*Unified Modelling Language (UML)* adalah bahasa grafis yang standar untuk memodelkan *software object oriented* (Lethbridge, 2002, p151). *UML* mengandung tipe diagram yang bervariasi, termasuk:

- *Class Diagram*
- *Sequence Diagram*
- *Use Case Diagram*

## A. Class Diagram

*Class diagram* adalah salah satu diagram struktur statis yang menunjukkan struktur dari sistem dengan menunjukkan *class-class* yang ada pada sistem, *attribute* dan *method class-class* tersebut, dan hubungan antar *class*. Hubungan *class* terdiri dari *link*, *association*, *aggregation*, dan *composition*.

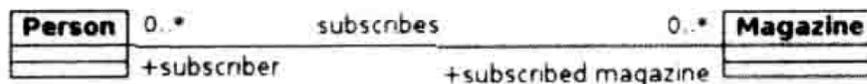


**Gambar 2.7 Notasi Class**

**Sumber: Lethbridge (2002, p439)**

*Link* adalah hubungan dasar antar obyek yang menggambarkan garis penghubung antara dua atau lebih *class*. *Link* merupakan bagian dari *association*.

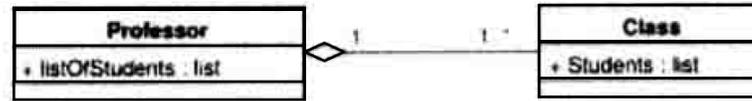
*Association* menggambarkan kumpulan *link* yang saling berhubungan. *Binary Association* (dengan dua titik akhir) biasanya digambarkan sebagai sebuah garis, di mana masing-masing titik akhir dihubungkan dengan sebuah *class*. *Association* memiliki dua atau lebih titik akhir.



**Gambar 2.8 Hubungan Association pada Class Diagram**

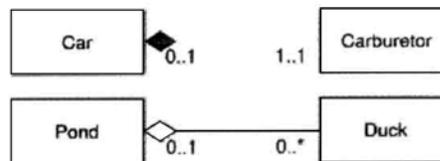
*Aggregation* adalah lambang dari memiliki sebuah atau hubungan *association*, tetapi *aggregation* lebih spesifik dari *association*. Meskipun *aggregation* merupakan perluasan *association*, hubungan *aggregation* hanya dapat melibatkan dua *class*. *Aggregation* terjadi bila suatu *class* mengandung

satu atau lebih obyek dari *class* lain, tetapi *class* yang dikandung tidak memiliki *life cycle dependency* dengan *class* yang mengandung.



**Gambar 2.9 Hubungan Aggregation pada Class Diagram**

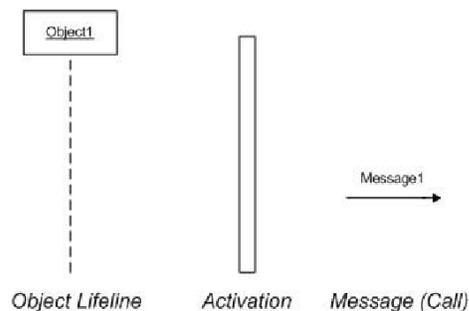
*Composition* merupakan hubungan *aggregation* di mana *class* yang dikandung telah memiliki *life cycle dependency* dengan *class* yang mengandung.



**Gambar 2.10 Hubungan Composition pada Class Diagram**

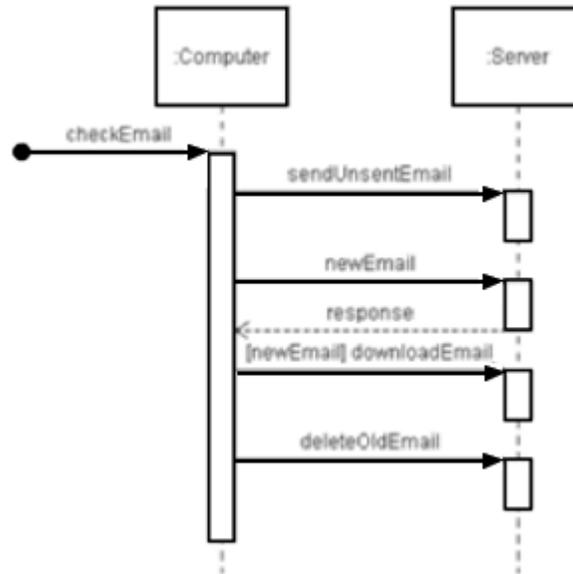
## B. Sequence Diagram

*Sequence diagram* adalah diagram yang menunjukkan urutan proses dan penukaran pesan oleh sejumlah objek (dan seorang aktor yang opsional) dalam melakukan tugas tertentu. *Sequence diagram* menggambarkan skenario *runtime* sederhana secara grafis.



**Gambar 2.11 Notasi Object, Message, dan Activation**

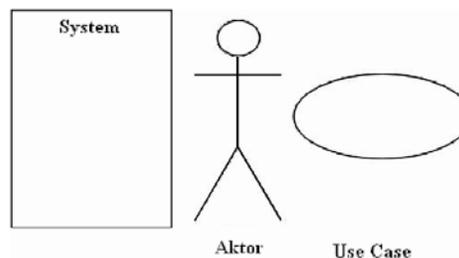
Sumber: Lethbridge (2002, p.440)



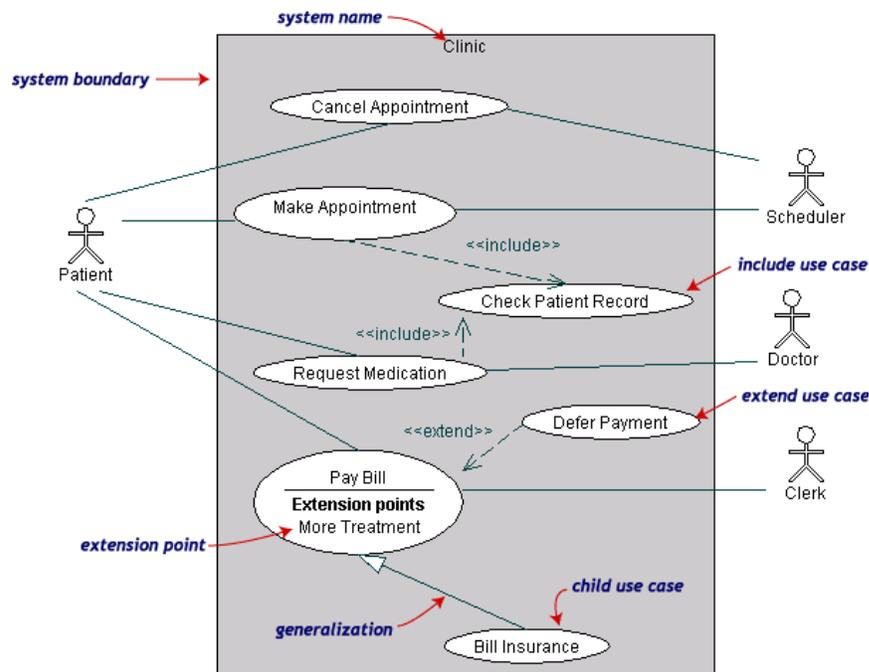
**Gambar 2.12** Contoh *Sequence Diagram*

### C. Use Case Diagram

*Use case diagram* menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah 'apa' yang diperbuat sistem, dan bukan 'bagaimana'. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem. *Use case* merupakan sebuah pekerjaan tertentu, misalnya *login* ke sistem, menciptakan sebuah daftar belanja, dan sebagainya. Seorang aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu.



**Gambar 2.13** Notasi *Use Case Diagram*



**Gambar 2.14** Contoh *Use Case Diagram*

### 2.10.3 Interaksi Manusia dan Komputer

Menurut Shneiderman (2005, p4), Interaksi manusia dan komputer merupakan disiplin ilmu yang berhubungan dengan, perancangan, evaluasi, dan implementasi sistem komputer interaktif untuk digunakan oleh manusia, serta studi fenomena-fenomena besar yang berhubungan dengannya.

Pada interaksi manusia dan komputer ditekankan pada pembuatan antarmuka pemakai (*user interface*), dimana *user interface* yang dibuat diusahakan sedemikian rupa sehingga seorang *user* dapat dengan baik dan nyaman menggunakan aplikasi perangkat lunak dibuat. Antar muka pemakai (*user interface*) adalah bagian sistem komputer yang memungkinkan manusia berinteraksi dengan komputer. Tujuan antar muka pemakai adalah agar sistem komputer dapat digunakan oleh pemakai (*user interface*), istilah tersebut

digunakan untuk menunjuk kepada kemampuan yang dimiliki oleh piranti lunak atau program aplikasi yang mudah dioperasikan dan dapat membantu menyelesaikan suatu persoalan dengan hasil yang sesuai dengan keinginan pengguna atau biasa disebut *user friendly*.

Pedoman untuk menghasilkan suatu rancangan antar muka program yang *user friendly* adalah dengan menggunakan pedoman *Eight Golden Rules*. *EightGolden Rules* tersebut menjelaskan mengenai beberapa aturan yang diperbolehkan dan tidak diperbolehkan sebagai pedoman untuk merancang antar muka program. Kedelapan aturan tersebut antara lain sebagai berikut.

- a. *Strive for consistency*, konsistensi dalam perancangan antar muka.
- b. *Enable frequent user to use shortcuts*, memungkinkan pengguna menggunakan *shortcuts* secara berkala.
- c. *Offer informative feed back*, memberikan umpan balik yang informative.
- d. *Design dialogs to yield closure*, merancang dialog untuk menghasilkan keadaan akhir.
- e. *Offer simple error handling*, memberikan penanganan kesalahan yang sederhana.
- f. *Permit easy reversal of actions*, mengizinkan pembalikkan aksi dengan mudah.
- g. *Support internal locus of control*, mendukung pengguna menguasai system yang dibuat.
- h. *Short-term memory load*, mengurangi beban jangka pendek kepada pengguna.