

## BAB III

### LANDASAN TEORI

#### 3.1 Konsep Dasar Sistem Informasi

Menurut Jerry, dkk dalam Hartono (1999:1) menyatakan bahwa sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu. Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya (Hartono, 1999: 8)

Menurut Robert dalam Hartono (1999: 11) menyatakan bahwa sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan.

John Burch dan Gary Grudnitski dalam Hartono (1999: 12) mengemukakan bahwa sistem informasi terdiri dari komponen-komponen yang disebutnya dengan istilah blok bangunan (*building block*), yaitu:

1. Blok Masukan

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

## 2. Blok Model

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

## 3. Blok Keluaran

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

## 4. Blok Teknologi

Teknologi merupakan “kotak alat” (*tool box*) dalam sistem informasi. Teknologi digunakan untuk menerima *input*, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian dari sistem secara keseluruhan.

## 5. Blok Basis Data

Basis data (*database*) merupakan kumpulan dari data yang saling berhubungan satu dengan lainnya, tersimpan di perangkat keras computer dan digunakan perangkat lunak untuk memanipulasinya. Data perlu disimpan di dalam basis data untuk keperluan penyediaan informasi lebih lanjut. Data di dalam basis data perlu diorganisasikan sedemikian rupa, supaya informasi yang ditampilkan berkualitas. Organisasi basis data yang baik juga berguna untuk efisiensi kapasitas penyimpanannya. Basis data diakses atau dimanipulasi dengan menggunakan perangkat lunak paket yang disebut dengan DBMS (*Database Management System*).

## 6. Blok Kendali

Banyak hal yang dapat merusak sistem informasi, seperti misalnya bencana alam, api, temperatur, air, debu, kecurangan-kecurangan, kegagalan-kegagalan sistem sistem itu sendiri, kesalahan-kesalahan, inefisiensi, sabotase dan lain sebagainya. Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah ataupun bila terlanjur terjadi kesalahan-kesalahan dapat langsung cepat diatasi.

### 3.2 Terminal Curah Kering

Terminal adalah tempat keluar masuknya alat transportasi, baik itu alat transportasi untuk manusia maupun alat transportasi untuk barang. Curah Kering adalah istilah pihak pengelola transportasi yang artinya adalah komoditi yang bersifat kering misalnya batubara, pasir, kerikil, dll. Terminal curah kering sendiri adalah pelabuhan tempat bersandarnya kapal pengangkut komoditi curah kering serta tempat untuk melakukan pembongkaran muatan dari kapal menuju alat angkut berikutnya yaitu truk atau menuju tempat penumpukan muatan sementara yaitu gudang

### 3.3 Sistem Informasi Terminal Curah Kering

Sistem Informasi Terminal Curah Kering merupakan sebuah aplikasi yang bertujuan untuk mempermudah jalannya transaksi, mulai dari permohonan penyandaran kapal dari pihak pemohon, penjadwalan sandar kapal, penghitungan beban muatan, sampai dengan jumlah uang yang harus dibayar pihak pemohon

kepada petugas terminal. Selama ini proses masih dilakukan secara manual dengan pencatatan secara manual pula sehingga proses menjadi lambat dan memungkinkan adanya rekayasa. Sistem Informasi Terminal Curah Kering dapat membuat transaksi dalam terminal curah kering menjadi lebih cepat, efisien, sehingga bisa meningkatkan penghasilan bagi PT. Pelindo III.

### 3.4 Analisa dan Perancangan Sistem

Analisis sistem dapat didefinisikan sebagai penguraian dari suatu sistem informasi yang utuh ke dalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan, kesempatan, hambatan yang terjadi dan kebutuhan yang diharapkan sehingga dapat diusulkan perbaikannya.

Analisis sistem dilakukan setelah tahap perencanaan sistem dan sebelum tahap perancangan sistem. Langkah-langkah dasar dalam melakukan analisa sistem :

1. *Identify*, yaitu mengidentifikasi masalah.
2. *Understand*, yaitu memahami kinerja dari sistem yang ada.
3. *Analyze*, yaitu menganalisa sistem
4. *Report*, yaitu membuat laporan hasil analisis

Setelah analisis sistem dilakukan, tahap selanjutnya adalah perancangan sistem. Perancangan sistem mempunyai dua tujuan utama, yaitu untuk memenuhi kebutuhan kepada pemakai serta memberikan gambaran yang jelas dan lengkap kepada pemrogram dan ahli teknik lainnya yang terlibat (Jogiyanto, HM, 1990: 23).

### 3.5 Unified Modeling Language

Menurut Nugroho (2010:6), UML adalah bahasa pemodelan untuk sistem atau perangkat lunak yang berparadigma berorientasi objek. Pemodelan sesungguhnya digunakan untuk penyederhanaan permasalahan-permasalahan yang kompleks sedemikian rupa sehingga lebih mudah dipelajari dan dipahami. UML dikembangkan oleh 3 pendekar 'berorientasi objek', yaitu Grady Booch, Jim Rumbaugh, dan Ivar Jacobson. UML menjadi bahasa yang bisa digunakan untuk berkomunikasi dalam perspektif obyek antara user dengan *developer*, antara *developer* dengan *developer*, antara *developer* analisis dengan *developer* desain dan antara *developer* desain dengan *developer* program (Hermawan, 2008: 7).

### 3.6 Entity Relationship Diagram (ERD)

*Entity Relationship Diagram* menurut Jogiyanto:1990 digunakan untuk menggambarkan hubungan antar *entity* yang terlibat dalam sistem yang akan di buat. Jenis *relationship* diagram dapat berbentuk:

- a. *One to One* yaitu relasi satu lawan satu yang terjadi bila satu record yang ada pada satu entity/tabel hanya punya satu relasi pada file lain. Misalnya suatu departemen hanya mengerjakan satu jenis pekerjaan saja dan satu pekerjaan hanya dikerjakan oleh satu departemen saja.
- b. *One to Many* yaitu relasi satu lawan banyak yang terjadi bila *record* dengan kunci tertentu pada satu file mempunyai relasi banyak *record* pada file lain. Misalnya suatu pekerjaan hanya dikerjakan oleh satu departemen saja, namun suatu departemen dapat mengerjakan beberapa macam pekerjaan sekaligus.

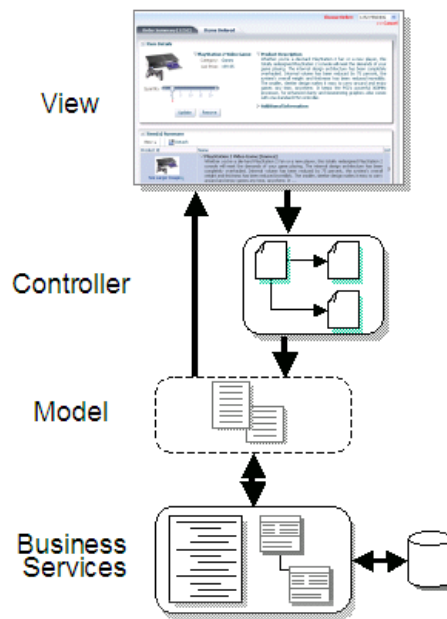
*Many to Many* yaitu relasi banyak lawan banyak yang terjadi bila kedua file saling mempunyai relasi banyak *record* pada file yang lain. Misalnya satu departemen mampu mengerjakan banyak pekerjaan, juga satu pekerjaan dapat ditangani oleh banyak departemen.

### 3.7 Oracle Software

#### 3.7.1 Application Development Framework (ADF)

*Oracle Application Development Framework* (Oracle ADF) adalah sebuah *end-to-end application framework* yang dibangun di atas *platform* JAVA *Enterprise Edition* (JAVA EE) standar dan teknologi *open source* yang memudahkan dan mempercepat dalam mengimplementasi aplikasi yang berorientasi layanan. Oracle ADF menyediakan *environment* yang melingkupi seluruh siklus pengembangan perangkat lunak, mulai dari pembuatan desain sampai dengan *deployment* aplikasi, dengan hanya melakukan *drag-and-drop* dari *data binding*, desain antar muka dan fitur pengembangan aplikasi bersama tim seperti sudah adanya fitur sub-versioning (SVN) yang digunakan untuk melakukan penyimpanan dan pembaruan data pada 1 sumber .

Aplikasi yang dibuat menggunakan *Fusion web technology* benar-benar dibedakan antara *business logic*, *page navigation* dan *user interface* dengan mengacu pada arsitektur model-view-controller (MVC). Arsitektur MVC pada Oracle ADF dapat dilihat pada Gambar 3.1.



Gambar 3.1 Arsitektur Model-View-Controller pada Oracle ADF

- Lapisan *model* merepresentasikan nilai dari data yang saling berhubungan kedalam halaman aplikasi.
- Lapisan *view* berisi halaman antar muka yang digunakan untuk melihat dan mengubah data tersebut.
- Lapisan *controller* memproses *input*-an dari pengguna dan menentukan arah perpindahan halaman.

Lapisan *business service* menangani akses data dan proses enkapsulasi dari *business logic*.

### 3.7.2 Oracle 10g Enterprise Edition

Oracle database 10g Enterprise Edition adalah *tools database* yang ideal untuk perusahaan yang membutuhkan dukungan dari proses *Online Transaction Processing* yang ber-*volume* tinggi dan *query* yang intensif dari aplikasi datawarehouse. Oracle 10g menyediakan skalabilitas dari pengaturan semua

perangkat keras., dan juga bisa digunakan untuk mengelola informasi dengan jumlah yang sangat besar, dengan jaminan keamanan tingkat tinggi di dalam sebuah industri. Oracle 10g Enterprise Edition juga menyediakan keuntungan yang unik yaitu melindungi data dari kesalahan yang dilakukan oleh manusia, mengurangi downtime yang berasosiasi dengan pemeliharaan secara rutin, dan termasuk kemampuan *self-managing* untuk membantu mengurangi biaya operasional.

Berikut ini adalah fungsi dan keunggulan yang dimiliki oleh Oracle 10g

Enterprise:

- a. Mendukung di semua lingkungan sistem operasi
- b. Mengelola semua data
- c. Mengintegrasikan semua informasi yang ada
- d. Bisa dijalankan di semua aplikasi
- e. Tersedia setiap waktu
- f. Jaminan keamanan yang terpercaya
- g. Mudah dalam instalasi dan mudah dikelola
- h. Dikembangkan untuk *Grid Computing*

### 3.8 Jaspersoft iReport

iReport adalah sebuah *tools* yang digunakan untuk mendesain sebuah laporan. Penggunaan iReport sendiri hampir sama dengan penggunaan Crystal Report yang biasa digunakan oleh *developer* VB.NET, bedanya adalah iReport adalah aplikasi *open source* yang dibangun dengan dasar bahasa pemrograman java dan digunakan untuk membuat laporan untuk program yang menggunakan



bahasa java juga. iReport bisa berinteraksi secara dinamis dengan banyak software database, diantaranya : Oracle dan MySQL. iReport membutuhkan Sun Java SDK versi 1.5 atau versi yang lebih baru agar bisa melakukan *compiling* pada *script* yang telah dibuat, bukan hanya dengan meng-*install Java Runtime Environment* (JRE). Untuk perangkat kerasnya, seperti semua program java yang sudah ada, iReport memakan banyak alokasi memori, jadi minimal memori yang dialokasikan sebanyak 256MB dari *Random Access Memory* (RAM) dan 20MB dari ruang penyimpanan di *hard disk* (Toffoli, 2007:1).

