

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **II.1. Analisis Sistem**

Sistem merupakan serangkaian bagian yang saling tergantung dan bekerja sama untuk mencapai tujuan tertentu. Suatu sistem pasti tersusun dari sub-sub sistem yang lebih kecil yang juga saling tergantung dan bekerja sama untuk mencapai tujuan. Sebagai contoh, sistem administrasi universitas terdiri dari sub-sub sistem administrasi fakultas dan sub-sistem fakultas terdiri dari sub-sub sistem administrasi jurusan.

Tujuan dasar suatu sistem tergantung pada jenis sistem itu sendiri. Sebagai contoh, sistem peredaran darah manusia merupakan sistem biologi yang memiliki tujuan untuk mengedarkan darah yang mengandung oksigen dan sari makanan ke seluruh tubuh. Sedangkan sistem buatan manusia seperti sistem yang terdapat di sekolah, organisasi bisnis, atau instansi pemerintah juga mempunyai tujuan yang berbeda. Organisasi bisnis biasanya memiliki tujuan yang lebih jelas, seperti yang telah disebutkan pada bagian sebelumnya, yaitu mendapatkan laba.

Sistem informasi yang kadang kala disebut sebagai sistem pemrosesan data, merupakan sistem buatan manusia yang biasanya terdiri dari sekumpulan komponen, baik manual ataupun berbasis komputer yang terintegrasi untuk mengumpulkan, menyimpan dan mengelola data serta menyediakan informasi kepada pihak yang berkepentingan sebagai pemakai informasi tersebut (Anastasia Diana ; 2011 : 3).

## II.2. Pengertian Sistem

Untuk mengawali pembahasan tentang analisis dan perancangan sistem informasi, pemahaman akan sistem terlebih dahulu harus ditekankan. Definisi sistem berkembang sesuai dengan konteks di mana pengertian sistem itu digunakan. Berikut akan diberikan beberapa definisi sistem secara umum :

Kumpulan dari bagian-bagian yang bekerja sama untuk mencapai tujuan yang sama. Contoh :

1. Sistem Tatasurya
2. Sistem Pencernaan
3. Sistem Transportasi Umum
4. Sistem Otomatif
5. Sistem Komputer
6. Sistem Informasi

Sekumpulan objek-objek yang saling berelasi dan berinteraksi serta hubungan antar objek bisa dilihat sebagai satu kesatuan yang dirancang untuk mencapai satu tujuan.

Dengan demikian, secara sederhana sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur atau variable-variabel yang saling terorganisasi, saling berinteraksi, dan saling bergantung sama lain. Sistem sebagai seperangkat elemen yang digabungkan satu dengan lainnya untuk satu tujuan bersama. Sementara definisi sistem dalam kamus *Webster's Unbringed* adalah elemen-elemen yang saling berhubungan dan membentuk satu kesatuan atau organisasi (Hanif Al Fatta ; 2007 : 3).

Sistem merupakan sekumpulan elemen-elemen yang saling terintegrasi serta melaksanakan fungsinya masing-masing untuk mencapai tujuan yang telah ditetapkan. Karakteristik sistem terdiri dari :

1. Komponen Sistem

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk suatu kesatuan. Komponen-komponen sistem atau elemen-elemen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem.

2. Batasan Sistem

Batasan merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang suatu kesatuan. Batasan suatu sistem menunjukkan ruang lingkup (*scope*) dari sistem tersebut.

3. Lingkungan Luar Sistem

Lingkungan luar dari suatu sistem adalah apapun diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut.

4. Penghubung Sistem

Penghubung merupakan media penghubung antara satu subsistem dengan subsistem lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem lainnya.

#### 5. Masukan Sistem

Masukan sistem adalah energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan (*maintenance input*) dan masukan sinyal (*signal input*).

#### 6. Keluaran Sistem

Keluaran sistem adalah hasil energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan.

#### 7. Pengolah Sistem

Suatu sistem dapat mempunyai suatu bagian pengolah atau sistem itu sendiri sebagai pengolahnya. Pengolah akan mengubah masukan menjadi keluaran.

#### 8. Sasaran Sistem

Suatu sistem mempunyai tujuan (*goal*) atau sasaran (*objective*). Kalau suatu sistem tidak mempunyai sasaran, maka operasi sistem tidak ada gunanya (Sulindawati ; 2010 : 135).

### **II.3. Pengertian Informasi**

Informasi adalah data yang berguna yang telah diolah sehingga dapat dijadikan dasar untuk mengambil keputusan yang tepat. Informasi sangat penting bagi organisasi. Pada dasarnya informasi adalah penting seperti sumber daya yang lain, misalnya peralatan, bahan, tenaga, dsb.

Informasi yang berkualitas dapat mendukung keunggulan kompetitif suatu organisasi. Dalam sistem informasi akuntansi, kualitas dari informasi yang disediakan merupakan hal penting dalam kesuksesan sistem.

Secara konseptual seluruh sistem organisasional mencapai tujuannya melalui proses alokasi sumber daya, yang diwujudkan melalui proses pengambilan keputusan manajerial. Informasi memiliki nilai ekonomik pada saat mendukung keputusan alokasi sumber daya, sehingga dengan demikian mendukung sistem untuk mencapai tujuan.

Pemakai informasi akuntansi dapat dibagi dalam dua kelompok besar: ekstern dan intern. Pemakai ekstern mencakup pemegang saham, investor, kreditor, pemerintah, pelanggan, pemasok, pesaing, serikat pekerja, dan masyarakat. Pemakai intern terutama para manajer, kebutuhannya bervariasi tergantung pada tingkatannya (Agustinus ; 2012 : 1).

#### **II.4. Pengertian Akuntansi**

Menurut Rita Eni (2007 : 3) Ada beberapa definisi mengenai akuntansi di mana masing-masing definisi tersebut memandang akuntansi dari sudut yang berbeda. Pada umumnya akuntansi didefinisikan dengan melihat proses kegiatannya. Definisi akuntansi dilihat dari kegiatannya merupakan suatu proses pencatatan, penggolongan, peringkasan, pelaporan dan penganalisisan data keuangan dari suatu organisasi atau perusahaan. Kegiatan akuntansi diawali dengan mencatat setiap transaksi yang terjadi ke dalam jurnal dan kemudian memasukkan catatan tersebut ke dalam masing rekening buku besar (*posting*). Setelah itu diadakan peringkasan terhadap data pada buku besar dan proses kegiatan akuntansi akan berakhir dengan penyajian laporan keuangan yang antara lain berupa neraca, laporan rugi laba, dan laporan perubahan modal.

## II.5. Sistem Informasi Akuntansi

Sistem informasi akuntansi adalah sistem yang bertujuan untuk mengumpulkan dan memproses data serta melaporkan informasi yang berkaitan dengan saksi keuangan. Lingkup sistem informasi akuntansi dapat dijelaskan dari manfaat yang didapat dari informasi akuntansi. Manfaat atau tujuan sistem informasi akuntansi tersebut adalah sebagai berikut :

1. Mengamankan harta / kekayaan perusahaan. Harta / kekayaan di sini meliputi kas perusahaan, persediaan barang dagangan, termasuk aset tetap perusahaan.
2. Menghasilkan beragam informasi untuk pengambilan keputusan. misal, pengelola toko swalayan memerlukan informasi mengenai barang apa saja yang diminati oleh konsumen. Membeli barang yang kurang laku berarti kas akan terjebak dalam persediaan dan berarti kehilangan kesempatan untuk membeli barang dagangan yang laku.
3. Menghasilkan informasi untuk pihak eksternal. Setiap pengelola usaha memiliki kewajiban untuk membayar pajak. Besarnya pajak yang dibayar tergantung pada omset penjualan (jika pengelola memilih menggunakan norma dalam perhitungan pajaknya) atau tergantung pada laba rugi usaha (jika pengelola memilih untuk tidak menggunakan norma dalam perhitungan pajaknya).
4. Menghasilkan informasi untuk penilaian kinerja karyawan atau divisi. Sistem informasi dapat juga dimanfaatkan untuk penilaian kinerja karyawan atau divisi.

5. Menyediakan data masa lalu untuk kepentingan audit (pemeriksaan). Data yang tersimpan dengan baik sangat memudahkan proses audit (pemeriksaan).
6. Menghasilkan informasi untuk penyusunan dan evaluasi anggaran perusahaan. Anggaran merupakan alat yang sering digunakan perusahaan untuk mengendalikan pengeluaran kas.
7. Menghasilkan informasi yang diperlukan dalam kegiatan perencanaan dan pengendalian. Selain berguna untuk membandingkan informasi yang berkaitan dengan anggaran dan biaya standar dengan kenyataan seperti yang telah dikemukakan (Anastasia Diana ; 2011 : 6).

## **II.6. Pengertian Surat Permintaan Pembayaran (SPP)**

Surat permintaan pembayaran atau disingkat SPP adalah Dokumen yang diajukan oleh Bendahara Pengeluaran kepada Pengguna Anggaran/Kuasa Pengguna Anggaran melalui PPK-SKPD berdasarkan SPD yang telah dikeluarkan untuk permintaan pembayaran langsung kepada pihak ketiga atas dasar perjanjian kontrak atau surat perintah kerja lainnya dan pembayaran gaji dengan jumlah, penerima, peruntukan dan waktu pembayaran tertentu yang dokumennya disiapkan oleh Pejabat Pelaksana Teknis Kegiatan (PPTK).

SPP diajukan dengan SPD sebagai dasar jumlah yang diminta untuk dibayarkan kepada SKPD, SPP memiliki 4 (empat) jenis, yaitu:

- a. SPP-UP (Surat Permintaan Pembayaran-Uang Persediaan)
- b. SPP-GU (Surat Permintaan Pembayaran-Ganti Uang)
- c. SPP-TU (Surat Permintaan Pembayaran-Tambahan Uang)

- d. SPP-LS (Surat Permintaan Pembayaran-Langsung), SPP-LS Barang dan Jasa dan SPP-LS Belanja Bunga, Hibah, Bantuan dan Tak Terduga (Ayu Raini ; 2009 : 1).

## II.7. Pengertian Java

Java adalah bahasa pemrograman yang dapat dijalankan di berbagai jenis komputer dan berbagai sistem operasi termasuk telepon genggam. Java dikembangkan oleh *Sun Microsystem* dan dirilis tahun 1995. Java merupakan suatu teknologi perangkat lunak yang digolongkan *multi platform*. Selain itu, Java juga merupakan suatu *platform* yang memiliki *virtual machine* dan *library* yang diperlukan untuk menulis dan menjalankan suatu program.

Bahasa pemrograman java pertama lahir dari *The Green Project*, yang berjalan selama 18 bulan, dari awal tahun 1991 hingga musim panas 1992. Proyek tersebut belum menggunakan *versi* yang dinamakan Oak. Proyek ini dimotori oleh Patrick Naughton, Mike Sheridan, James Gosling dan Bill Joy, serta Sembilan pemrograman lainnya dari *Sun Microsystem*. Salah satu hasil proyek ini adalah mascot Duke yang dibuat oleh Joe Palrang (Wahana Komputer ; 2010 : 1).

## II.8. Pengertian NetBeans

NetBeans merupakan salah satu proyek *open source* yang disponsori oleh *Sun Microsystem*. Proyek ini berdiri pada tahun 2000 dan telah menghasilkan 2 produk, yaitu NetBeanss IDE dan NetBeans Platform. NetBeans IDE merupakan produk yang digunakan untuk melakukan pemrograman baik menulis kode, meng-*compile*, mencari kesalahan dan mendistribusikan program. Sedangkan NetBeans

Platform adalah sebuah modul yang merupakan kerangka awal / pondasi dalam bangun aplikasi desktop yang besar.

NetBeans juga menyediakan paket yang lengkap dalam pemrograman dari pemrograman standar (aplikasi desktop), pemrograman *enterprise*, dan pemrograman perangkat mobile. Saat ini NetBeans telah mencapai versi 6.8 (Wahana Komputer ; 2010 : 15).

## II.9. Pengertian Database

*Database* merupakan kumpulan data yang saling berhubungan, hubungan antar data dapat ditunjukkan dengan adanya *field* kunci dari setiap tabel yang beda.

Dalam satu *file* atau tabel terdapat *record-record* yang sejenis, sama besar, sama bentuk, yang merupakan satu kumpulan entitas yang seragam. Satu *record* terdiri dari *field* yang saling berhubungan menunjukkan bahwa *fiel* tersebut satu pengertian yang lengkap dan disimpan dalam satu *record*. Basis data mempunyai beberapa kriteria penting yaitu :

1. Bersifat data oriented dan bukan program oriented.
2. Dapat digunakan oleh beberapa program aplikasi tanpa perlu mengubah basis datanya.
3. Dapat dikembangkan dengan mudah, baik *volume* maupun strukturnya.
4. Dapat memenuhi kebutuhan sistem-sistem baru secara mudah.
5. Dapat digunakan dengan cara-cara yang berbeda.

Prinsip utama *database* adalah pengaturan data dengan tujuan utama fleksibel dan kecepatan pada saat pengambilan data kembali. Adapun ciri-ciri basis data di antaranya adalah sebagai berikut :

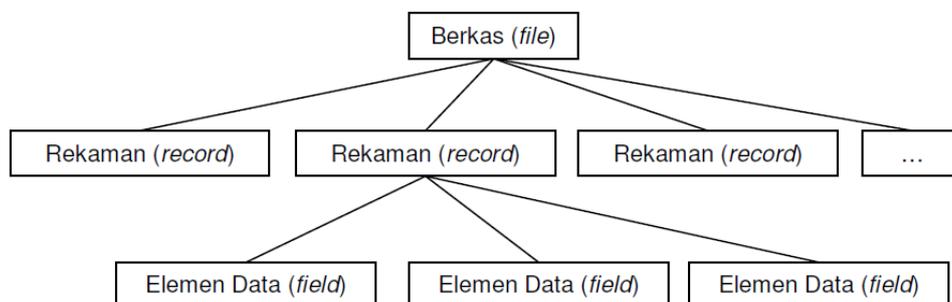
1. Efisiensi meliputi kecepatan, ukuran dan ketepatan.
2. Data dalam jumlah besar.
3. Berbagi pakai (dipakai bersama-sama atau *sharebility*).

Mengurangi bahkan menghilangkan terjadinya duplikasi dan data yang tidak konsisten (Windu Gata ; 2013 : 19).

### II.9.1. Hirarki Database

Data diorganisasikan ke dalam bentuk elemen data (*field*), rekaman (*record*), dan berkas (*file*). Definisi dari ketiganya adalah sebagai berikut:

Elemen data adalah satuan data terkecil yang tidak dapat dipecah lagi menjadi unit lain yang bermakna. Misalnya data siswa terdiri dari NIS, Nama, Alamat, Telepon atau Jenis Kelamin. Rekaman merupakan gabungan sejumlah elemen data yang saling terkait. Istilah lain dari rekaman adalah baris atau tupel. Berkas adalah himpunan seluruh rekaman yang bertipe sama (Haidar Dzacko ; 2007 : 1).



**Gambar II.1. Hirarki Data**  
(Sumber : Haidar Dzacko ; 2007 : 1)

## II.9.2. Model Database

Model data dapat dikelompokkan berdasarkan konsep pembuatan deskripsi struktur basis data, yaitu:

1. Model data konseptual (*high level*) menyajikan konsep tentang bagaimana *user* memandang atau memperlakukan data. Dalam model ini dikenalkan tiga konsep penyajian data yaitu:
  - a. *Entity* (entitas) merupakan penyajian obyek, kejadian atau konsep dunia nyata yang keberadaannya secara eksplisit didefinisikan dan disimpan dalam basis data, contohnya Mahasiswa, Matakuliah, Dosen, Nilai dan lain sebagainya.
  - b. *Attribute* (atribut) adalah keterangan-keterangan yang menjelaskan karakteristik dari suatu entitas seperti NIM, Nama, Fakultas, Jurusan untuk entitas Mahasiswa.
  - c. *Relationship* (hubungan) merupakan hubungan atau interaksi antara satu entitas dengan yang lainnya, misalnya entitas pelanggan berhubungan dengan entitas barang yang dibelinya.
2. Model data fisik (*low level*) merupakan konsep bagaimana deskripsi detail data disimpan ke dalam komputer dengan menyajikan informasi tentang format rekaman, urutan rekaman, dan jalur pengaksesan data yang dapat membuat pencarian rekaman data lebih efisien.
3. Model data implementasi (*representational*) merupakan konsep deskripsi data disimpan dalam komputer dengan menyembunyikan sebagian detail deskripsi data sehingga para *user* mendapat gambaran global bagaimana data disimpan

dalam komputer. Model ini merupakan konsep model data yang digunakan oleh model hierarki, jaringan dan relasional (Haidar Dzacko ; 2007 : 3).

## **II.10. Pengertian MySQL**

MySQL adalah program *database* yang mampu mengirim dan menerima data dengan sangat cepat dan *multiuser*. MySQL memiliki dua bentuk lisensi, yaitu *freeware* dan *shareware*. penulis sendiri dalam menjelaskan buku ini menggunakan *database* ini untuk keperluan pribadi atau usaha tanpa harus membeli atau membayar lisensi, yang berada di bawah lisensi GNU/GPL (*general public license*) (Wahana ; 2010 : 5).

## **II.11. Entity Relationship Diagram (ERD)**

*Entity Relationship Diagram* atau ERD merupakan salah satu alat (tool) berbentuk grafis yang populer untuk *desain database*. Tool ini relatif lebih mudah dibandingkan dengan Normalisasi. Kebanyakan sistem analis memakai alat ini, tetapi yang jadi masalah, kalau kita cermati secara seksama, tool ini mencapai 2NF (Yuniar Supardi ; 2010 : 448).

## **II.12. Kamus Data**

Kamus data (*data dictionary*) mencakup definisi-definisi dari data yang disimpan di dalam basis data dan dikendalikan oleh sistem manajemen basis data. Figur 6.5 menunjukkan hanya satu tabel dalam basis data jadwal. Struktur basis data yang dimuat dalam kamus data adalah kumpulan dari seluruh definisi *field*, definisi tabel, relasi tabel, dan hal-hal lainnya. Nama *field* data, jenis data (seperti teks atau angka atau tanggal), nilai-nilai yang valid untuk data, dan karakteristik-

karakteristik lainnya akan disimpan dalam kamus data. Perubahan-perubahan pada struktur data hanya dilakukan satu kali di dalam kamus data, program-program aplikasi yang mempergunakan data tidak akan ikut terpengaruh (Raymond McLeod ; 2008 : 171).

### **II.13. Teknik Normalisasi**

Normalisasi adalah teknik perancangan yang banyak digunakan sebagai pemandu dalam merancang basis data relasional. Pada dasarnya, normalisasi adalah proses dua langkah yang meletakkan data dalam bentuk tabulasi dengan menghilangkan kelompok berulang lalu menghilangkan data yang terduplikasi dari tabel rasional.

Teori normalisasi didasarkan pada konsep bentuk normal. Sebuah tabel relasional dikatakan berada pada bentuk normal tertentu jika tabel memenuhi himpunan batasan tertentu. Ada lima bentuk normal yang telah ditemukan (Janner Simarmata ; 2010 : 76).

Salah satu topik yang cukup kompleks dalam dunia manajemen *database* adalah proses untuk menormalisasi tabel-tabel dalam *database relasional*. Dengan normalisasi kita ingin mendesain *database relasional* yang terdiri dari tabel-tabel berikut :

1. Berisi data yang diperlukan.
2. Memiliki sesedikit mungkin redundansi.
3. Mengakomodasi banyak nilai untuk tipe data yang diperlukan.
4. Mengefisienkan *update*.

5. Menghindari kemungkinan kehilangan data secara tidak disengaja/tidak diketahui.

Alasan utama dari normalisasi *database* minimal sampai dengan bentuk normal ketiga adalah menghilangkan kemungkinan adanya “*insertion anomalies*”, “*deletion anomalies*”, dan “*update anomalies*”. Tipe-tipe kesalahan tersebut sangat mungkin terjadi pada *database* yang tidak normal.

#### **II.14. UML (*Unified Modeling Language*)**

Hasil pemodelan pada OOAD terdokumentasikan dalam bentuk *Unified Modeling Language* (UML). UML adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak.

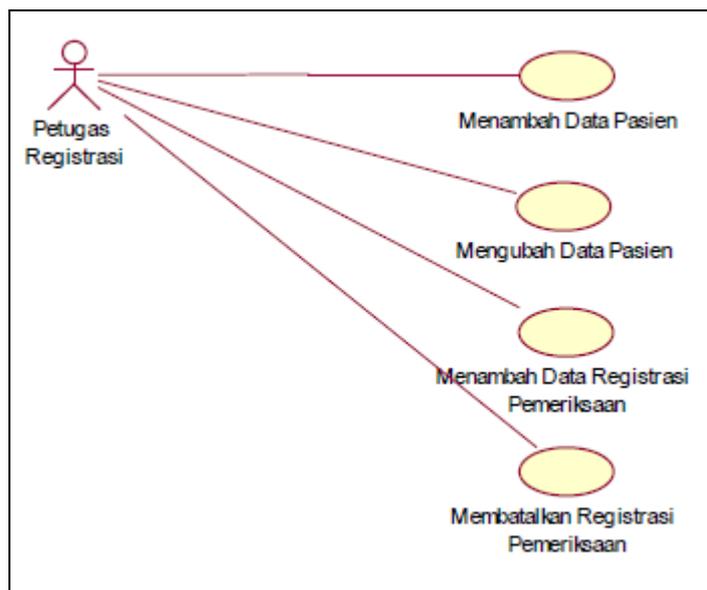
UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan sistem. UML saat ini sangat banyak dipergunakan dalam dunia industri yang merupakan standar bahasa pemodelan umum dalam industri perangkat lunak dan pengembangan sistem (Haviluddin ; 2011 : 3).

Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis UML adalah sebagai berikut :

##### **1. Use Case Diagram**

*Use case diagram* dibuat berdasarkan pengguna sistem informasi rawat jalan dan fungsi-fungsi yang ditangani oleh sistem informasi rawat jalan yang didefinisikan pada sub bab analisis sistem rawat jalan yang akan dibuat. Aktor-aktor pada *use case diagram* didapatkan dari pengguna sistem informasi rawat

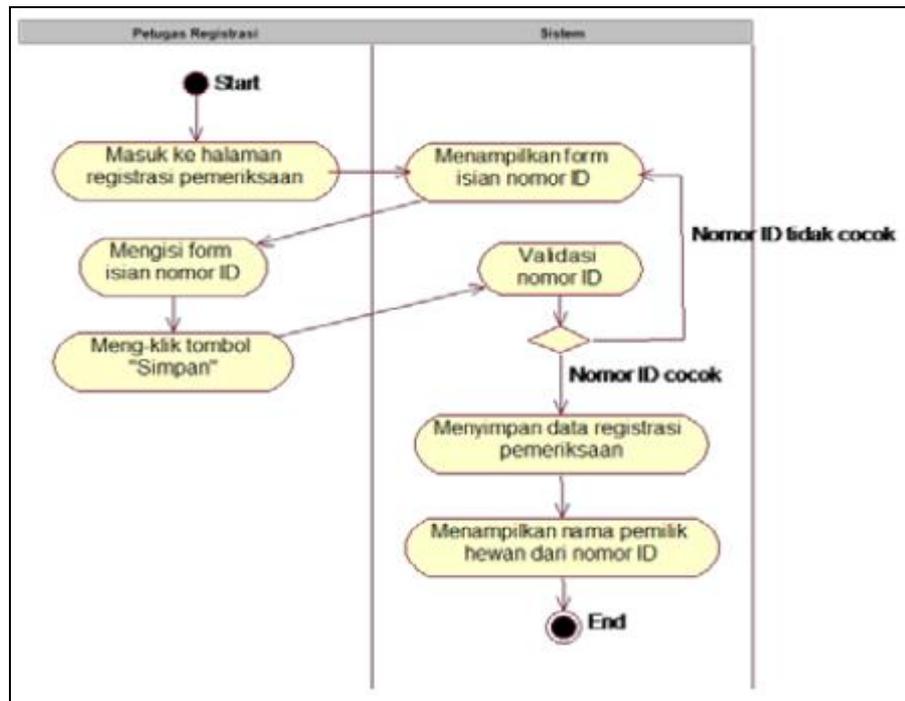
jalan, sedangkan *use case* untuk setiap aktor didapatkan dari fungsi-fungsi yang ditangani oleh sistem informasi rawat jalan. Contoh gambar *use case* untuk aktor petugas registrasi dan dokter pada rancangan sistem informasi rawat jalan yang akan dibuat dapat dilihat pada Gambar.



**Gambar II.2 Contoh Usecase diagram**

## 2. Activity Diagram

*Activity diagram* dibuat berdasarkan aliran dasar dan aliran alternatif pada skenario *use case diagram*. Pada *activity diagram* digambarkan interaksi antara aktor pada *use case diagram* dengan sistem. Contoh *activity diagram* yang dibuat berdasarkan skenario pada *use case* menambah data registrasi pemeriksaan dan menambah data pemeriksaan medis lanjutan dapat dilihat pada Gambar

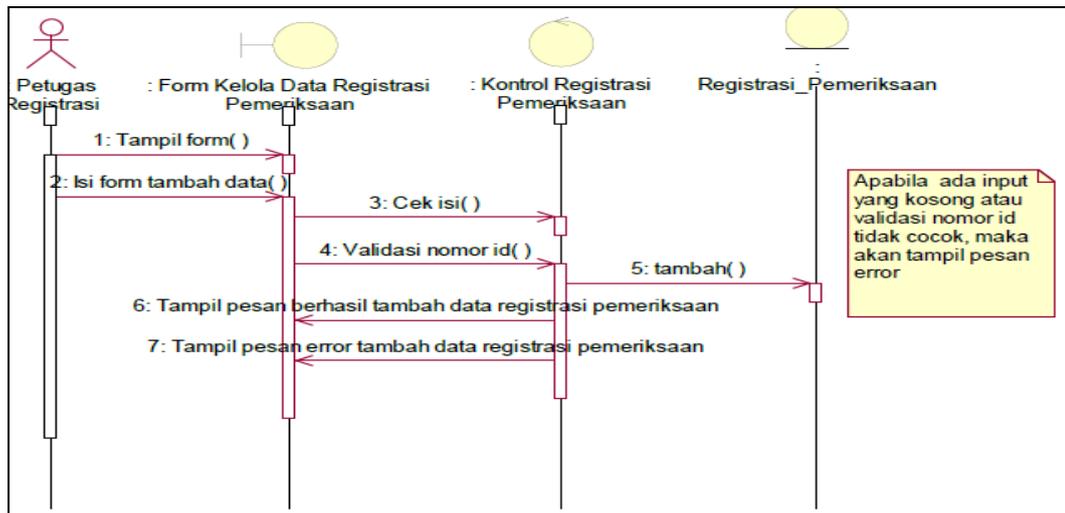


**Gambar II.3. Contoh Activity Diagram**

### 3. *Sequence diagram*

*Sequence diagram* dibuat berdasarkan *activity diagram* dan *class diagram*.

*Sequence diagram* menggambarkan aliran pesan yang terjadi antar kelas yang dideskripsikan pada *class diagram* dengan menggunakan operasi yang dimiliki kelas tersebut. Untuk aliran pesan, *sequence diagram* merujuk pada alur sistem *activity diagram* yang telah dibuat sebelumnya. Berikut adalah contoh *sequence diagram* menambah data registrasi pemeriksaan :

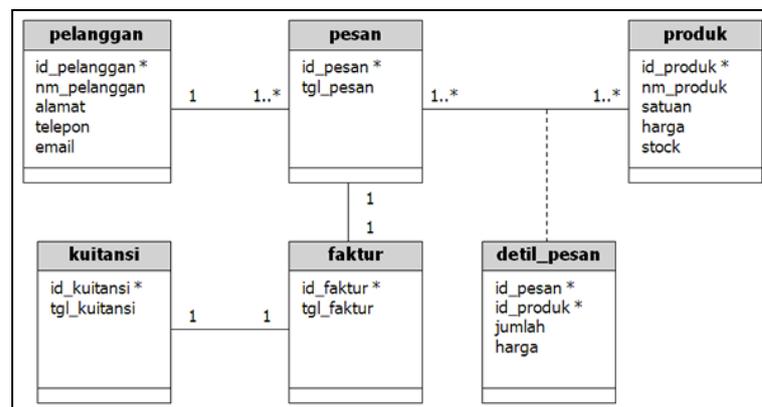


Gambar II.4. Contoh *Sequence Diagram*

#### 4. Class Diagram

*Class diagram* dibuat berdasarkan *use case diagram* dan *activity diagram*.

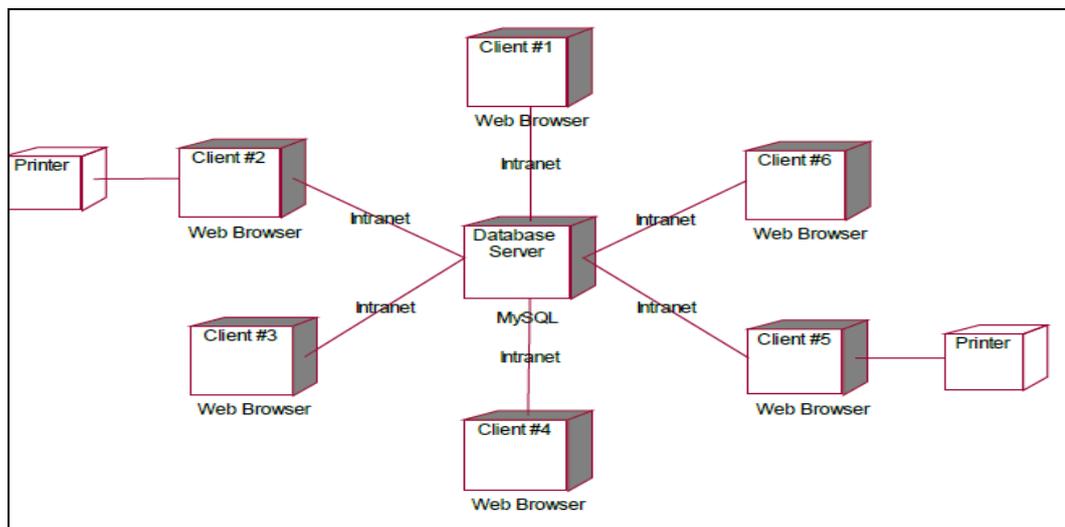
*Class diagram* dapat mendeskripsikan kelas-kelas yang digunakan dalam sistem informasi rawat jalan yang akan dibuat. Kelas dengan jenis *actor* diperoleh dari aktor-aktor yang digambarkan dalam *use case diagram*, sedangkan kelas dengan jenis lain seperti *entity*, *boundary*, dan *control* diperoleh dari gambaran alur sistem pada *activity diagram*. Berikut adalah contoh setiap jenis kelas :



Gambar II.5. Contoh *Class Diagram*

## 5. Deployment Diagram

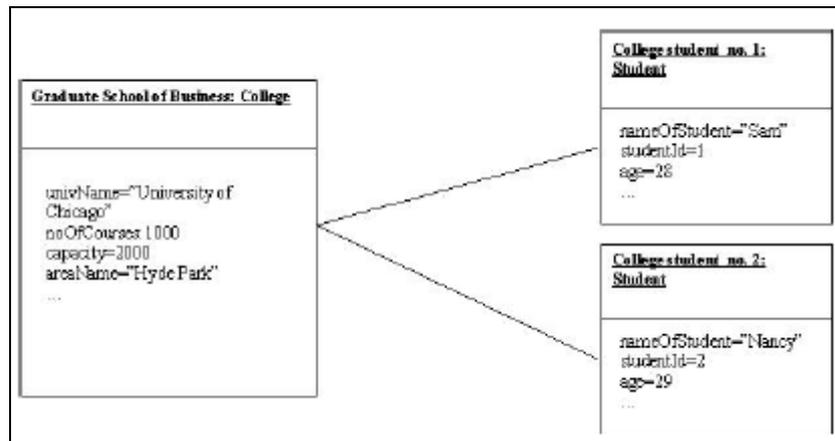
*Deployment diagram* dibuat untuk menunjukkan tata letak sebuah sistem secara fisik, menampakan bagian-bagian *software* yang berjalan pada bagian-bagian *hardware*, dan keterhubungan antara komponen-komponen *hardware* tersebut.



**Gambar II.6. Contoh *Deployment Diagram***

## 6. Objek Diagram

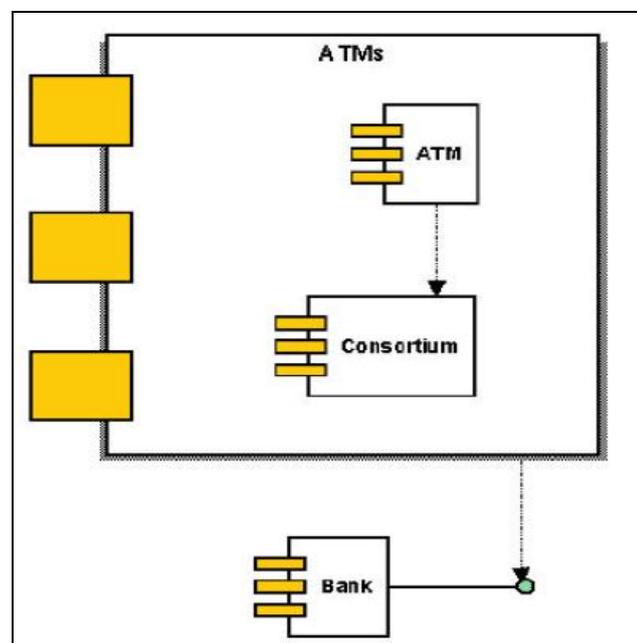
*Object diagram* menggambarkan kejelasan kelas dan warisan dan kadang-kadang diambil ketika merencanakan kelas, atau untuk membantu pemangku kepentingan non-program yang mungkin menemukan diagram kelas terlalu abstrak.



**Gambar II.7. Contoh Objek Diagram**

### 7. *Component diagram*

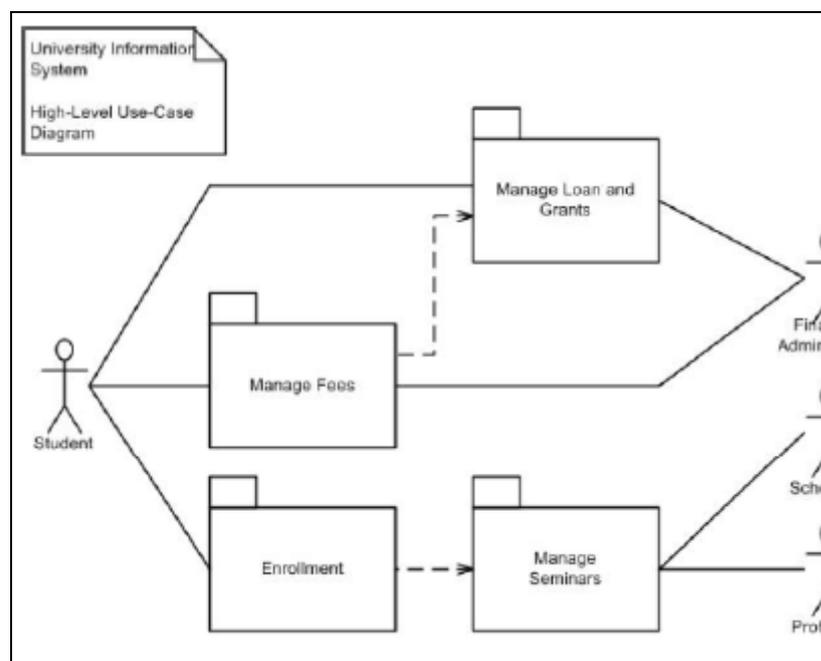
*Component diagram* menggambarkan struktur fisik dari kode, pemetaan pandangan logis dari kelas proyek untuk kode aktual di mana logika ini dilaksanakan.



**Gambar II.8. Contoh *Component Diagram***

## 8. Package Diagram

Paket diagram biasanya digunakan untuk menggambarkan tingkat organisasi yang tinggi dari suatu proyek *software*. Atau dengan kata lain untuk menghasilkan diagram ketergantungan paket untuk setiap paket dalam Pohon Model.



**Gambar II.9. Contoh Package Diagram**