

## BAB II

### LANDASAN TEORI

#### 2.1 OPI

Menurut PT. PLN (PERSERO) Distribusi Jawa Timur (2012), metode OPI atau *Operational Performance Improvement* adalah metode perbaikan kinerja suatu unit atau bidang kerja secara holistik dan melibatkan fungsi di bidang teknik, manajemen, sistem organisasi dan kesiapan sumber dayanya (SDM). Oleh karena itu metode OPI terbagi dalam 3 *workstream* (kerangka kerja) yaitu: *Technical System* (TS), *Management Infrastucture* (MI), dan *Mindset Capabilities – Leadership* (MCL).

Implementasi OPI ini melalui tiga tahap yaitu *Diagnose*, *Design* dan *Deliver*. *Diagnose* adalah tahap dimana mencari *bottleneck* yang terjadi pada proses bisnis (tahap identifikasi *gap*), selanjutnya *Design* adalah tahapan mencari ide-ide perbaikan untuk menutup atau menghilangkan *gap-gap* yang sudah diidentifikasi dan selanjutnya adalah tahap implementasi atau *Deliver*, di tahap inilah ide-ide perbaikan dijalankan dan dimonitor pelaksanaannya, dampaknya serta hasilnya.

#### 2.2 Aplikasi OPTIMUS, OPI dengan *Microsoft Excel*, dan OPTIMUS+

Untuk Meningkatkan efisiensi dan efektifitas dalam Monitoring dan Tracking Program *Initiative* OPI, Tim OPI Kantor Pusat sudah membuat sebuah aplikasi untuk program tersebut. OPTIMUS (OPI Tracking *Initiative Machine Upload System*), yang sebelumnya sudah di *launching* pada Desember 2011, saat ini sudah dengan tampilan dan penggunaan lebih maksimal. *Upload File*, *Entry*

*Initiative*, *Workplan*, *Task*, *News*, *Artikel*, dll, sudah bisa digunakan di aplikasi OPTIMUS ini. (PT. PLN (PERSERO), t.t.).

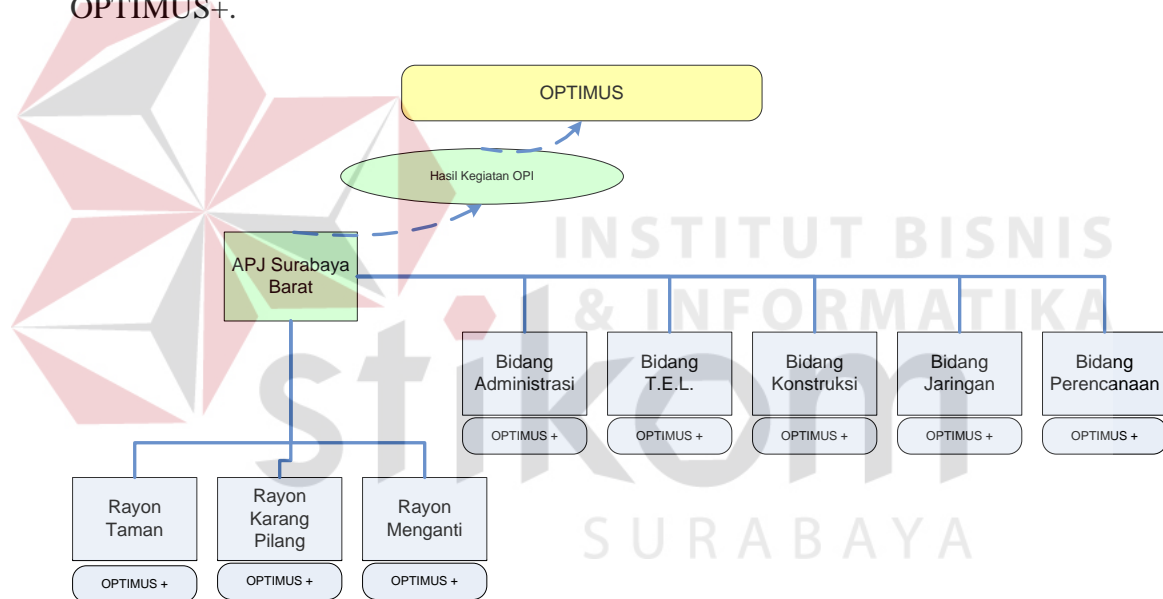
Menurut Modul Pelatihan OPTIMUS (2012), *OPTIMUS Application is a tool that will allow managing the tasks/Activities associated with Initiatives. We can discuss the progress of the Initiatives, associate important files to relevant tasks, share reports, manage workflow etc.* Dengan fitur-fitur yang dimiliki OPTIMUS tersebut, PLN mengelola hasil kegiatan OPI dari seluruh PLN Area.

Pada PLN Rayon (Taman, Karang Pilang, dan Menganti) dan Tingkat Bagian (Asman Perencanaan, Asman Konstruksi, Asman Jaringan, Asman T. E. L., dan Asman Administrasi), proses OPI dikelola dengan *Microsoft Excel*. Sama seperti OPTIMUS, OPI dengan *Microsoft Excel* juga mengelola *initiative* dan *workplan*. Dasarnya pun sama, yaitu OPI.

Setiap Rayon memiliki *workplan* masing-masing. *Workplan* ini bisa dimiliki oleh Rayon dan Tingkat Bagian/Bidang itu sendiri atau diperoleh dari penurunan dari *workplan* yang dimiliki Area. Penggunaan *Microsoft Excel* oleh PLN Rayon dan Tingkat Bagian/Bidang adalah untuk memantau *workplan* yang ada di Rayon maupun Tingkat Bagian/Bidang itu sendiri.

Ada beberapa permasalahan pada OPI dengan *Microsoft Excel*. Permasalahan seperti penyajian informasi *workplan* yang statis, belum adanya informasi yang tepat untuk manajer untuk kebutuhan pemantauan, dan belum adanya modul *EMI Survey Online*, dapat memberikan dampak yang kurang baik. Oleh karena itu, dibutuhkan adanya aplikasi baru yang dapat mengatasi permasalahan tersebut, yaitu OPTIMUS+.

Nama OPTIMUS+ diambil dari OPTIMUS, aplikasi OPI Pusat, karena OPTIMUS+ memiliki dasar yang sama dengan OPTIMUS, yaitu OPI. Hanya saja, nantinya OPTIMUS+ akan digunakan dalam lingkup Rayon sampai Tingkat Bagian/Bidang menggantikan OPI dengan *Microsoft Excel*. OPTIMUS+ digunakan untuk menampung hasil kegiatan OPI di Rayon dan Tingkat Bagian/Bidang. Hasil kegiatan OPI Rayon dan Tingkat Bagian/Bidang yang berkaitan dengan *workplan* Area, nantinya akan digabungkan ke dalam OPTIMUS. Berikut ini adalah gambar hubungan antara OPTIMUS dan OPTIMUS+.



Gambar 2.1 Gambar Hubungan OPTIMUS dan OPTIMUS+

### 2.3 SMS Gateway

Menurut Purnamasari (2010), SMS atau *Short Message Services* merupakan salah satu fasilitas atau layanan pada Teknologi Komunikasi Bergerak yang paling banyak digunakan saat ini karena biaya murah, prosesnya cepat, dan dapat langsung diterima oleh tujuan.

Cara kerja SMS Gateway adalah:

1. Nomor *Mobile-Station*
2. SMS Gateway akan memproses pengambilan data dari *database* sesuai dengan format *database* yang telah ditentukan
3. Kemudian SMS Gateway akan mengirimkan kepada *user* sesuai dengan format SMS yang telah ditentukan.

## 2.4 Testing Software

Menurut Romeo (2003), *testing software* adalah proses mengoperasikan software dalam suatu kondisi yang di kendalikan, untuk verifikasi apakah telah berlaku sebagaimana telah ditetapkan (menurut spesifikasi), mendeteksi *error*, dan validasi apakah spesifikasi yang telah ditetapkan sudah memenuhi keinginan atau kebutuhan dari pengguna yang sebenarnya. Verifikasi adalah pengecekan atau pengetesan entitas-entitas, termasuk *software*, untuk pemenuhan dan konsistensi dengan melakukan evaluasi hasil terhadap kebutuhan yang telah ditetapkan. Validasi adalah melihat kebenaran sistem, apakah proses yang telah dilakukan adalah apa yang sebenarnya diinginkan atau dibutuhkan oleh *user*. Jadi, dapat disimpulkan bahwa *testing* merupakan tiap-tiap aktifitas pengumpulan informasi yang dibutuhkan untuk melakukan evaluasi atau mengukur suatu atribut dari *software*.

*Testing software* dilakukan untuk mendapatkan informasi *reliable* terhadap *software* dengan cara termudah dan paling efektif, antara lain:

1. Apakah *software* telah siap digunakan ?
2. Apa saja risikonya ?
3. Apa saja kemampuannya ?

4. Apa saja keterbatasannya ?
5. Apa saja masalahnya ?
6. Apakah telah berlaku seperti yang diharapkan ?

#### **2.4.1 Black box Testing**

*Black box testing*, dilakukan tanpa pengetahuan detail struktur internal dari sistem atau komponen yang ditest, juga disebut sebagai *behavioral testing*, *specification-based testing*, *input / output testing* atau *functional testing*. *Black box testing* berfokus pada kebutuhan fungsional pada *software*, berdasarkan pada spesifikasi kebutuhan dari *software*. Kategori *error* yang akan diketahui melalui *black box testing* adalah sebagai berikut:

- a. Fungsi yang hilang atau tidak benar.
- b. *Error* dari antar muka.
- c. *Error* dari struktur data atau akses eksternal *database*.
- d. *Error* dari kinerja atau tingkah laku.
- e. *Error* dari inisialisasi dan terminasi.

Test di desain untuk menjawab pertanyaan sebagai berikut:

1. Bagaimana validasi fungsi yang akan ditest ?
2. Bagaimana tingkah laku kinerja dari sistem yang akan ditest ?
3. Kategori masukan apa saja yang bagus digunakan untuk test case ?
4. Apakah sebagian sistem sensitif terhadap suatu nilai masukan tertentu ?
5. Bagaimana batasan suatu kategori masukan ditetapkan ?
6. Sistem mempunyai toleransi jenjang dan volume data apa saja ?