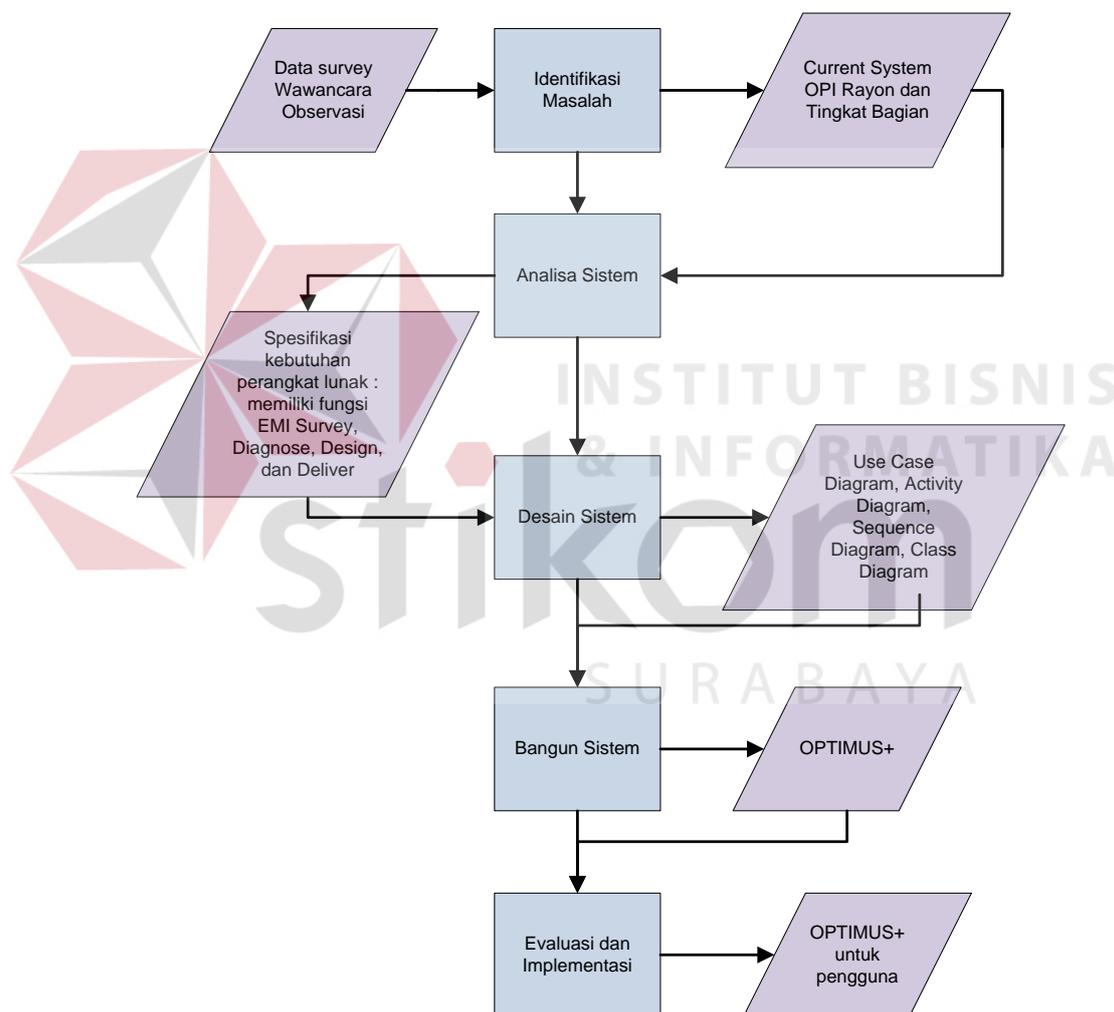


## BAB III

### ANALISIS DAN PERANCANGAN SISTEM

#### 3.1 Analisa Permasalahan

Pada bab ini akan dibahas mengenai analisa dan perancangan sistem dengan menggunakan model pengembangan *waterfall*. Berikut ini adalah gambaran mengenai model *waterfall* dalam penelitian ini:



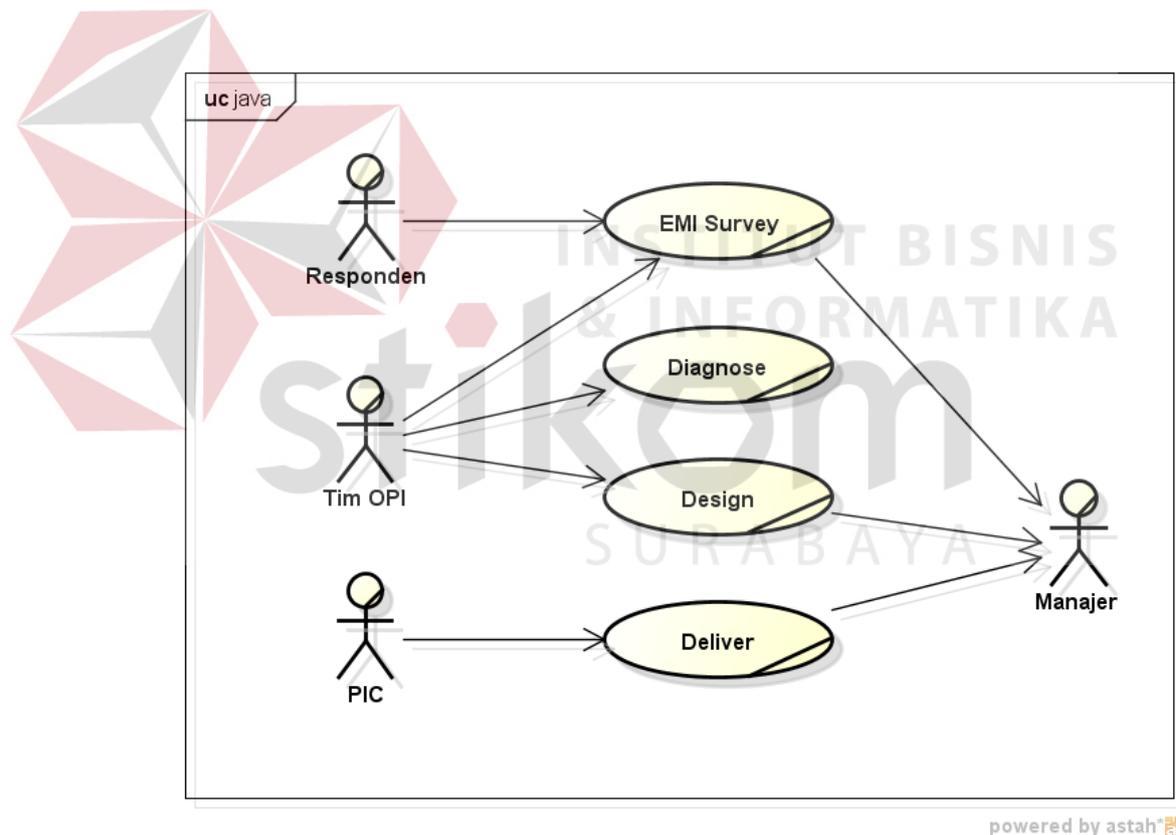
Gambar 3.1 Langkah-Langkah Pengembangan *Waterfall*

Model pengembangan *waterfall* terdiri dari proses identifikasi masalah yang dalam penelitian ini inputnya berupa data *survey*, wawancara, dan observasi

sehingga menghasilkan *output* berupa *current system* OPI Rayon dan Tingkat Bagian, serta permasalahan yang ada di dalamnya. Secara detail, *current system* OPI Rayon dan Tingkat Bagian tersebut dapat digambarkan dalam *Business Use Case Diagram* pada Gambar 3.2.

### 3.1.1 Business Use Case Diagram

Pada Gambar 3.2 terdapat empat proses, yakni: *EMI Survey*, *Diagnose*, *Design* dan *Deliver*. Masing-masing proses tersebut akan dijelaskan dalam *Activity Diagram* pada Gambar 3.3.

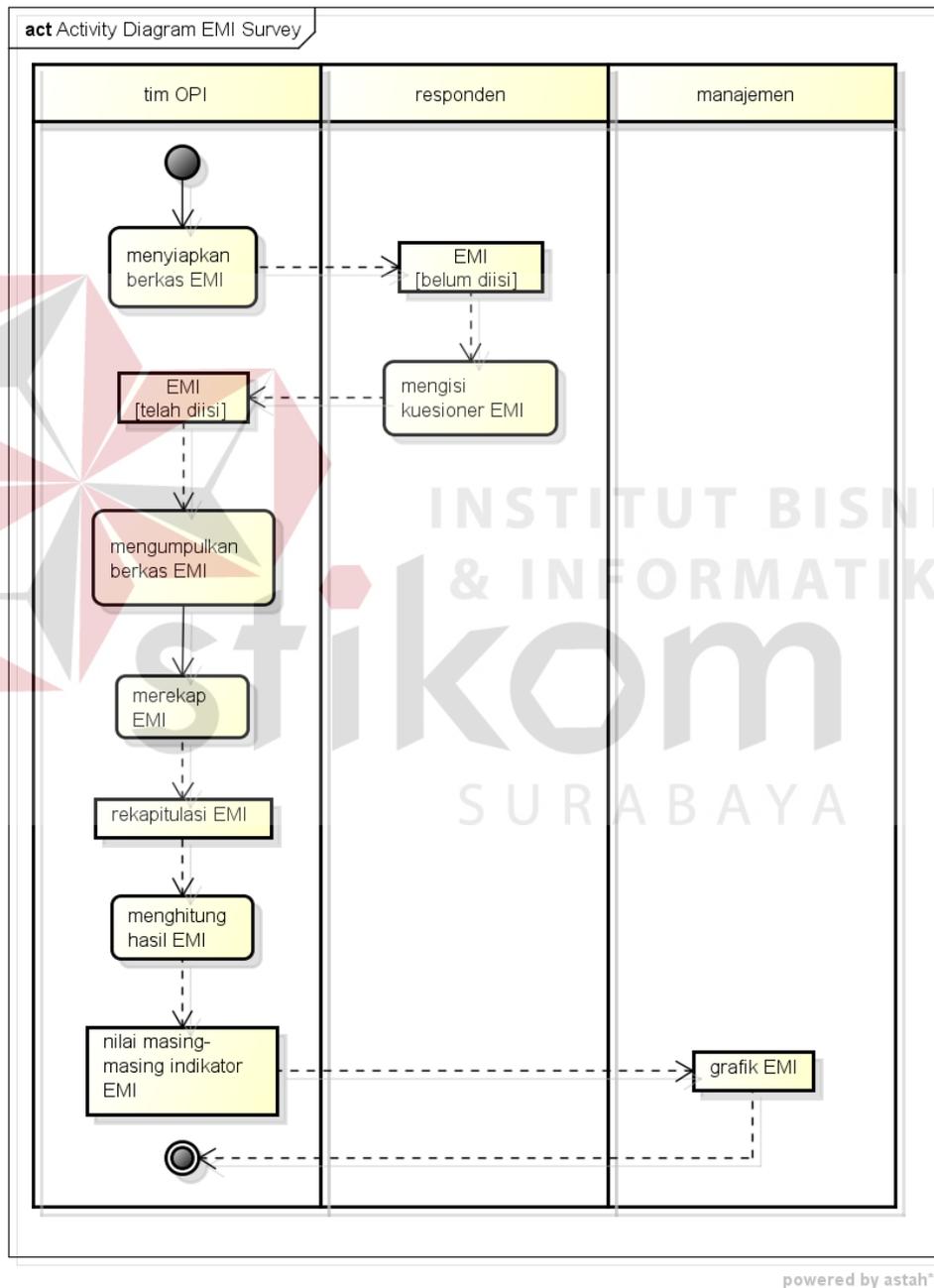


Gambar 3.2 *Business Use Case Diagram OPI*

#### A. *EMI Survey*

*EMI Survey* merupakan analisa yang dilakukan untuk mengetahui tingkah laku dan *mindset* para karyawan (staf *frontliner* dan *supervisor*). Hasil

dari survei ini dapat dijadikan evaluasi dalam menentukan kebijakan pada masa yang akan datang (*Roadmap Operational Performance Improvement (OPI)*, 2012). *EMI Survey* merupakan sebuah survei untuk mengevaluasi kekuatan dan area perbaikan yang terdapat di APJ.



Gambar 3.3 Activity Diagram EMI Survey

Dalam *activity diagram EMI Survey* terdapat permasalahan dalam mendapatkan nilai dari masing-masing indikator *EMI Survey*. *EMI Survey* yang ada saat ini, masih manual, dengan menggunakan kertas. Hal ini dapat menjadi masalah dalam mendapatkan nilai indikator, seandainya data kuesioner hilang atau rusak. Selain itu, dengan kuesioner manual seperti itu, untuk mendapatkan nilai indikator, tim OPI harus melakukan perekapan terlebih dahulu secara manual. Hal ini membutuhkan waktu yang tidak sebentar, sebab data kuesioner tidak sedikit, yaitu berasal dari minimal 50% responden. Dalam perekapan manual seperti ini juga terdapat kemungkinan terjadi kesalahan rekap, sehingga membuat hasil perhitungan menjadi tidak valid.

Formulasi untuk menghitung nilai masing-masing indikator dalam *EMI Survey* adalah sebagai berikut:

$$\text{NilaiIndikator} = \frac{\sum_{i=1}^n N}{X}$$

(Hapsari, 2012)

Keterangan:

N : pertanyaan

l : batas bawah

n : batas atas

X : jumlah pertanyaan yang mewakili indikator

Berdasarkan *Form EMI Survey*, terdapat tujuh indikator dalam *EMI Survey*. Setiap indikator memiliki pertanyaannya masing-masing, seperti berikut ini:

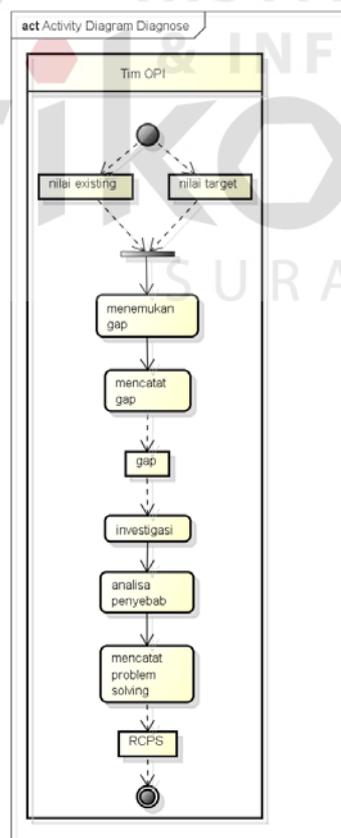
1. Keterbukaan dan transparasi
  - a. Pekerja telah berani mengutarakan pendapatnya secara terbuka, sekalipun berbeda pendapat dengan pimpinanya.
  - b. Kondisi saat ini pimpinan meluangkan cukup waktu untuk berkomunikasi secara terbuka bersama pekerja
  - c. Pekerja telah mengetahui target dan pencapaian kinerja bagiannya
  - d. Kondisi saat ini masing-masing bagian saling mendukung, sehingga tidak terjadi saling menyalahkan
2. Akuntabilitas
  - a. Kondisi saat ini setiap pekerja peduli atas hasil kerja mereka dan dampaknya ke pencapaian kinerja Unit Anda secara keseluruhan
3. Penghargaan (*Rewards*) dan Konsekuensi (*Consequences*)
  - a. SMUK saat ini telah mencerminkan kinerja individu yang sesungguhnya
  - b. Sistem pemberian penghargaan saat ini telah berlangsung secara berkeadilan dan efektif
  - c. Sistem promosi pekerja sudah berlangsung sesuai prosedur, transparan dan efektif
4. Kapabilitas kelas dunia
  - a. Program pelatihan / training sudah memadai dan sesuai dengan kebutuhan pekerjaan
  - b. Sudah ada sistem yang memantau tingkat kompetensi pekerja maupun pimpinan beserta program pengembangannya

- c. Pekerja Unit Anda saat ini telah menjalankan tugas dan tanggung jawab berdasarkan prosedur kerja yang baku
  - d. Saat ini pimpinan selalu siap membagikan pengetahuannya untuk mengembangkan kapabilitas para pekerja
  - e. Saya merasa saya saat ini seluruh fasilitas/peralatan di Unit Anda terpelihara dengan baik
5. Visi transformasi
- a. Pekerja percaya bahwa para pimpinan telah sepakat bahwa Unit Anda harus berubah untuk menjadi lebih baik
  - b. Pekerja telah mengetahui dan memahami visi Unit Anda
6. Mengembangkan para pemimpin masa depan
- a. Pimpinan telah mampu mendidik dan menyiapkan para calon pemimpin berikutnya dengan baik
  - b. Saya merasa saat ini potensi kepemimpinan saya dihargai dan dikembangkan oleh perusahaan
7. Budaya *safety*
- a. Pekerja telah dilengkapi dengan peralatan keselamatan kerja yang memadai untuk pelaksanaan pekerjaannya
  - b. Pekerjaan telah memiliki kapabilitas dan kepedulian yang tinggi terhadap keselamatan dirinya dan pekerja lainnya
  - c. Pekerja saat ini merasakan kondisi yang bersih, aman dan nyaman selama bekerja di Unit Anda

Setelah diperoleh nilai berdasarkan formula yang tertera sebelumnya, nilai atau angka dari indikator tersebut akan disajikan dalam grafik untuk Manajer.

### B. Diagnose

Menurut PT. PLN (Persero) Distribusi Jawa Timur (2012), *diagnose* adalah tahap dimana mencari *bottleneck* yang terjadi pada proses bisnis. *Diagnose* terdiri dari dua langkah, yaitu pembuatan *gap* dan RCPS. *Gap* dilakukan dengan melakukan identifikasi terhadap permasalahan dan menemukan selisih dari nilai target dan nilai yang diperoleh dari topik permasalahan tersebut. RCPS dilakukan dengan menemukan akar dan *problem solving* / ide perbaikan dari suatu permasalahan.

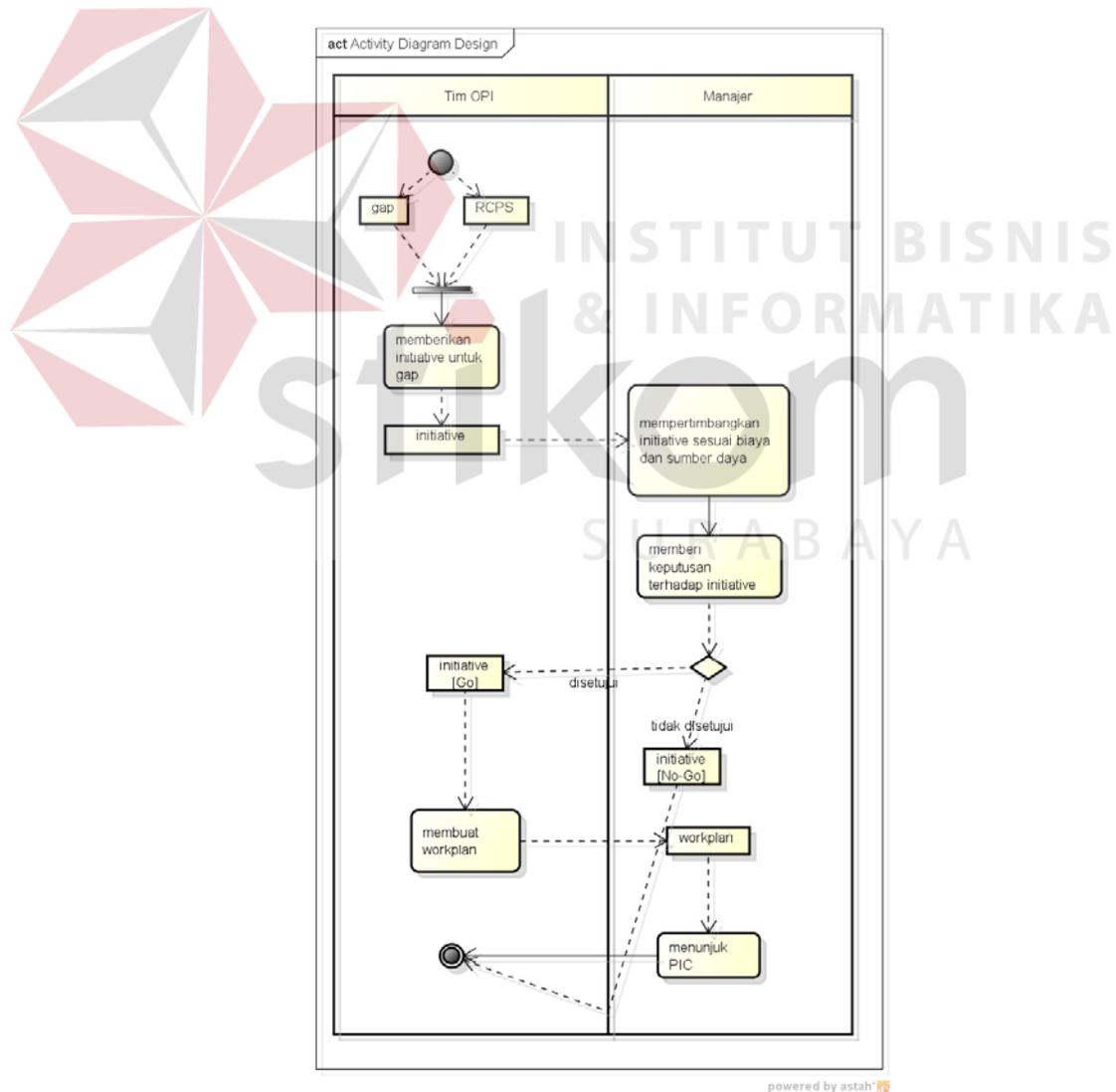


Gambar 3.4 Activity Diagram Diagnose

Gambar 3.4 menjelaskan alur proses *Diagnose*. Dari *activity diagram* *Diagnose* tersebut terdapat dua proses yang akan diikutkan ke dalam sistem yang baru, yaitu: mencatat *gap* dan mencatat *problem solving*.

### C. Design

Menurut PT. PLN (Persero) Distribusi Jawa Timur (2012), *design* adalah tahapan mencari ide-ide perbaikan untuk menutup atau menghilangkan *gap-gap* yang sudah diidentifikasi. *Design* dilakukan dengan membuat *initiative* dan *workplan*.



Gambar 3.5 Activity Diagram Design

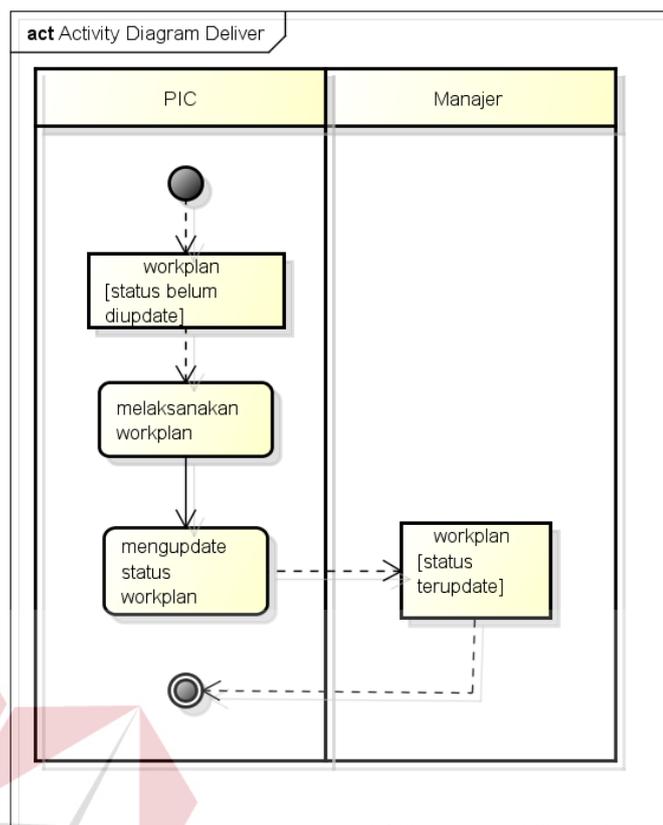
Gambar 3.5 menjelaskan alur proses *Design*. Dari *activity diagram Design* tersebut terdapat beberapa proses yang akan diikuti ke dalam sistem yang baru, yaitu: memberikan *initiative* untuk *gap*, memberi keputusan terhadap setiap *initiative*, dan mengembangkan *initiative* menjadi rangkaian aktivitas (*workplan*).

#### **D. Deliver**

Menurut PT. PLN (Persero) Distribusi Jawa Timur (2012), *deliver* adalah tahap ide-ide perbaikan dijalankan dan dimonitor pelaksanaannya, dampaknya serta hasilnya.

Gambar 3.6 menjelaskan tentang alur proses bisnis *Deliver*. Dalam proses tersebut terdapat permasalahan, yaitu dalam memperbarui status aktivitas *workplan*. Informasi status aktivitas *workplan* ini dibutuhkan oleh banyak pihak, akan tetapi informasi yang disajikan tidak aktual, sehingga bisa mengakibatkan kesalahan informasi. Kesalahan informasi mengakibatkan keterlambatan aktivitas *workplan*.

Selain itu, dalam aktivitas pelaksanaan *workplan* tersebut belum ditunjukkan adanya suatu informasi untuk manajer untuk kebutuhan pemantauan. Manajer Area belum memperoleh informasi yang tepat agar bisa memantau perkembangan *initiative* yang dimilikinya. Hal ini dapat menjadi permasalahan, karena tanpa manajer Area mendapatkan informasi yang tepat, manajer Area selaku pemilik *initiative* tidak dapat mengetahui perkembangan suatu *initiative*.



Gambar 3.6 Activity Diagram Deliver

### 3.2 Analisis Kebutuhan Perangkat Lunak

Dari hasil pengidentifikasian masalah di atas, terdapat beberapa permasalahan, seperti: penggunaan *EMI Survey* masih manual dan berisiko terhadap nilai indikator yang dihasilkan. Selain itu, status aktivitas *workplan* yang statis, tidak dapat memberikan informasi yang aktual. Yang terakhir, belum adanya informasi yang dibutuhkan oleh manajer untuk memantau perkembangan *initiative*. Permasalahan ini nantinya akan digunakan sebagai bahan atau input pada proses selanjutnya dalam model pengembangan *Waterfall*, yaitu analisa sistem.

Analisis sistem mengurai proses dalam *current system* OPI serta permasalahan-permasalahannya menjadi suatu spesifikasi perangkat lunak yang

dibutuhkan oleh masing-masing pengguna. Spesifikasi kebutuhan perangkat lunak tersebut terbagi menjadi empat, antara lain:

1. *EMI Survey*

*EMI Survey* memiliki beberapa fungsi, yaitu: Pengisian *EMI Survey*, Perekapan *EMI Survey*, dan Penghitungan nilai indikator *EMI Survey*.

2. *Diagnose*

*Diagnose* terdiri dari pencatatan *gap* dan RCPS

3. *Design*

*Design* terdiri dari pencatatan *initiative*, pemberian persetujuan oleh manajer terhadap *initiative* yang akan dilanjutkan, dan pembuatan *workplan*.

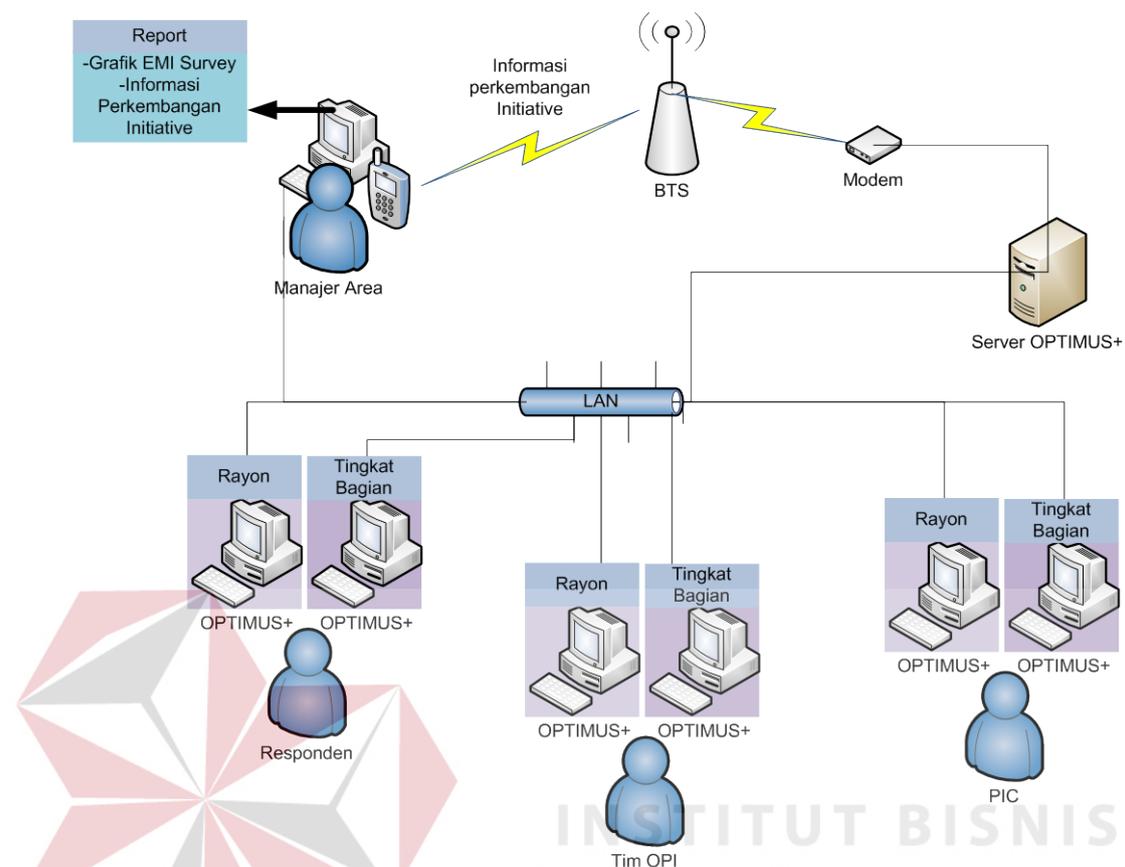
4. *Deliver*

*Deliver* terdiri dari proses memperbarui status dan notifikasi via SMS kepada manajer.

Setelah melakukan proses analisis sistem seperti di atas, dilakukan desain sistem. Tahap desain sistem akan dijelaskan pada sub bab selanjutnya, yaitu perancangan sistem.

### 3.3 Perancangan Sistem

OPTIMUS+ adalah aplikasi yang dibangun agar dapat digunakan PLN Rayon atau Tingkat Bagian Area Surabaya Barat untuk memantau sejauh mana perkembangan kegiatan OPI dilakukan. Berdasarkan identifikasi masalah dan analisa sistem yang sebelumnya telah dilakukan, diperoleh suatu spesifikasi kebutuhan perangkat lunak yang selanjutnya akan dipetakan melalui blok diagram, seperti tampak pada Gambar 3.7.



Gambar 3.7 Blok Diagram OPTIMUS+

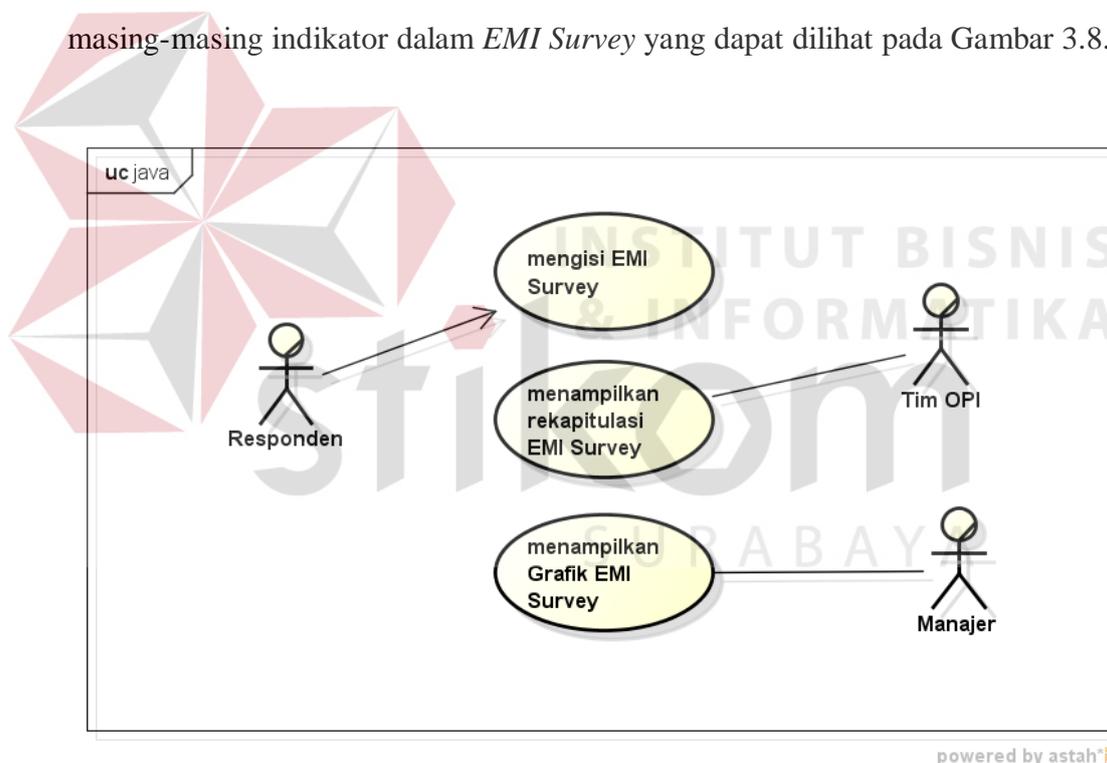
Dalam blok diagram dijelaskan bahwa dalam sistem terdapat empat aktor, yaitu: responden, tim OPI, PIC, dan Manajer Area. OPTIMUS+ akan digunakan di PLN Rayon maupun Tingkat Bagian. Responden berkenaan dengan sistem untuk kepentingan *EMI Survey*. Sedangkan tim OPI untuk kepentingan tanggung jawabnya dalam *gap*, *RCPS*, *Initiative* dan *Workplan*. Ketika suatu *gap* telah memiliki *RCPS*, *initiative*, serta *workplan*, kegiatan berjalan atau dalam *progress*. *Progress* ini ditangani oleh PIC. Sampai pada tahapan ini, OPI sudah mencapai tahap *Deliver*. Dan Manajer Area adalah pihak yang akan memperoleh informasi yang dibutuhkannya untuk kemudian dapat melakukan pemantauan sebagai timbal balik dalam kegiatan OPI.

Secara detail, blok diagram tersebut dapat digambarkan ke dalam *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram*, *Class Diagram*.

### 3.3.1 OPTIMUS+ Use Case Diagram

#### A. EMI Survey

Pada spesifikasi kebutuhan perangkat lunak OPTIMUS+ terdapat beberapa fungsi, salah satunya adalah fungsi *EMI Survey*. Fungsi *EMI Survey* tersebut memiliki beberapa proses, yaitu: mengisi *EMI Survey*, menampilkan rekapitulasi *EMI Survey*, dan menampilkan grafik *EMI Survey* atau nilai dari masing-masing indikator dalam *EMI Survey* yang dapat dilihat pada Gambar 3.8.

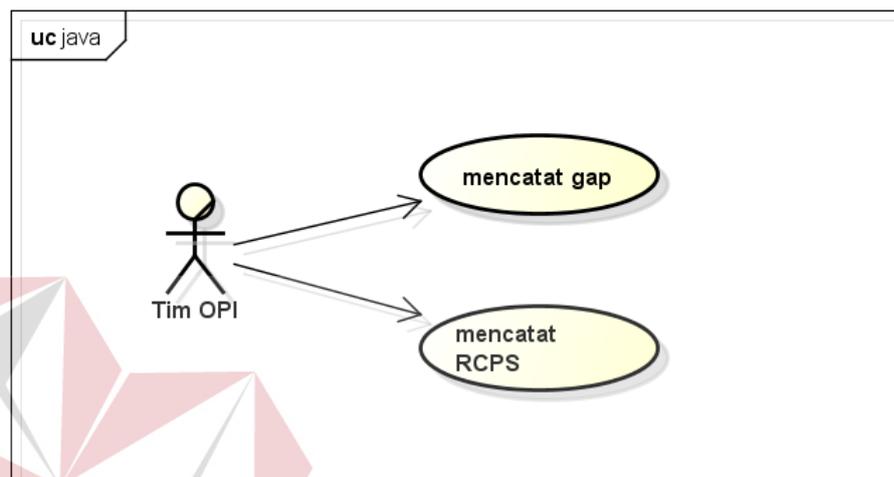


Gambar 3. 8 Use Case Diagram EMI Survey

Mengisi *EMI Survey* dilakukan oleh responden. Jawaban-jawaban dari responden tersebut kemudian masuk ke dalam sistem, kemudian direkap dan dihitung hasilnya oleh sistem. Hasil perekapan dan perhitungan *EMI Survey* diberikan kepada Tim OPI dan Manajer.

## B. Diagnose

Spesifikasi kebutuhan perangkat lunak OPTIMUS+ selanjutnya adalah fungsi *Diagnose*. Fungsi *Diagnose* tersebut memiliki dua proses, yaitu: mencatat *gap* dan RCPS yang ditunjukkan pada Gambar 3.9.

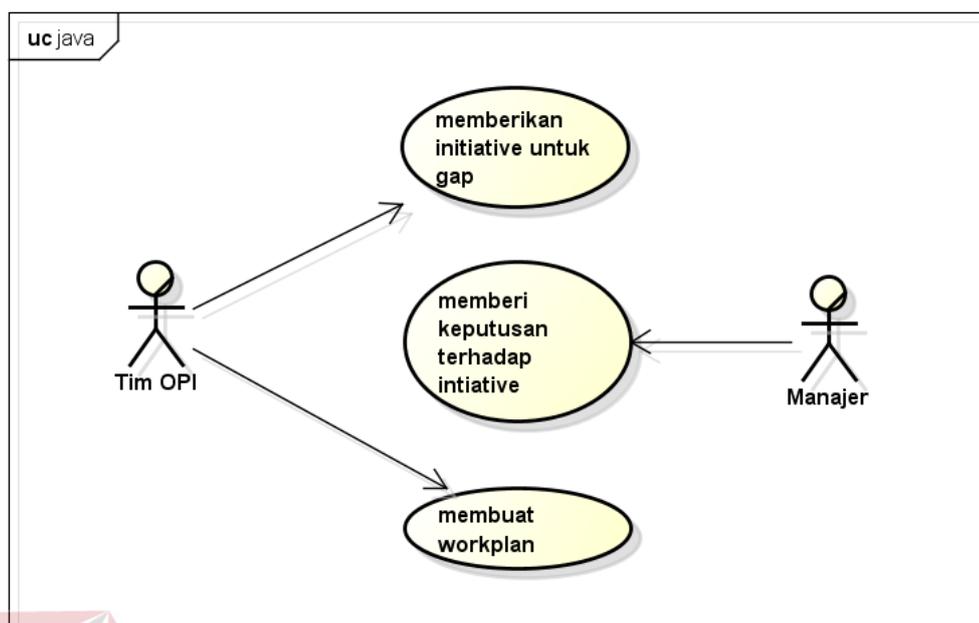


Gambar 3. 9 Use Case Diagram Diagnose

Proses Mencatat *gap* maupun RCPS merupakan tahap *Diagnose* dalam sistem. Dua proses tersebut ditangani oleh Tim OPI sebagai pengganggu jawab tahap *Diagnose*.

## C. Design

Spesifikasi kebutuhan perangkat lunak OPTIMUS+ ketiga adalah fungsi *Design*. Fungsi *Design* tersebut memiliki tiga proses, yaitu: memberikan *initiative* untuk *gap*, memberi keputusan terhadap *initiative* , dan membuat *workplan* atau rangkaian aktivitas. Proses-proses tersebut dapat dilihat pada Gambar 3.10.



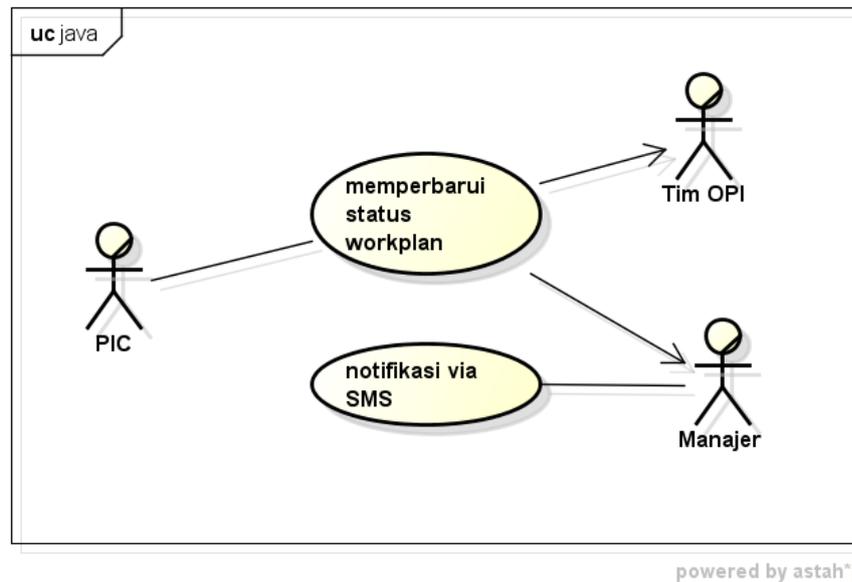
powered by astah

Gambar 3. 10 Use Case Diagram Design

Setelah melakukan pencatatan *gap* dan RCPS, Tim OPI memberikan *initiative* terhadap *gap* tersebut. *Initiative* merupakan ide perbaikan untuk *gap*. Setelah Tim OPI memasukkan *initiative*, Manajer dapat memberikan keputusan, apakah *initiative* tersebut akan dilanjutkan atau tidak. Jika *initiative* tersebut disetujui, maka Tim OPI akan membuat rangkaian aktivitas nyata untuk *initiative* tersebut.

#### D. Deliver

Spesifikasi kebutuhan perangkat lunak OPTIMUS+ terakhir adalah fungsi *Deliver*. Fungsi *Deliver* tersebut memiliki dua proses, yaitu: memperbarui status *workplan* dan melakukan notifikasi via SMS. Dua proses tersebut dapat dilihat pada Gambar 3.11.



Gambar 3. 11 Use Case Diagram Deliver

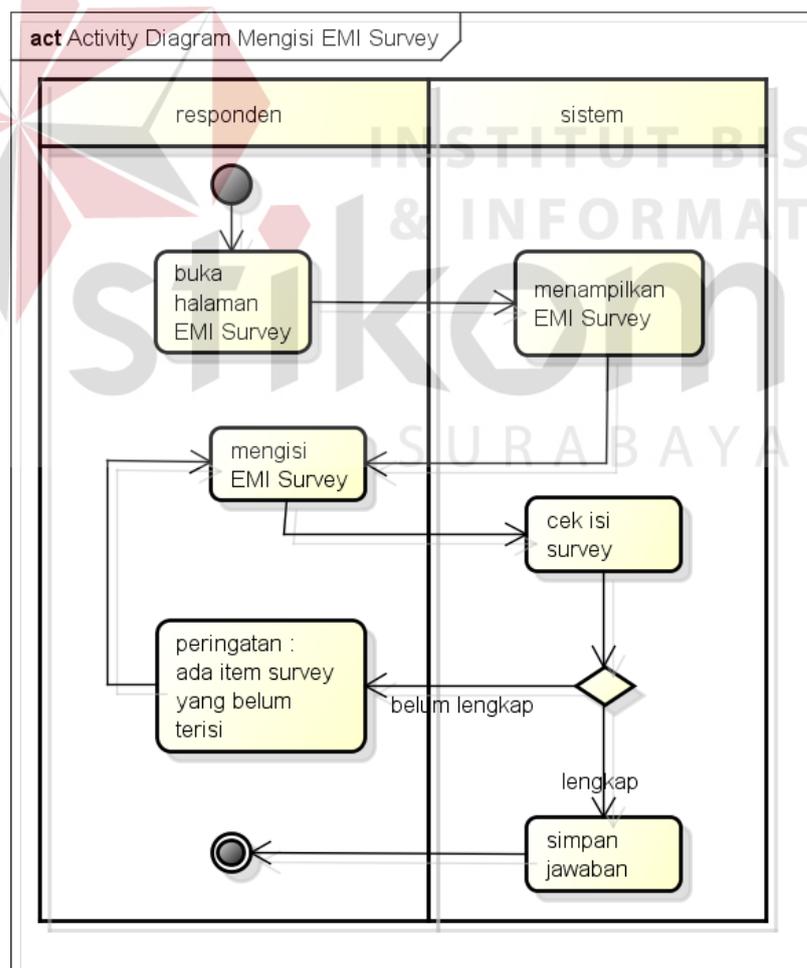
Dalam sistem, *workplan* memiliki status-status. Status tersebut antara lain, “Belum Dimulai”, “Dalam Proses”, “Selesai”, “Ditunda”, dan “Melebihi *Deadline*”. Sistem akan menampilkan setiap aktivitas *workplan* dengan statusnya masing-masing. Sistem dapat mengubah status aktivitas *workplan* secara otomatis sesuai perubahan tanggal menjadi “Belum Dimulai”, “Dalam Proses”, dan “Melebihi *Deadline*”. Selain itu, sistem dapat merespon aksi yang dilakukan PIC, sehingga status “Belum Dimulai”, “Dalam Proses”, dan “Melebihi *Deadline*” bisa berubah menjadi “Selesai” atau “Ditunda”.

### 3.3.2 OPTIMUS+ Activity Diagram

Pada tahap pemodelan sistem, diagram aktivitas dapat digunakan untuk menjelaskan aktivitas yang terjadi di dalam sebuah *use case*. Berikut ini adalah *activity diagram* dari masing-masing *use case* yang telah digambarkan di atas:

### A. Mengisi *EMI Survey*

Diawali dengan halaman web menampilkan *EMI Survey*, kemudian responden melakukan pengisian *EMI Survey* di halaman tersebut. Setelah itu, ketika responden mengklik button Simpan, maka sistem akan memeriksa, apakah semua item dalam halaman survei telah diisi atau tidak. Jika semua item telah terisi, maka sistem akan melanjutkan penyimpanan isian survey ke dalam database. Jika tidak, maka sistem akan memberikan peringatan agar responden melengkapi isiannya terlebih dahulu, kemudian menyimpannya lagi. Proses Mengisi *EMI Survey* dapat dilihat pada Gambar 3.12.

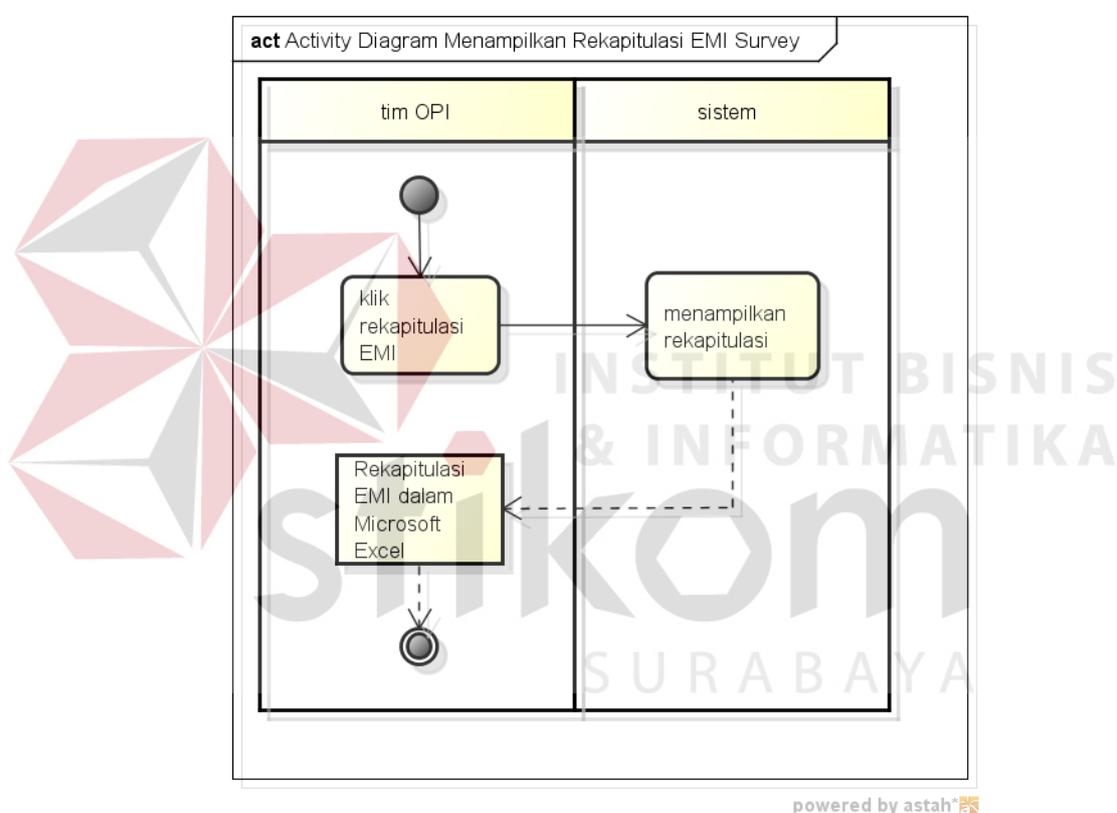


powered by astah\*

Gambar 3. 12 Activity Diagram Mengisi *EMI Survey*

## B. Menampilkan Rekapitulasi *EMI Survey*

Perekapan *EMI Survey* dilakukan dengan cara mengklik button rekapitulasi yang telah disediakan di dalam halaman *report*, kemudian sistem akan memuat data *EMI Survey* yang diperlukan dari *database*. Pemuatan data tersebut ditampilkan dalam bentuk Microsoft Excel. Proses menampilkan rekapitulasi *EMI Survey* dapat dilihat pada Gambar 3.13.

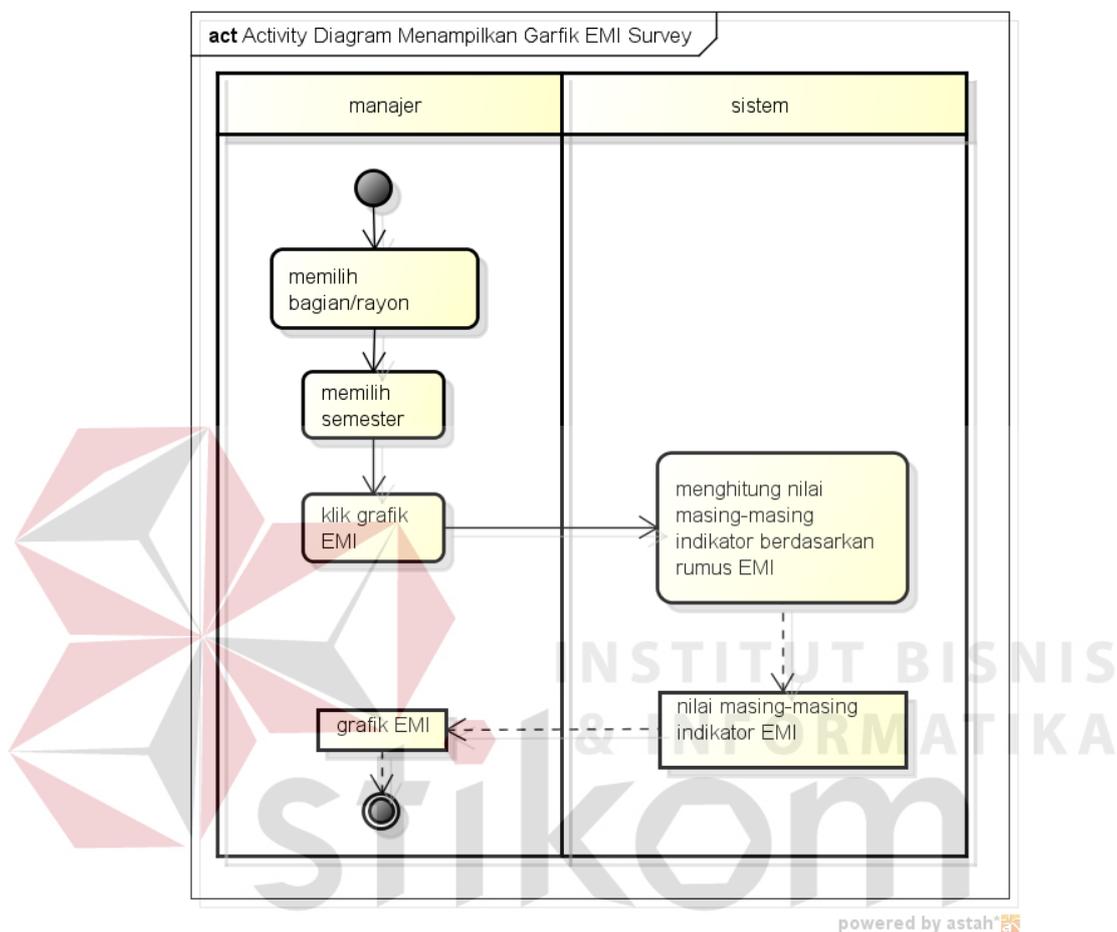


Gambar 3. 13 Activity Diagram Menampilkan Rekapitulasi *EMI Survey*

## C. Menampilkan Grafik *EMI Survey*

Menampilkan *EMI Survey* dilakukan dengan cara mengklik button grafik yang telah disediakan di dalam halaman *report*, kemudian sistem akan memuat data *EMI Survey* yang diperlukan dari *database* berupa angka-angka jawaban dari responden. Angka-angka tersebut diolah menjadi nilai dari masing-masing

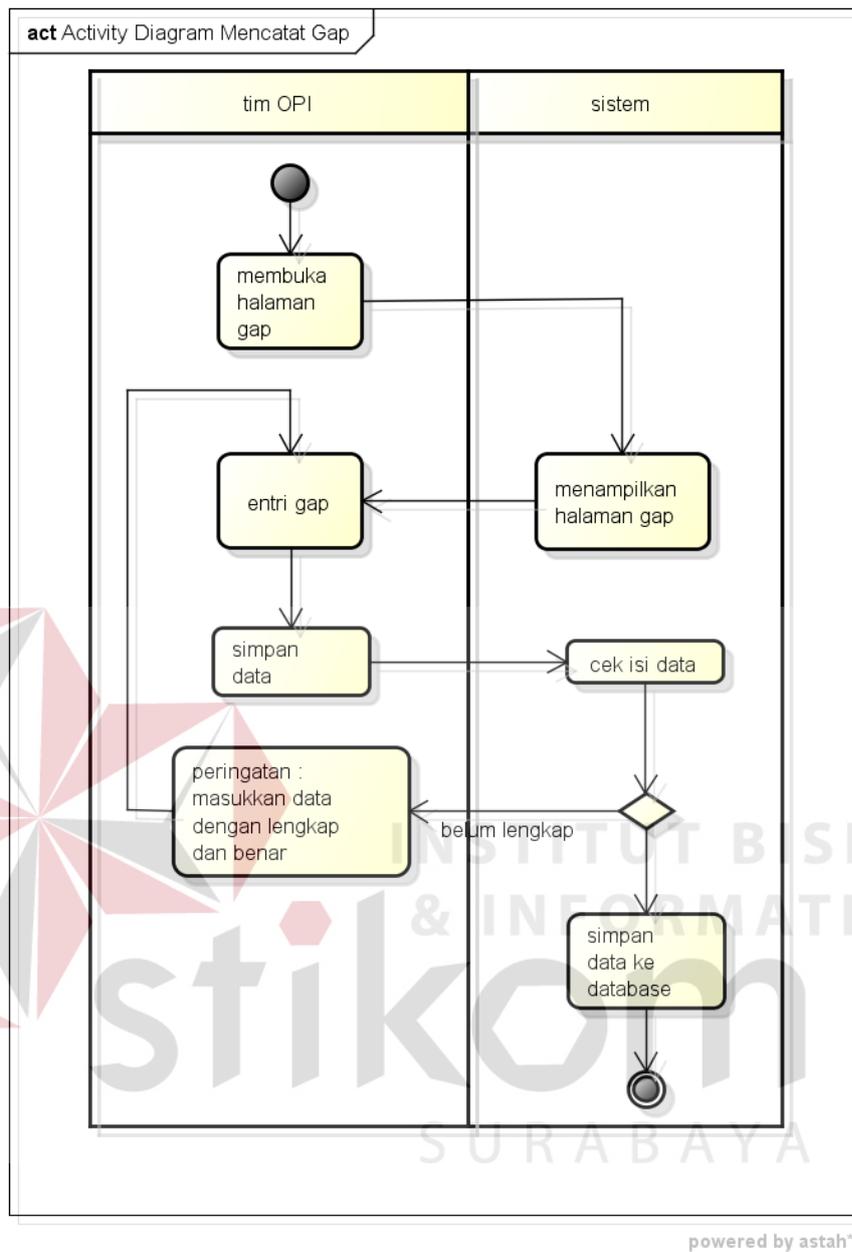
indikator *EMI Survey*. Proses menampilkan grafik *EMI Survey* dapat dilihat pada Gambar 3.14.



Gambar 3. 14 Activity Diagram Menampilkan Grafik *EMI Survey*

#### D. Mencatat *Gap*

Proses pencatatan *gap* ini yaitu: web menampilkan halaman *gap*. Kemudian tim OPI memasukkan data-data *gap* dan menyimpannya. Sistem akan menolak penyimpanan, jika data *gap* yang diisikan belum lengkap. Selain itu, sistem juga akan memberikan peringatan agar tim OPI melengkapi data *gap*-nya. Jika data telah lengkap, maka data akan disimpan ke dalam *database*. Proses Mencatat *gap* dapat dilihat pada Gambar 3.15.



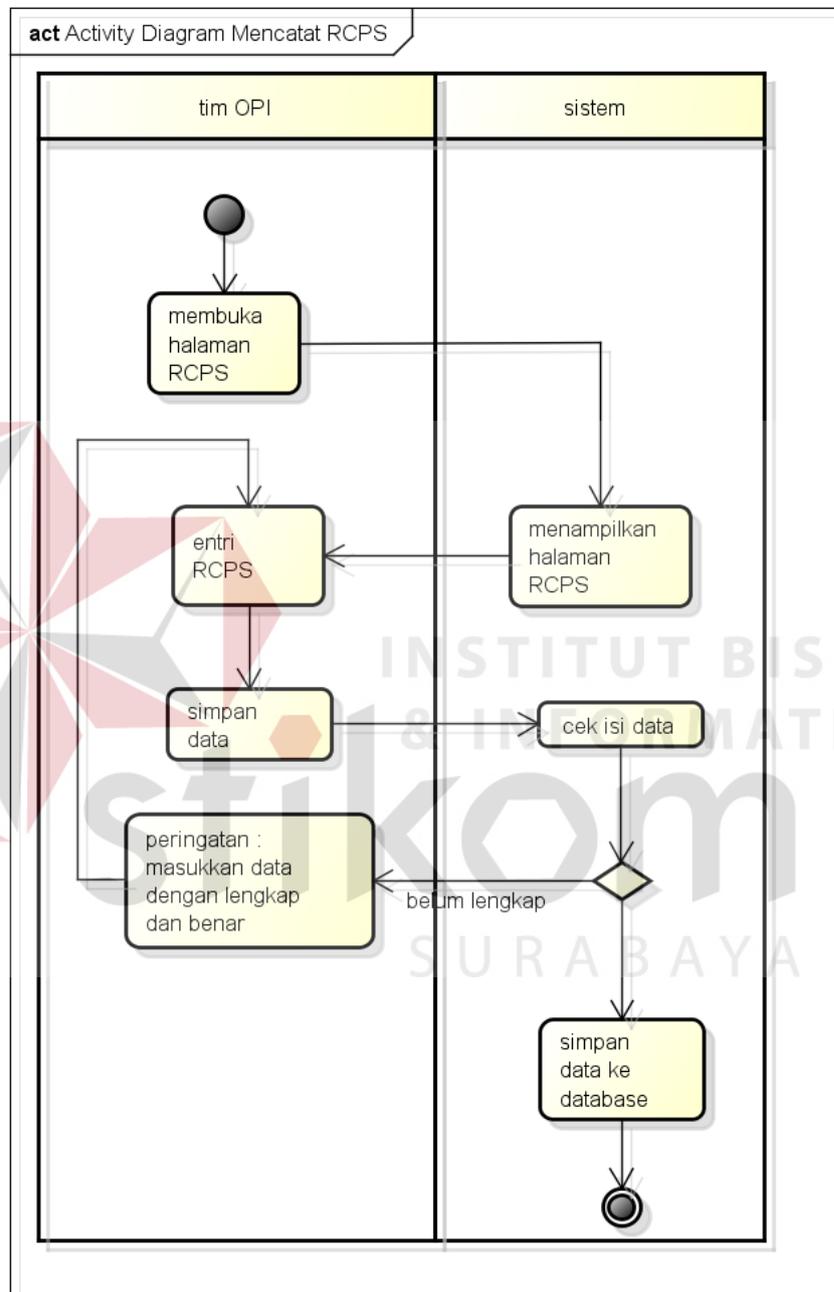
Gambar 3. 15 Activity Diagram Mencatat Gap

### E. Mencatat RCPS

Proses pencatatan RCPS ini yaitu: web menampilkan halaman RCPS. Kemudian Tim OPI memasukkan data-data RCPS dan menyimpannya. Sistem akan menolak penyimpanan, jika data RCPS yang diisikan belum lengkap. Selain itu, sistem juga akan memberikan peringatan agar tim OPI melengkapi data

RCPSnya. Jika data telah lengkap, maka data akan disimpan ke dalam *database*.

Proses Mencatat RCPS dapat dilihat pada Gambar 3.16.

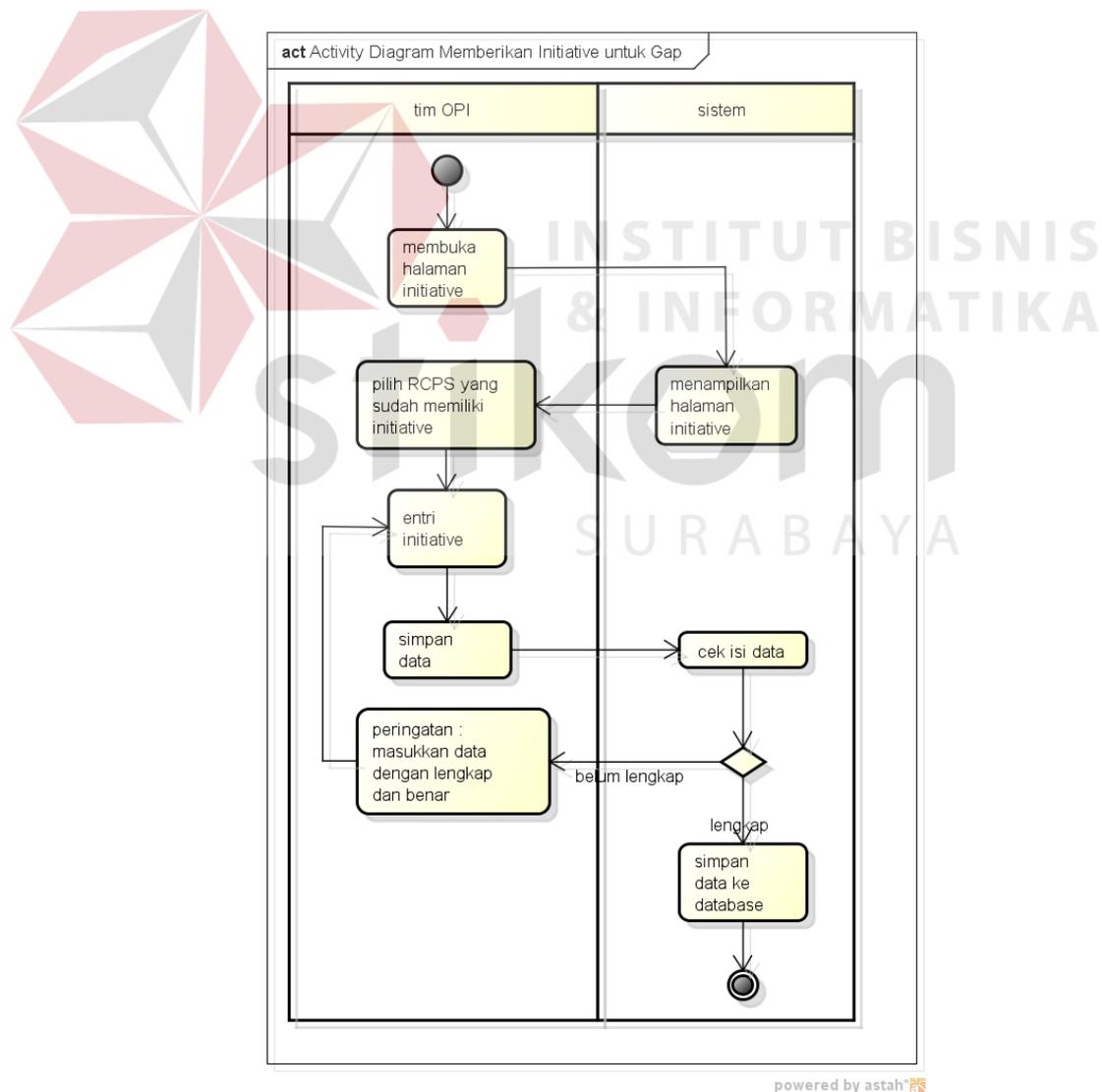


powered by astah\*

Gambar 3. 16 Activity Diagram Mencatat RCPS

## F. Memberikan *Initiative* untuk *Gap*

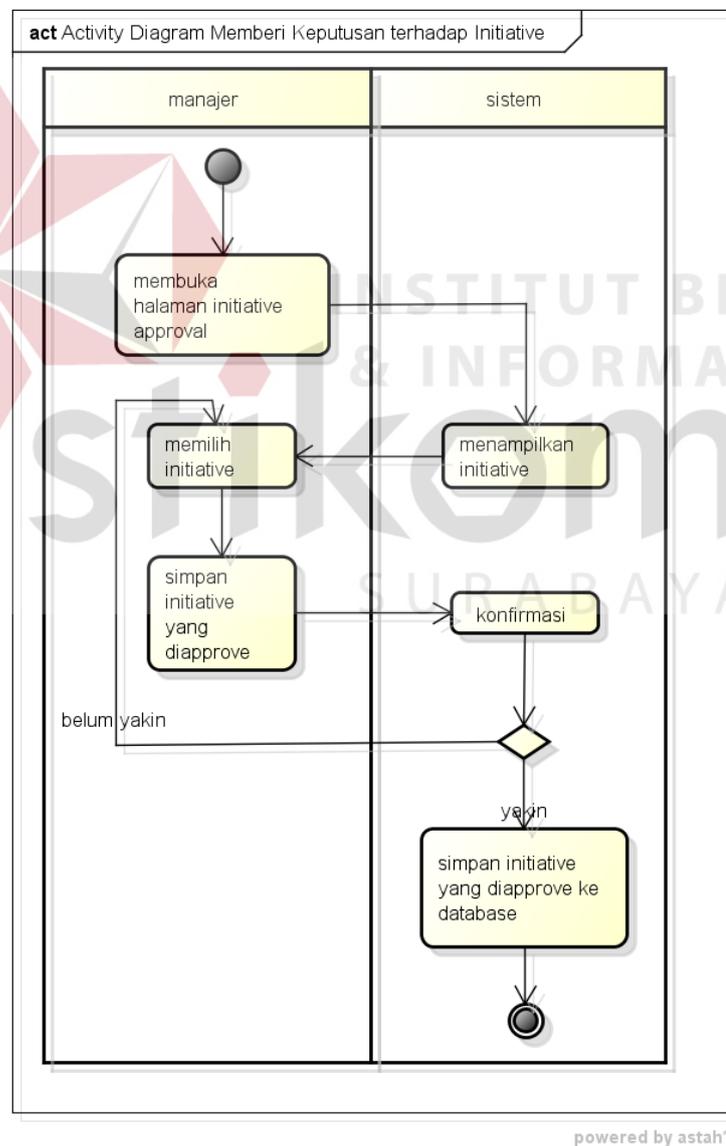
Proses pencatatan *initiative* ini yaitu: web menampilkan halaman *initiative*. Kemudian tim OPI memasukkan data-data *initiative* dan menyimpannya. Sistem akan menolak penyimpanan, jika data *initiative* yang diisikan belum lengkap. Selain itu, sistem juga akan memberikan peringatan agar tim OPI melengkapi data *initiative* nya. Jika data telah lengkap, maka data akan disimpan ke dalam *database*. Proses Memberikan *Initiative* untuk *gap* dapat dilihat pada Gambar 3.17.



Gambar 3. 17 Activity Diagram Memberikan Initiative untuk Gap

### G. Memberi Keputusan terhadap setiap *Initiative*

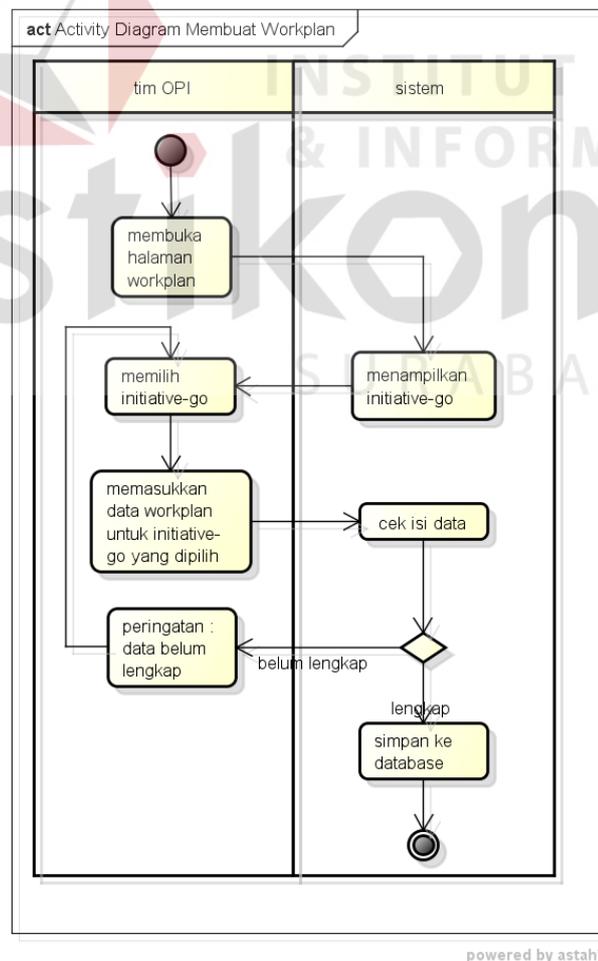
Setiap *initiative* adalah milik dari manajer. Untuk itu pemberian keputusan oleh manajer terhadap setiap *initiative* menjadi penting. Pada mulanya, pada proses manualnya, pemberian keputusan ini dilakukan saat rapat. Dengan OPTIMUS+, pemberian keputusan dilakukan dengan tujuan sebagai *filter* bagi *initiative-initiative* yang akan ditindak-lanjuti. Proses Memberikan Keputusan terhadap setiap *Initiative* dapat dilihat pada Gambar 3.18.



Gambar 3. 18 *Activity Diagram* Memberi Keputusan terhadap setiap *Initiative*

## H. Membuat *Workplan*

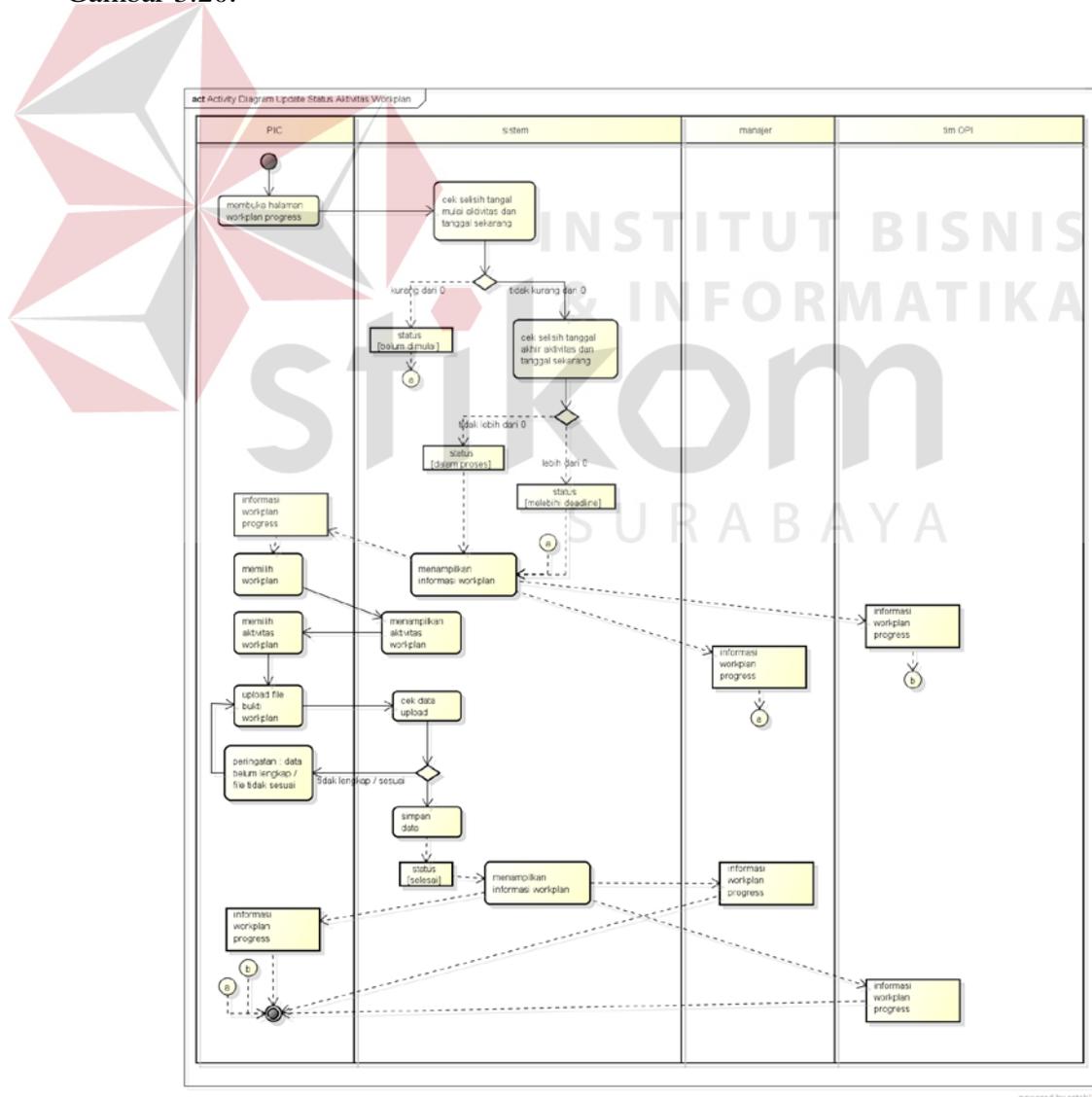
*Initiative -Go* atau *initiative* yang telah disetujui oleh manajer, harus ditindak-lanjuti oleh tim OPI dengan cara membuat *workplan*. Di dalam *workplan* tersebut terdapat dua hal penting, yaitu tanggal mulai dan tanggal akhir aktivitas. Dua tanggal itu masuk ke dalam daftar isian *workplan*. Setelah tim OPI selesai mengisi data *workplan* dengan lengkap, menyimpan, maka sistem akan melakukan penyimpanan ke dalam *database*. Jika tidak, maka sistem akan memberikan peringatan agar tim OPI melengkapi data isian terlebih dahulu, kemudian coba menyimpannya lagi. Proses Membuat *Workplan* dapat dilihat pada Gambar 3.19.



Gambar 3. 19 Activity Diagram Membuat *Workplan*

## I. Memperbarui Status Aktivitas *Workplan*

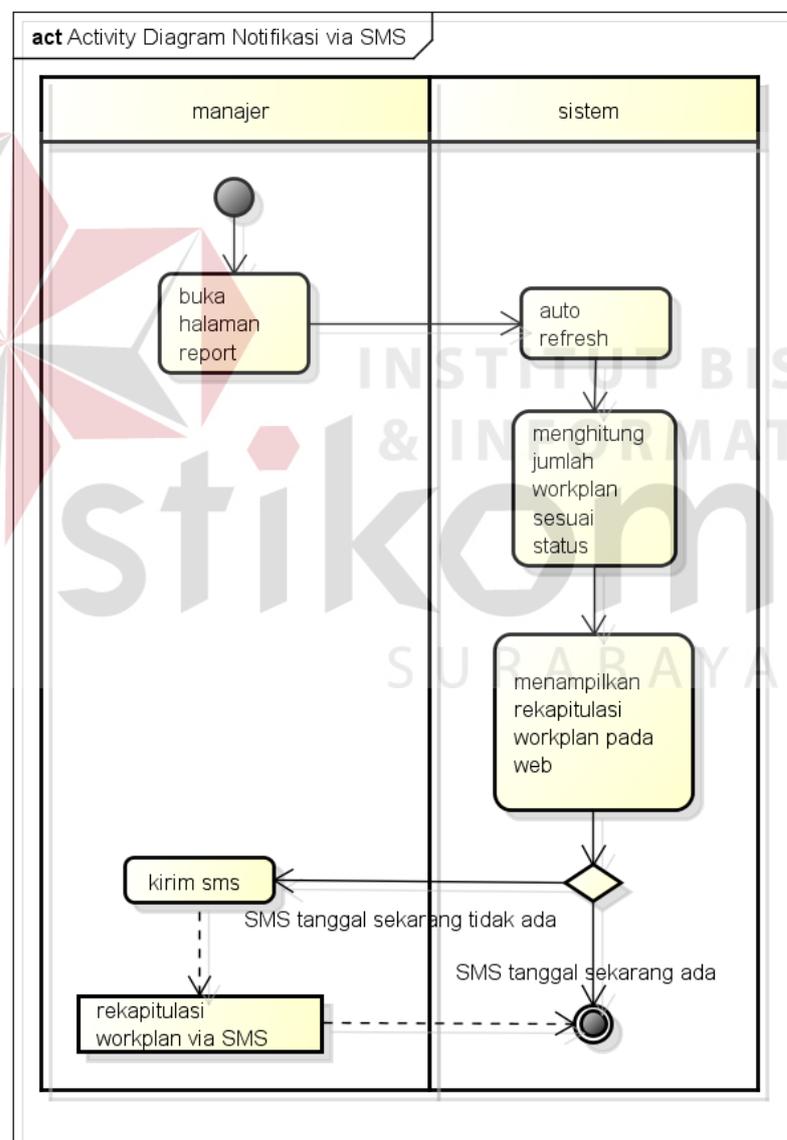
Status diperbarui karena perubahan tanggal dan/atau PIC memasukkan *progress* aktivitas *workplan*. Sistem akan memeriksa selisih tanggal sekarang dengan tanggal mulai/akhir aktivitas *workplan*. Selisih tanggal sekarang dengan tanggal mulai/akhir aktivitas *workplan* dapat berupa status “Belum Dimulai”, “Dalam Proses”, atau “Melebihi *Deadline*”. Sedangkan proses memasukkan *progress workplan* dapat mengubah status aktivitas menjadi “Selesai” atau “Ditunda”. Proses Memperbarui Status Aktivitas *Workplan* dapat dilihat pada Gambar 3.20.



Gambar 3. 20 Activity Diagram Memperbarui Status Aktivitas *Workplan*

## J. Notifikasi via SMS

Notifikasi SMS ini diawali dengan penghitungan jumlah aktivitas sesuai statusnya. Hasilnya berupa rangkuman yang nantinya akan dikirim dengan menggunakan SMS kepada manajer selaku pemantau perkembangan *initiative* sebelum tanggal akhir aktivitas berakhir. Proses Notifikasi via SMS dapat dilihat pada Gambar 3.21.



Gambar 3. 21 Activity Diagram Notifikasi via SMS

### 3.3.3 Flow of Event

Detail spesifikasi *Use Case* ditulis dalam *Flow of Event*. Tujuan utama *Flow of Event* adalah untuk mendokumentasikan aliran logika dalam *Use Case* dan menjelaskan secara rinci apa yang pemakai akan lakukan serta apa yang sistem itu sendiri lakukan. Sistematika *Flow of Event* terdiri dari beberapa elemen berikut:

1. Deskripsi singkat
2. Prasyarat
3. Alur utama
4. Alur alternatif dan alur salah
5. Kondisi akhir

#### A. Flow of Event Use Case Mengisi EMI Survey

*Flow of Event* pada Tabel 3.1 menjelaskan tentang alur logika pada *Use Case* Mengisi EMI Survey. Dari *flow of Event* tersebut dapat diketahui deskripsi singkat, prasyarat, alur utama, alur alternatif dan alur salah, serta kondisi akhir dari *Use Case* Mengisi EMI Survey.

Tabel 3.1 *Flow of Event Use Case* Mengisi EMI Survey

Deskripsi Use Case	
<b>Nama Use Case</b>	Mengisi EMI Survey
<b>Deskripsi Singkat</b>	Mengisi EMI Survey oleh responden dilakukan untuk memperoleh <i>Diagnose</i> awal
<b>Aktor</b>	Responden
<b>Prasyarat (kondisi)</b>	Tidak ada
<b>Alur Utama</b>	<ol style="list-style-type: none"> <li>1. Pengguna mengklik link ke EMI Survey</li> <li>2. Sistem menampilkan daftar pertanyaan EMI Survey</li> <li>3. Pengguna mengisi jawaban atas setiap pertanyaan EMI Survey</li> <li>4. Pengguna menekan tombol “Simpan” untuk</li> </ol>

<b>Deskripsi Use Case</b>	
	<p>meyimpan jawaban ke dalam <i>database</i></p> <ol style="list-style-type: none"> <li>5. Sistem mengkonfirmasi apakah pengguna telah yakin dengan jawabannya</li> <li>6. Pengguna menekan tombol “Ya”</li> <li>7. Pengguna dapat meninggalkan halaman <i>EMI Survey</i></li> </ol>
<b>Alur Alternatif</b>	<ol style="list-style-type: none"> <li>3. 1. Jika pengguna tidak lengkap mengisi jawaban, dan pengguna menekan tombol “Simpan”, maka sistem akan menampilkan pesan dialog bahwa pengguna harus melengkapi jawabannya terlebih dahulu</li> <li>6. 1. Jika pengguna menekan tombol “Tidak”, maka data belum akan tersimpan dalam <i>database</i>. pengguna juga masih dapat memperbaiki jawabannya.</li> </ol>
<b>Kondisi Akhir Sukses</b>	Jawaban kuesioner tersimpan dalam <i>database</i>
<b>Kondisi Akhir Gagal</b>	Jawaban kuesioner tidak tersimpan dalam <i>database</i>

#### B. *Flow of Event Use Case Menampilkan Rekapitulasi EMI Survey*

*Flow of Event* pada Tabel 3.2 menjelaskan tentang alur logika pada *Use Case Menampilkan Rekapitulasi EMI Survey*. Dari *flow of Event* tersebut dapat diketahui deskripsi singkat, prasyarat, alur utama, alur alternatif dan alur salah, serta kondisi akhir dari *Use Case Menampilkan Rekapitulasi EMI Survey*.

Tabel 3.2 *Flow of Event Use Case Menampilkan Rekapitulasi EMI Survey*

<b>Deskripsi Use Case</b>	
<b>Nama Use Case</b>	Menampilkan Rekapitulasi <i>EMI Survey</i>
<b>Deskripsi Singkat</b>	Menampilkan Rekapitulasi <i>EMI Survey</i> oleh sistem dilakukan untuk memperoleh rekapitulasi yang dibutuhkan Tim OPI dan Manajer
<b>Aktor</b>	Sistem
<b>Prasyarat (kondisi)</b>	Tidak ada
<b>Alur Utama</b>	<ol style="list-style-type: none"> <li>1. Sistem memperoleh jawaban dari responden, lalu menyimpannya ke dalam <i>database</i></li> <li>2. Pengguna membuka laporan “Rekapitulasi <i>EMI Survey</i>”</li> <li>3. Sistem mengunduh rekapitulasi yang berisi jawaban-jawaban <i>EMI Survey</i></li> <li>4. Sistem menampilkan rekapitulasi <i>EMI Survey</i></li> </ol>

<b>Deskripsi Use Case</b>	
	dalam berkas Microsoft Excel.
<b>Alur Alternatif</b>	Tidak ada
<b>Kondisi Akhir Sukses</b>	Sistem mengunduh dan menampilkan Rekapitulasi <i>EMI Survey</i>
<b>Kondisi Akhir Gagal</b>	Sistem tidak dapat mengunduh dan menampilkan Rekapitulasi <i>EMI Survey</i>

### C. *Flow of Event Use Case Menampilkan Grafik EMI Survey*

*Flow of Event* pada Tabel 3.3 menjelaskan tentang alur logika pada *Use Case Menampilkan Grafik EMI Survey*. Dari *flow of Event* tersebut dapat diketahui deskripsi singkat, prasyarat, alur utama, alur alternatif dan alur salah, serta kondisi akhir dari *Use Case Menampilkan Grafik EMI Survey*.

Tabel 3.3 *Flow of Event Use Case Menampilkan Grafik EMI Survey*

<b>Deskripsi Use Case</b>	
<b>Nama Use Case</b>	Menampilkan Grafik <i>EMI Survey</i>
<b>Deskripsi Singkat</b>	Menampilkan Grafik <i>EMI Survey</i> dilakukan untuk memperoleh masing-masing nilai indikator, kemudian menampilkannya dalam bentuk grafik <i>EMI Survey</i>
<b>Aktor</b>	Sistem
<b>Prasyarat (kondisi)</b>	Tidak ada
<b>Alur Utama</b>	<ol style="list-style-type: none"> <li>1. Sistem memperoleh jawaban yang dimasukkan responden</li> <li>2. Sistem menghitung nilai masing-masing indikator yang berasal dari jawaban yang dimasukkan responden-responden dengan rumus masing-masing indikator</li> <li>3. Pengguna membuka grafik <i>EMI Survey</i></li> <li>4. Sistem menampilkan grafik <i>EMI Survey</i></li> </ol>
<b>Alur Alternatif</b>	Tidak ada
<b>Kondisi Akhir Sukses</b>	Sistem menampilkan grafik <i>EMI Survey</i>
<b>Kondisi Akhir Gagal</b>	Sistem tidak dapat menampilkan grafik <i>EMI Survey</i>

### D. *Flow of Event Use Case Mencatat Gap*

*Flow of Event* pada Tabel 3.4 menjelaskan tentang alur logika pada *Use Case Mencatat Gap*. Dari *flow of Event* tersebut dapat diketahui deskripsi singkat,

prasyarat, alur utama, alur alternatif dan alur salah, serta kondisi akhir dari *Use Case* Mencatat *Gap*.

Tabel 3.4 *Flow of Event Use Case* Mencatat *Gap*

<b>Deskripsi Use Case</b>	
<b>Nama Use Case</b>	Mencatat <i>Gap</i>
<b>Deskripsi Singkat</b>	Mencatat <i>gap</i> dilakukan untuk kebutuhan diagnosa awal
<b>Aktor</b>	Tim OPI
<b>Prasyarat (kondisi)</b>	Tidak ada
<b>Alur Utama</b>	<ol style="list-style-type: none"> <li>1. Pengguna membuka halaman <i>gap</i></li> <li>2. Sistem menampilkan halaman entri <i>gap</i></li> <li>3. Pengguna memasukkan data <i>gap</i></li> <li>4. Sistem akan mengecek data <i>gap</i> yang dimasukkan, apakah telah lengkap atau belum</li> <li>5. Pengguna menekan tombol “Simpan”</li> </ol>
<b>Alur Alternatif</b>	4.1 Jika data yang dimasukkan belum lengkap, dan pengguna menekan tombol “Simpan”, maka akan keluar pesan peringatan bahwa data belum lengkap
<b>Kondisi Akhir Sukses</b>	Data <i>gap</i> tersimpan ke dalam <i>database</i>
<b>Kondisi Akhir Gagal</b>	Data <i>gap</i> tidak tersimpan ke dalam <i>database</i>

#### E. *Flow of Event Use Case* Mencatat RCPS

*Flow of Event* pada Tabel 3.5 menjelaskan tentang alur logika pada *Use Case* Mencatat RCPS. Dari *flow of Event* tersebut dapat diketahui deskripsi singkat, prasyarat, alur utama, alur alternatif dan alur salah, serta kondisi akhir dari *Use Case* Mencatat RCPS.

Tabel 3.5 *Flow of Event Use Case* Mencatat RCPS

<b>Deskripsi Use Case</b>	
<b>Nama Use Case</b>	Mencatat RCPS
<b>Deskripsi Singkat</b>	Mencatat RCPS dilakukan untuk kebutuhan diagnosa awal
<b>Aktor</b>	Tim OPI
<b>Prasyarat (kondisi)</b>	Tidak ada
<b>Alur Utama</b>	<ol style="list-style-type: none"> <li>1. Pengguna membuka halaman entri RCPS</li> </ol>

<b>Deskripsi Use Case</b>	
	2. Sistem menampilkan halaman entri RCPS 3. Pengguna memasukkan data RCPS 4. Sistem akan mengecek data RCPS yang dimasukkan, apakah telah lengkap atau belum 5. Pengguna menekan tombol “Simpan”
<b>Alur Alternatif</b>	4.1 Jika data yang dimasukkan belum lengkap, dan pengguna menekan tombol “Simpan”, maka akan keluar pesan peringatan bahwa data belum lengkap
<b>Kondisi Akhir Sukses</b>	Data RCPS tersimpan ke dalam <i>database</i>
<b>Kondisi Akhir Gagal</b>	Data RCPS tidak tersimpan ke dalam <i>database</i>

#### F. *Flow of Event Use Case Membuat Initiative untuk Gap*

*Flow of Event* pada Tabel 3.6 menjelaskan tentang alur logika pada *Use Case Membuat Initiative untuk Gap*. Dari *flow of Event* tersebut dapat diketahui deskripsi singkat, prasyarat, alur utama, alur alternatif dan alur salah, serta kondisi akhir dari *Use Case Membuat Initiative untuk Gap*.

Tabel 3.6 *Flow of Event Use Case Membuat Initiative untuk Gap*

<b>Deskripsi Use Case</b>	
<b>Nama Use Case</b>	Membuat <i>Initiative</i> untuk <i>Gap</i>
<b>Deskripsi Singkat</b>	Membuat <i>initiative</i> digunakan kebutuhan <i>Design</i>
<b>Aktor</b>	Tim OPI
<b>Prasyarat (kondisi)</b>	Tidak ada
<b>Alur Utama</b>	1. Pengguna membuka halaman <i>initiative</i> 2. Sistem menampilkan halaman entri <i>initiative</i> 3. Pengguna memasukkan data <i>initiative</i> 4. Sistem akan mengecek data <i>initiative</i> yang dimasukkan, apakah telah lengkap atau belum 5. Pengguna menekan tombol “Simpan”
<b>Alur Alternatif</b>	4.1 Jika data yang dimasukkan belum lengkap, dan pengguna menekan tombol “Simpan”, maka akan keluar pesan peringatan bahwa data belum lengkap
<b>Kondisi Akhir Sukses</b>	Data <i>initiative</i> tersimpan ke dalam <i>database</i>
<b>Kondisi Akhir Gagal</b>	Data <i>initiative</i> tidak tersimpan ke dalam <i>database</i>

### G. *Flow of Event Use Case Pemberian Keputusan terhadap Initiative*

*Flow of Event* pada Tabel 3.7 menjelaskan tentang alur logika pada *Use Case* Pemberian Keputusan terhadap *Initiative*. Dari *flow of Event* tersebut dapat diketahui deskripsi singkat, prasyarat, alur utama, alur alternatif dan alur salah, serta kondisi akhir dari *Use Case* Pemberian Keputusan terhadap *Initiative*.

Tabel 3.7 *Flow of Event Use Case* Pemberian Keputusan terhadap *Initiative*

Deskripsi Use Case	
<b>Nama Use Case</b>	Pemberian Keputusan terhadap <i>Initiative</i>
<b>Deskripsi Singkat</b>	Dilakukan ketika ada <i>initiative</i> yang perlu dipertimbangkan terlebih dahulu oleh Manajer, apakah akan ditindaklanjuti ( <i>Go</i> ) atau tidak ditindaklanjuti ( <i>No-Go</i> )
<b>Aktor</b>	Manajer
<b>Prasyarat (kondisi)</b>	Tidak ada
<b>Alur Utama</b>	<ol style="list-style-type: none"> <li>1. Pengguna membuka halaman <i>initiative approval</i></li> <li>2. Sistem menampilkan <i>initiative-initiative</i> yang perlu mendapatkan persetujuan</li> <li>3. Memilih <i>initiative</i></li> <li>4. Pengguna mengklik “Simpan”</li> <li>5. Sistem mengkonfirmasi, apakah pengguna telah yakin</li> </ol>
<b>Alur Alternatif</b>	5.1 Jika pengguna belum yakin dengan <i>initiative</i> yang dipilihnya, pengguna dapat memeriksa pilihannya kembali dengan mengklik “Batal” terlebih dahulu
<b>Kondisi Akhir Sukses</b>	<i>Intiative</i> tersimpan dengan status, apakah <i>Go</i> atau <i>No-Go</i>
<b>Kondisi Akhir Gagal</b>	<i>Intiative</i> tidak tersimpan dengan status, apakah <i>Go</i> atau <i>No-Go</i>

### H. *Flow of Event Use Case Membuat Workplan*

*Flow of Event* pada Tabel 3.8 menjelaskan tentang alur logika pada *Use Case* Membuat *Workplan*. Dari *flow of Event* tersebut dapat diketahui deskripsi singkat, prasyarat, alur utama, alur alternatif dan alur salah, serta kondisi akhir dari *Use Case* Membuat *Workplan*.

Tabel 3.8 *Flow of Event Use Case* Membuat *Workplan*

Deskripsi Use Case	
<b>Nama Use Case</b>	Mencatat <i>Workplan</i>
<b>Deskripsi Singkat</b>	Mencatat <i>Workplan</i> dilakukan untuk kebutuhan <i>Design</i>
<b>Aktor</b>	Tim OPI
<b>Prasyarat (kondisi)</b>	Tidak ada
<b>Alur Utama</b>	<ol style="list-style-type: none"> <li>1. Pengguna membuka halaman <i>workplan</i></li> <li>2. Sistem menampilkan <i>initiative Go</i></li> <li>3. Pengguna memilih salah satu <i>initiative Go</i></li> <li>4. Pengguna memasukkan <i>workplan</i> untuk <i>initiative</i> yang dipilih</li> <li>5. Sistem akan mengecek apakah data yang dimasukkan telah lengkap atau belum</li> <li>6. Pengguna menekan tombol “Simpan”</li> </ol>
<b>Alur Alternatif</b>	5.1 Jika data yang dimasukkan belum lengkap, dan pengguna menekan tombol “Simpan”, maka akan keluar pesan peringatan bahwa data belum lengkap
<b>Kondisi Akhir Sukses</b>	Data <i>workplan</i> tersimpan ke dalam <i>database</i>
<b>Kondisi Akhir Gagal</b>	Data <i>workplan</i> tidak tersimpan ke dalam <i>database</i>

### I. *Flow of Event Use Case* Memperbarui Status Aktivitas *Workplan*

*Flow of Event* pada Tabel 3.9 menjelaskan tentang alur logika pada *Use Case Update Status Aktivitas Workplan*. Dari *flow of Event* tersebut dapat diketahui deskripsi singkat, prasyarat, alur utama, alur alternatif dan alur salah, serta kondisi akhir dari *Use Case Update Status Aktivitas Workplan*.

Tabel 3.9 *Flow of Event Use Case* Memperbarui Status Aktivitas *Workplan*

Deskripsi Use Case	
<b>Nama Use Case</b>	Update Status Aktivitas <i>Workplan</i>
<b>Deskripsi Singkat</b>	Update Status Aktivitas <i>workplan</i> salah satunya dilakukan dengan memanfaatkan perubahan tanggal karena setiap aktivitas memiliki waktu pengerjaan yang berkaitan dengan statusnya. Selain itu dilakukan dengan pengunggahan foto bukti.
<b>Aktor</b>	Sistem dan PIC
<b>Prasyarat (kondisi)</b>	Tidak ada
<b>Alur Utama</b>	<ol style="list-style-type: none"> <li>1. Pengguna membuka halaman <i>workplan progress</i></li> </ol>

<b>Deskripsi Use Case</b>	
	<ol style="list-style-type: none"> <li>2. Sistem menampilkan <i>workplan</i> dan statusnya</li> <li>3. PIC mengklik ID Aktivitas <i>Workplan</i></li> <li>4. Mengunggah berkas sebagai bukti bahwa aktivitas tersebut telah dilakukan/ditunda</li> <li>5. Mengisi status aktivitas</li> <li>6. PIC mengklik tombol “Simpan”</li> <li>7. Sistem mengkonfirmasi apakah data yang dimasukkan telah benar</li> <li>8. PIC menekan tombol “Ya”</li> </ol>
<b>Alur Alternatif</b>	<ol style="list-style-type: none"> <li>2.1 Jika selisih tanggal mulai aktivitas dan tanggal sekarang kurang dari 0, maka sistem akan menampilkan status “Belum Dimulai”</li> <li>2.2 Jika selisih tanggal mulai aktivitas dan tanggal sekarang lebih dari atau sama dengan 0, maka sistem akan memeriksa selisih tanggal akhir dan tanggal sekarang               <ol style="list-style-type: none"> <li>2.2.1 Jika selisih tanggal mulai aktivitas dan tanggal sekarang lebih dari 0, maka sistem akan menampilkan status “Melebihi Deadline”</li> <li>2.2.2 Jika selisih tanggal mulai aktivitas dan tanggal sekarang kurang dari atau sama dengan 0, maka sistem akan menampilkan status “Dalam Proses”</li> </ol> </li> <li>8.1 Jika pengguna belum yakin dengan data yang dimasukkan, pengguna dapat mengklik tombol “Tidak” dan memperbaikinya</li> </ol>
<b>Kondisi Akhir Sukses</b>	<ol style="list-style-type: none"> <li>1. Status “Belum Dimulai”, “Dalam Proses”, dan “Melebihi <i>Deadline</i>” dapat berubah sesuai perubahan tanggal</li> <li>2. Data aktivitas tersimpan ke dalam <i>database</i> setelah tombol “Simpan” diklik</li> <li>3. Status aktivitas berubah menjadi “Selesai” atau “Ditunda” jika PIC telah mengunggah berkas bukti</li> </ol>
<b>Kondisi Akhir Gagal</b>	<ol style="list-style-type: none"> <li>1. Status “Belum Dimulai”, “Dalam Proses”, dan “Melebihi <i>Deadline</i>” tidak dapat berubah sesuai perubahan tanggal</li> <li>2. Data aktivitas tidak tersimpan ke dalam <i>database</i> setelah tombol “Simpan” diklik</li> <li>3. Status aktivitas tidak berubah menjadi “Telah Tercapai” ketika PIC selesai mengunggah berkas bukti</li> </ol>

## J. *Flow of Event Use Case Notifikasi via SMS*

*Flow of Event* pada Tabel 3.10 menjelaskan tentang alur logika pada *Use Case* Notifikasi via SMS. Dari *flow of Event* tersebut dapat diketahui deskripsi singkat, prasyarat, alur utama, alur alternatif dan alur salah, serta kondisi akhir dari *Use Case* Notifikasi via SMS.

Tabel 3.10 *Flow of Event Use Case* Notifikasi via SMS

<b>Deskripsi Use Case</b>	
<b>Nama Use Case</b>	Notifikasi via SMS
<b>Deskripsi Singkat</b>	Notifikasi via SMS digunakan untuk kebutuhan <i>Deliver</i>
<b>Aktor</b>	Manajer
<b>Prasyarat (kondisi)</b>	Tidak ada
<b>Alur Utama</b>	<ol style="list-style-type: none"> <li>1. Pengguna membuka halaman notifikasi</li> <li>2. Sistem <i>refresh</i> halaman notifikasi dalam web per waktu tertentu</li> <li>3. Per <i>refresh</i>, sistem menghitung jumlah aktivitas <i>workplan</i> sesuai statusnya (rekapitulasi <i>workplan</i>)</li> <li>4. Per <i>refresh</i>, sistem menampilkan rekapitulasi dalam web</li> <li>5. Per <i>refresh</i>, sistem mengecek dalam <i>database</i> SMS Gateway, apakah terdapat SMS dengan tanggal sekarang sistem</li> </ol>
<b>Alur Alternatif</b>	<ol style="list-style-type: none"> <li>5.1 Jika tidak ada, maka sistem akan mengirimkan SMS notifikasi ke Manajer</li> <li>5.2 Jika ada, maka sistem tidak akan mengirimkan SMS notifikasi ke Manajer</li> </ol>
<b>Kondisi Akhir Sukses</b>	<ol style="list-style-type: none"> <li>1. Sistem menampilkan rekapitulasi <i>workplan</i></li> <li>2. SMS terkirim ke Manajer</li> </ol>
<b>Kondisi Akhir Gagal</b>	<ol style="list-style-type: none"> <li>1. Sistem tidak dapat menampilkan rekapitulasi <i>workplan</i></li> <li>2. SMS tidak terkirim ke Manajer</li> </ol>

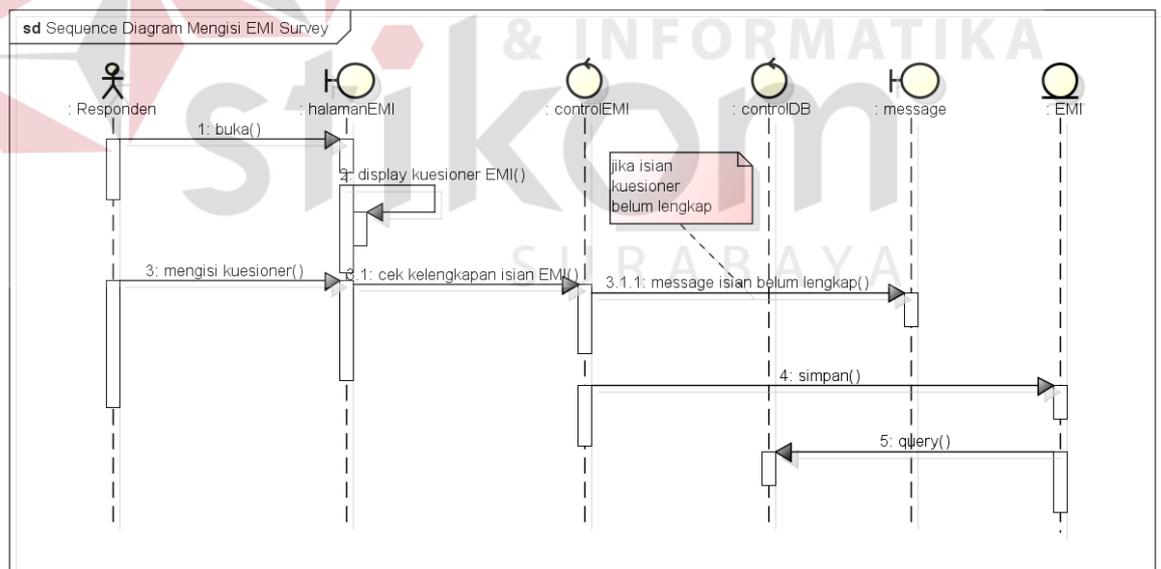
### 3.3.4 *OPTIMUS+* Sequence Diagram

*Sequence Diagram* menjelaskan realisasi dari *use case diagram*. *Diagram sequence* merupakan salah satu diagram interaksi yang mengurutkan

alur berdasarkan waktu. Berikut ini adalah *sequence diagram* dari masing-masing *use case diagram* OPTIMUS+:

### A. Mengisi *EMI Survey*

Proses dimulai ketika responden membuka halaman *EMI Survey* dalam *website*, kemudian web memberi respon dengan menampilkan halaman *EMI Survey*. Di dalam halaman *EMI Survey* tersebut, terdapat pertanyaan-pertanyaan kuesioner yang akan dijawab oleh responden. Selanjutnya, sistem akan memeriksa dengan *control*, apakah isian reponden telah lengkap atau belum. Jika belum, maka sistem akan menampilkan pesan agar responden melengkapi isian kuesionernya. Jika isian telah lengkap, maka data-data isian tersebut akan masuk ke dalam *database*. *Sequence Diagram* tersebut dapat dilihat pada Gambar 3.22.

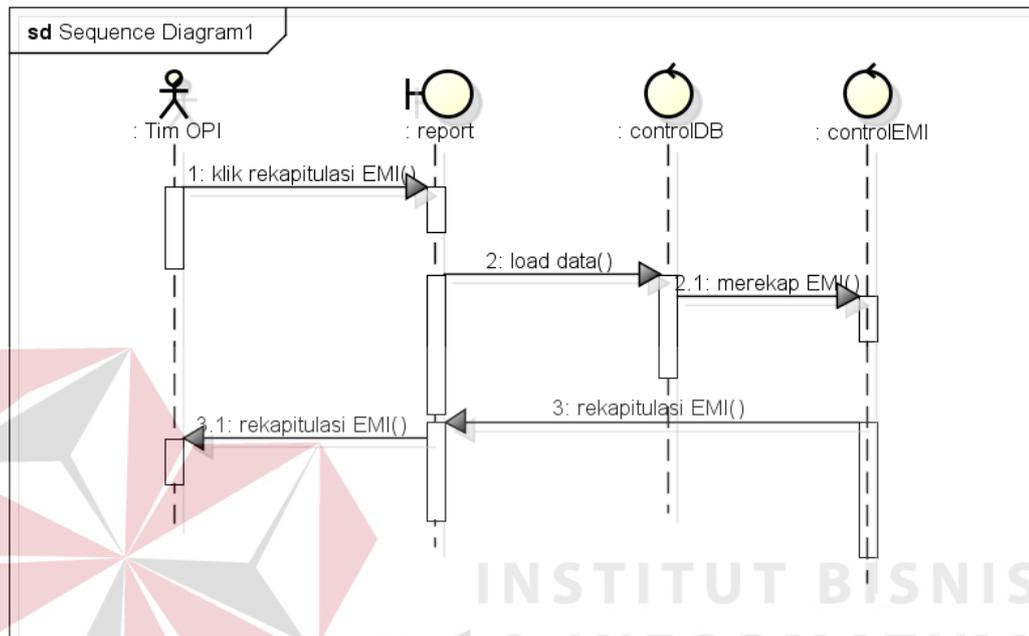


Gambar 3. 22 *Sequence Diagram* Mengisi *EMI Survey*

### B. Menampilkan Rekapitulasi *EMI Survey*

Perekapan *EMI Survey* dilakukan oleh sistem ketika ada perintah dari Tim OPI berupa klik untuk membuka rekap. Kemudian data-data yang dibutuhkan

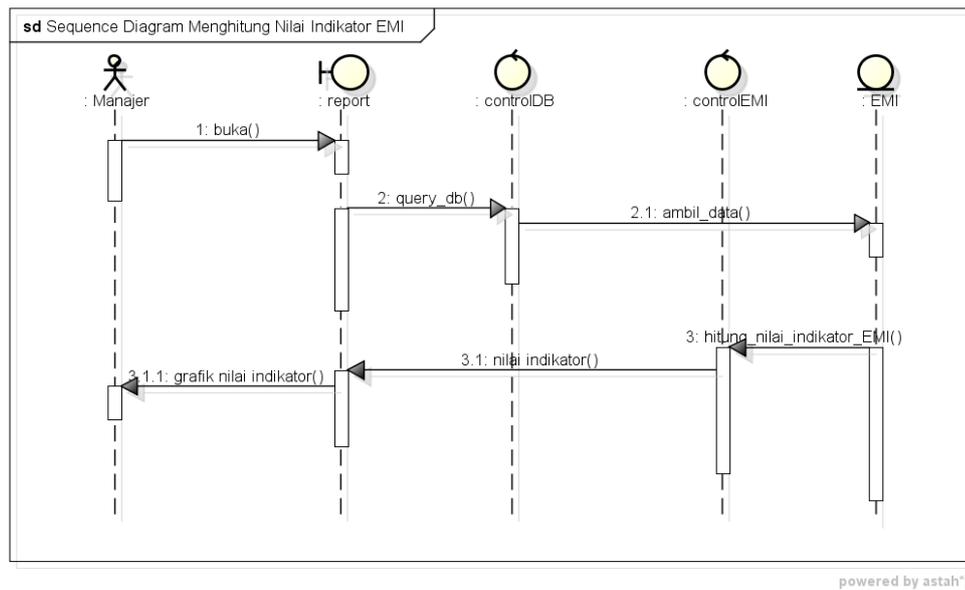
dari *database* dimuat untuk ditampilkan dalam bentuk rekapitulasi. Rekapitulasi disini berupa berkas excel. *Sequence Diagram* tersebut dapat dilihat pada Gambar 3.23.



Gambar 3. 23 *Sequence Diagram* Menampilkan Rekapitulasi *EMI Survey*

### C. Menampilkan Grafik *EMI Survey*

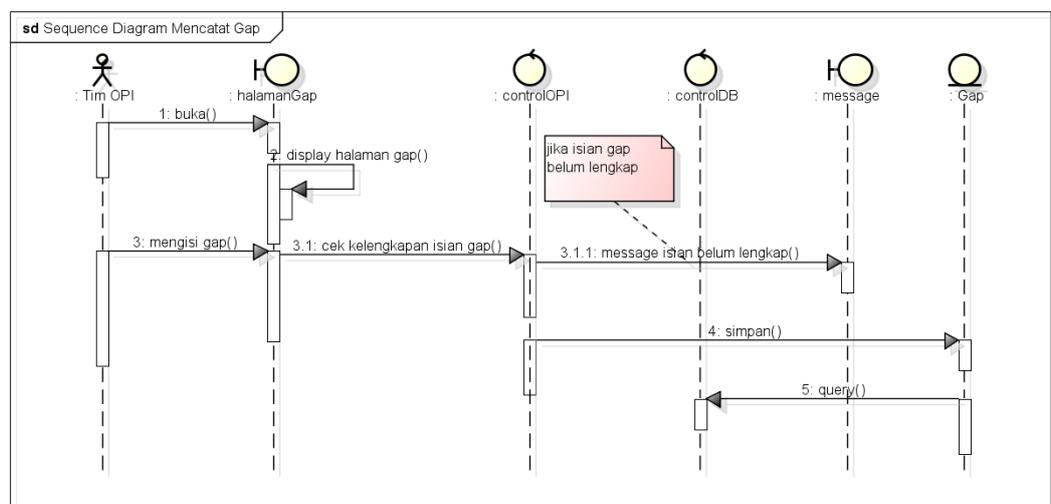
Penghitungan hasil *EMI Survey* dilakukan ketika ada perintah berupa klik dari Manajer untuk membuka grafik *EMI Survey*. Kemudian data yang dibutuhkan dari *database* di-load, dihitung oleh sistem sesuai rumus *EMI Survey* yang telah dibahas sebelumnya, lalu diperoleh hasil berupa angka-angka yang ditampilkan dalam grafik. *Sequence Diagram* tersebut dapat dilihat pada Gambar 3.24.



Gambar 3. 24 Sequence Diagram Menampilkan Grafik EMI Survey

#### D. Mencatat Gap

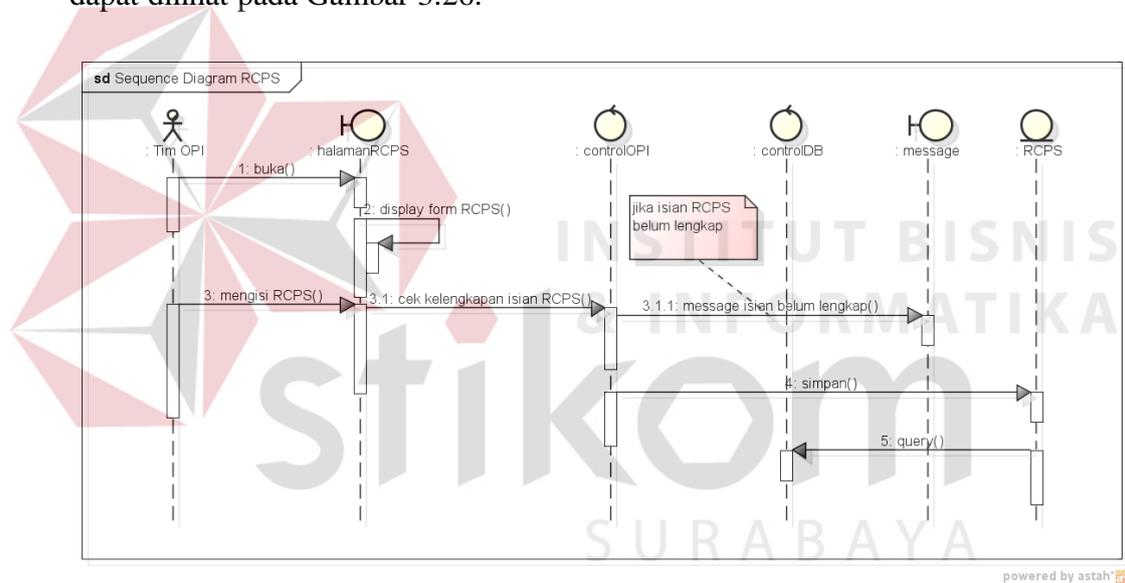
Untuk gap, Tim OPI dapat membuka halaman Gap. Dalam halaman gap tersebut terdapat data-data gap yang harus diisi. Data-data yang diisi akan dicek oleh sistem, tentang kelengkapannya. Jika lengkap, maka sistem akan menyimpan data-data tersebut ke dalam database. Jika belum, Tim OPI harus melengkapinya terlebih dahulu. Sequence Diagram tersebut dapat dilihat pada Gambar 3.25.



Gambar 3. 25 Sequence Diagram Mencatat Gap

### E. Mencatat RCPS

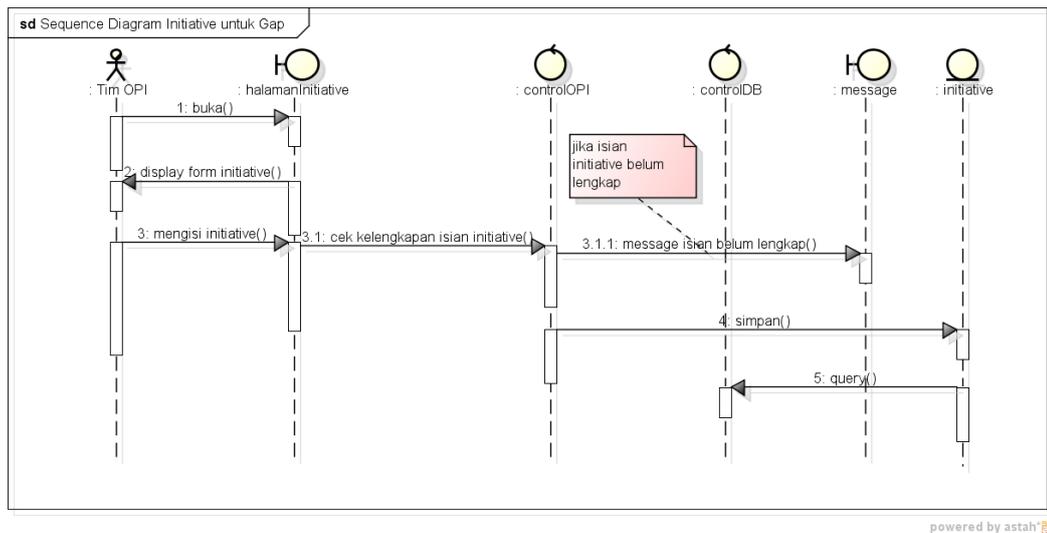
Pencatatan RCPS dimulai dengan Tim OPI mengklik menu RCPS. Lalu sistem akan menampilkan halaman RCPS yang di dalamnya terdapat masalah-masalah yang sudah/belum memiliki RCPS. Masalah yang belum memiliki RCPS dapat ditindaklanjuti dengan mengisi/mengentri form entri RCPS. Setelah itu, sistem akan merespon saat pengguna mengklik tombol “Simpan”. Sistem akan memberikan peringatan, jika isian belum lengkap. Sistem akan menyimpan data isian ke dalam *database*, jika isian telah lengkap. *Sequence Diagram* tersebut dapat dilihat pada Gambar 3.26.



Gambar 3. 26 *Sequence Diagram* Mencatat RCPS

### F. Membuat *Initiative* untuk *Gap*

Pembuatan *initiative* untuk *gap* dilakukan dengan cara membuka halaman *initiative*. Pada halaman *initiative* akan ditemukan masalah-masalah yang telah melalui tahap RCPS. Ketika muncul halaman entri ini *initiative* maka Tim OPI akan mengisikan data-data yang dibutuhkan. Selanjutnya, sistem akan menyimpannya ke dalam *database* jika data yang diisikan telah lengkap. *Sequence Diagram* tersebut dapat dilihat pada Gambar 3.27.

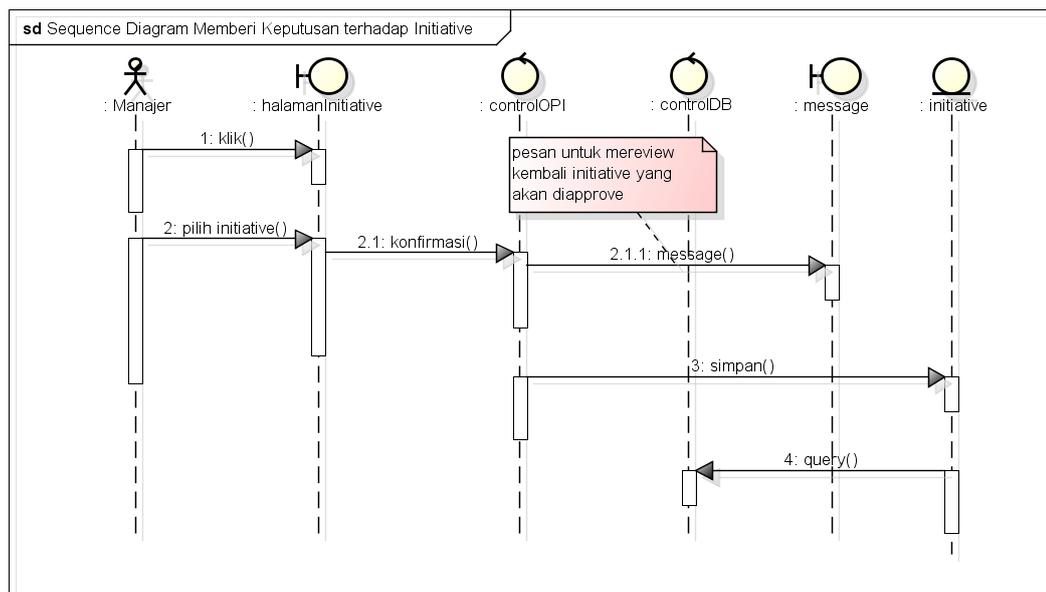


Gambar 3. 27 Sequence Diagram Membuat Initiative Untuk Gap

### G. Memberi Keputusan terhadap Initiative

Pemberian keputusan terhadap *initiative* dilakukan dengan membuka halaman *initiative approval*. Sistem akan menampilkan topik masalah yang telah melalui tahap RCPS dan pembuatan *initiative*. Dari sini, manajer selaku pemegang *initiative* dapat memilih *initiative-initiative* yang akan disetujui. Sebelum *initiative* yang disetujui masuk ke dalam *database*, terlebih dahulu sistem akan melakukan konfirmasi, apakah benar *initiative* tersebut yang disetujui. Sebab pemberian keputusan ini berpengaruh untuk proses selanjutnya.

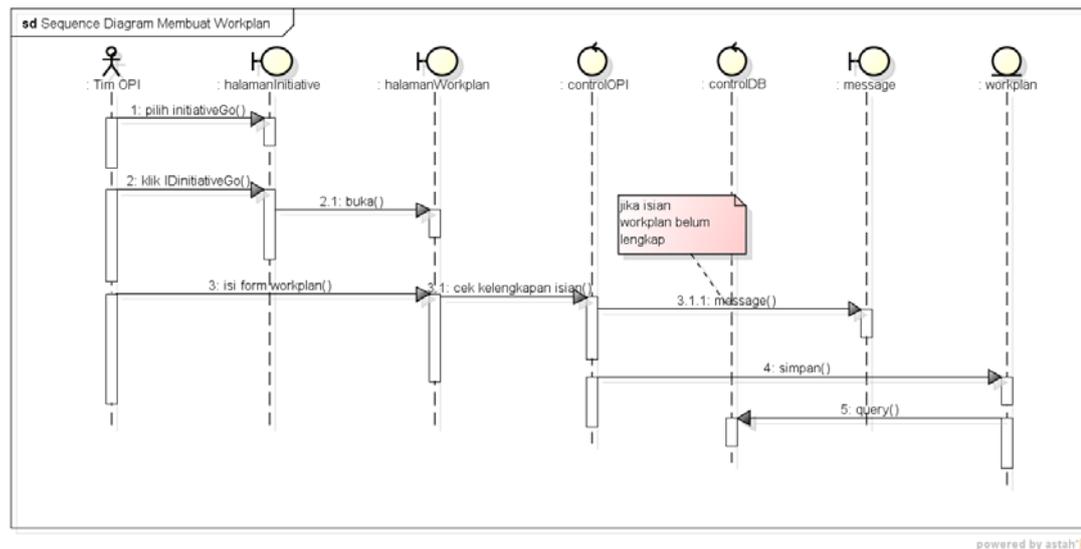
*Sequence Diagram* tersebut dapat dilihat pada Gambar 3.28.



Gambar 3. 28 *Sequence Diagram* Pemberian Keputusan terhadap *initiative*

## H. Membuat *Workplan*

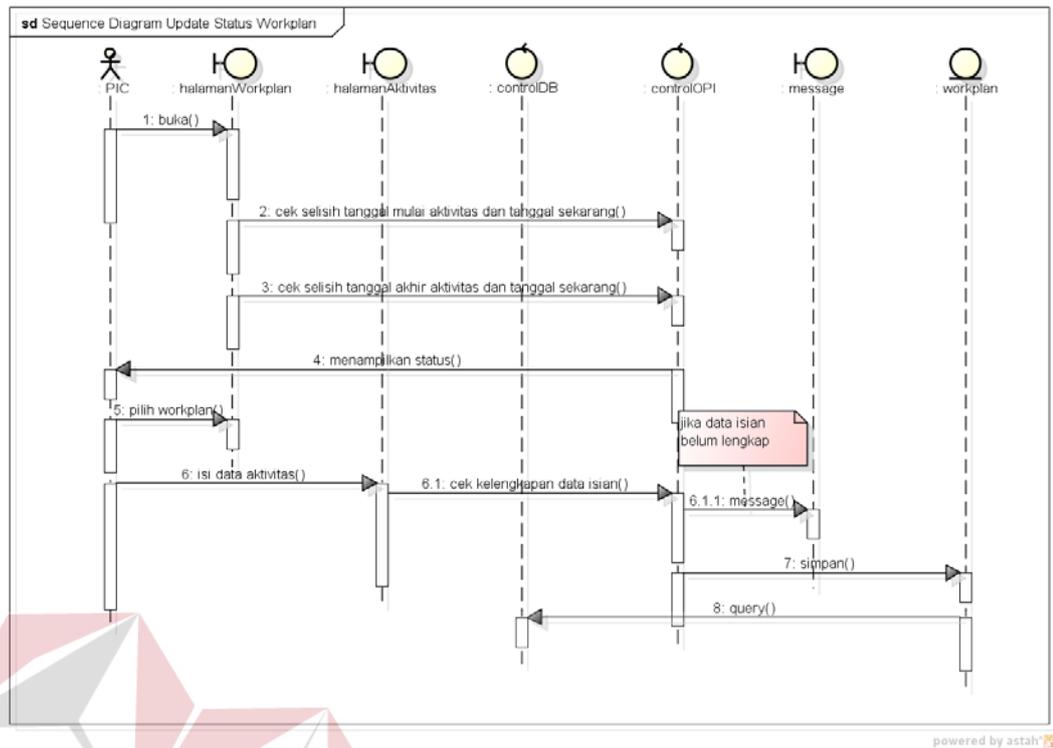
*Initiative-Go* adalah *initiative* yang disetujui oleh manajer. Dari manajer, proses berlanjut ke pembuatan *workplan* yang dihandle oleh Tim OPI kembali. Tim OPI dapat mengklik *initiative* yang akan dibuatkan *workplannya* dengan mengisi data-data yang dibutuhkan. Setelah itu, sistem akan mengecek kelengkapan data tersebut. Jika lengkap, simpan ke dalam *database*. Jika belum, sistem akan memberikan peringatan. *Sequence Diagram* tersebut dapat dilihat pada Gambar 3.29.



Gambar 3. 29 Sequence Diagram Membuat Workplan

### I. Update Status Aktivitas *Workplan*

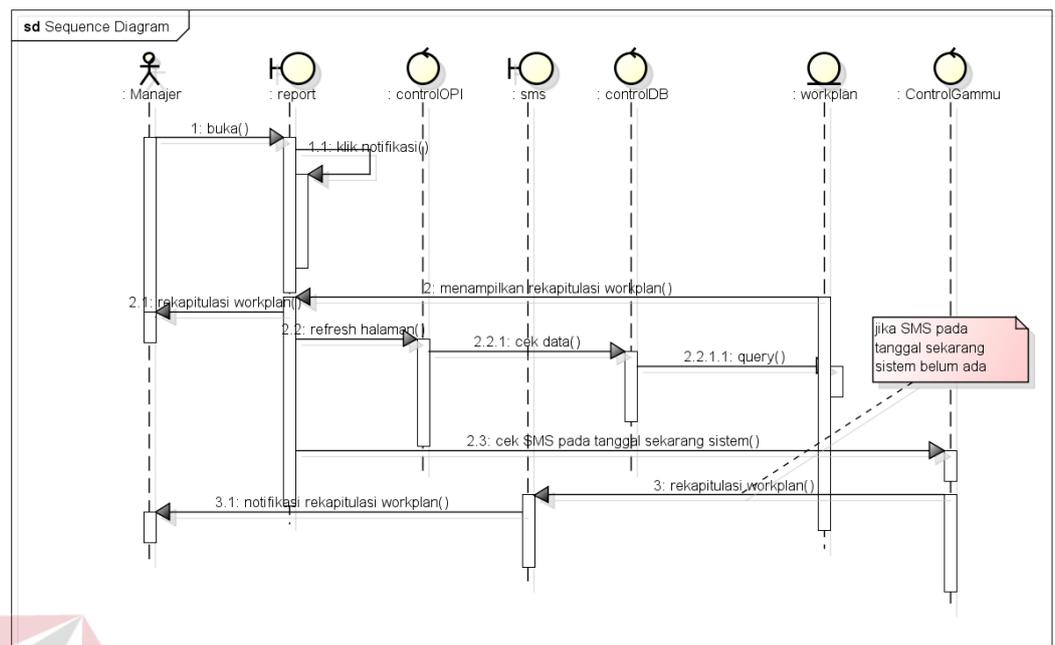
Ketika PIC membuka halaman *workplan*, maka akan ditampilkan data-data *workplan* yang ada beserta aktivitasnya dan status-status yang telah *update* secara otomatis oleh sistem berdasarkan perubahan tanggal. Status-status ini diperoleh karena sistem mengecek selisih tanggal mulai/akhir dengan tanggal sekarang. Status-status berdasarkan perubahan tanggal tersebut antara lain: 'Belum dimulai', 'Dalam Proses, dan 'Melebihi *Deadline*'. Kemudian, dari data-data dan status-status yang ditampilkan tersebut, PIC dapat memilih *id workplan* mana yang akan diupdate statusnya menjadi 'Ditunda' atau 'Selesai' melalui pengisian data. Pengisian data selanjutnya akan diperiksa oleh sistem, jika lengkap maka sistem akan menyimpan data-data tersebut ke dalam *database* dan mengubah status yang semula 'Belum dimulai', 'Dalam Proses, dan 'Melebihi *Deadline*' menjadi 'Ditunda' atau 'Selesai'. *Sequence Diagram* tersebut dapat dilihat pada Gambar 3.30.



Gambar 3. 30 *Sequence Diagram Update Status Aktivitas Workplan*

#### J. Notifikasi (via SMS)

Halaman notifikasi selalu terbuka, dan sistem akan melakukan *auto refresh* halaman tersebut per waktu tertentu. Ketika terjadi *refresh*, sistem akan selalu menghitung berapa jumlah aktivitas *workplan* sesuai statusnya. Selain itu, per refresh tersebut sistem akan melakukan pengecekan apakah pada *database* Gammu sudah ada SMS notifikasi pada tanggal sekarang sistem atau belum. Jika belum, maka sistem akan melakukan pengiriman SMS. Jika sudah, maka sistem tidak akan melakukan pengiriman SMS lagi. *Sequence Diagram* tersebut dapat dilihat pada Gambar 3.31.

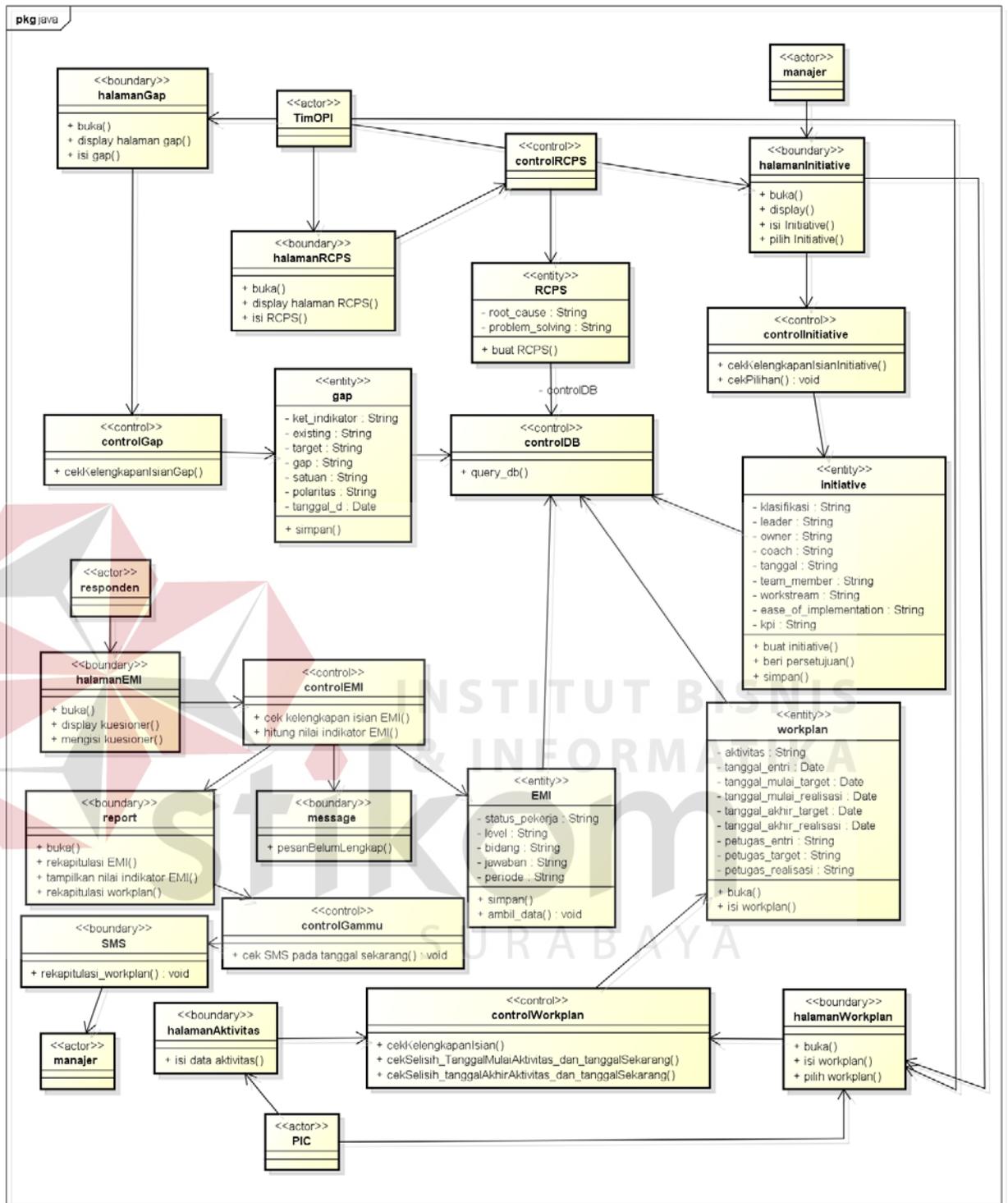


Gambar 3. 31 *Sequence Diagram* Notifikasi (via SMS)

### 3.3.5 *Class Diagram* OPTIMUS+

*Class Diagram* digunakan untuk menampilkan kelas-kelas di dalam sistem dan relasi antar kelas tersebut (menunjukkan interaksi antar kelas di dalam aplikasi). Menurut Sholih, Satu diagram kelas menampilkan subset dari kelas-kelas dan relasinya. Diagram kelas lainnya, mungkin menampilkan kelas-kelas termasuk atribut dan operasi dari kelas-kelas pembentuk diagram. Sedangkan diagram kelas yang lainnya lagi, mungkin menampilkan paket-paket kelas dan relasi antar paket-paket.

Gambar 3.32 menunjukkan *class diagram* OPTIMUS+. *Class diagram* tersebut merupakan gambaran lengkap terhadap aplikasi yang akan dibangun. *Class diagram* tersebut juga menggambarkan aplikasi sebelum dituliskan menjadi kode program.



Gambar 3. 32 Class Diagram OPTIMUS+

### 3.4 Desain *Input* dan *Output* Sistem

Pada tahap ini dilakukan perancangan *input/output* untuk berinteraksi antara pengguna dengan sistem. Desain antarmuka ini dibuat dengan menggunakan perangkat lunak *Pencil*. Adapun desain tampilan yang akan digunakan yaitu sebagai berikut:

#### 3.4.1 Rancangan Tampilan *EMI Survey*

Tampilan *EMI Survey* akan muncul ketika pengguna mengklik link *EMI Survey*. Pada tampilan ini terdapat Combo-box untuk memilih status responden serta waktu pengisian *EMI Survey*. Selain itu, terdapat daftar pertanyaan dan pilihan jawaban kuesioner. Rancangan tampilan dapat dilihat pada Gambar 3.33.

The screenshot shows a web form for an EMI Survey. At the top, there is a header for 'Institut Bisnis & Informatika Stikom Surabaya'. Below the header, there is a 'Login OPI' field. The main form area contains several sections:

- Semester :** xxx
- Status Pekerja**
  - Pegawai
  - Outsourcing
- Level**
  - Struktural
  - Fungsional
- Bagian**
  - Pilih-

Below these fields is a section titled 'Keterbukaan dan transparansi' with four survey questions, each followed by a set of radio buttons for response scales:

- Pekerja telah berani mengutarakan pendapatnya secara terbuka, sekalipun berbeda pendapat dengan pimpinanya  STS  TS  CTS  CS  S  SS
- Kondisi saat ini pimpinan meluangkan cukup waktu untuk berkomunikasi secara terbuka bersama pekerja  STS  TS  CTS  CS  S  SS
- Pekerja telah mengetahui target dan pencapaian kinerja bagiannya  STS  TS  CTS  CS  S  SS
- Kondisi saat ini masing-masing bagian saling mendukung, sehingga tidak terjadi saling menyalahkan  STS  TS  CTS  CS  S  SS

At the bottom of the form, there is a legend: STS = Sangat Tidak Setuju | TS = Tidak Setuju | CTS = Cenderung Tidak Setuju | S = Setuju | SS = Sangat Setuju. A 'Save' button is located at the bottom right of the form.

Gambar 3. 33 Tampilan *EMI Survey*

#### 3.4.2 Rancangan Tampilan *Gap*

Tampilan *gap* akan muncul ketika pengguna memilih menu *gap* pada menu. Pada tampilan ini terdapat *Combo-button* Unit/Rayon untuk memilih *gap*

dari Rayon mana yang akan ditampilkan. Selain itu, terdapat *button* Entri *Problem/Gap*. Jika button ini diklik, maka akan menampilkan suatu halaman baru untuk mengentri *gap*. Rancangan tampilan dapat dilihat pada Gambar 3.34.

Kode Diagnose	Capture Awal	Topik Problem	Existing	Target	Gap	Satuan	Polaritas	Edit

Gambar 3. 34 Tampilan *Gap*

**ENTRI GAP**

Unit : Nama Unit

Kode Diagnose

Topik Problem

Nilai Existing

Nilai Target

Satuan

Polaritas

Gap

Upload File (Maks 3MB \*.jpg/ \*.pdf)

Gambar 3. 35 Entri *Gap*

Tampilan halaman Entri *gap* pada Gambar 3.35 digunakan untuk masukan data-data *gap* yang dibutuhkan, salah satunya adalah tombol unggah. Berkas unggah dalam entri *gap* ini adalah lampiran berupa berkas .jpg atau .pdf. Lampiran ini nantinya difungsikan sebagai bukti bahwa memang terdapat permasalahan yang mungkin untuk mendapatkan perbaikan.

### 3.4.3 Rancangan Tampilan RCPS

Tampilan RCPS pada Gambar 3.36 akan muncul ketika pengguna memilih menu RCPS. Tampilan ini menampilkan *root-cause* atau penyebab dari suatu permasalahan/*gap/problem* dan *problem solving* dari permasalahan tersebut. Terdapat *combo-button* yang terdiri dari dua pilihan, yaitu: *Problem* belum ada RCPS dan *Problem* sudah ada RCPS. *Problem* belum ada RCPS artinya terdapat *problem/gap* yang belum dilanjutkan ke tahap RCPS sama sekali. Sedangkan RCPS sudah ada *gap* artinya suatu *gap* sudah memiliki RCPS.

Kode Diagnose	Capture RCPS	Root Cause	Problem Solving	Topik Masalah	Existing	Target	Gap	Satuan	Polaritas
<a href="#">Upload RCPS</a>									

Gambar 3. 36 Tampilan RCPS

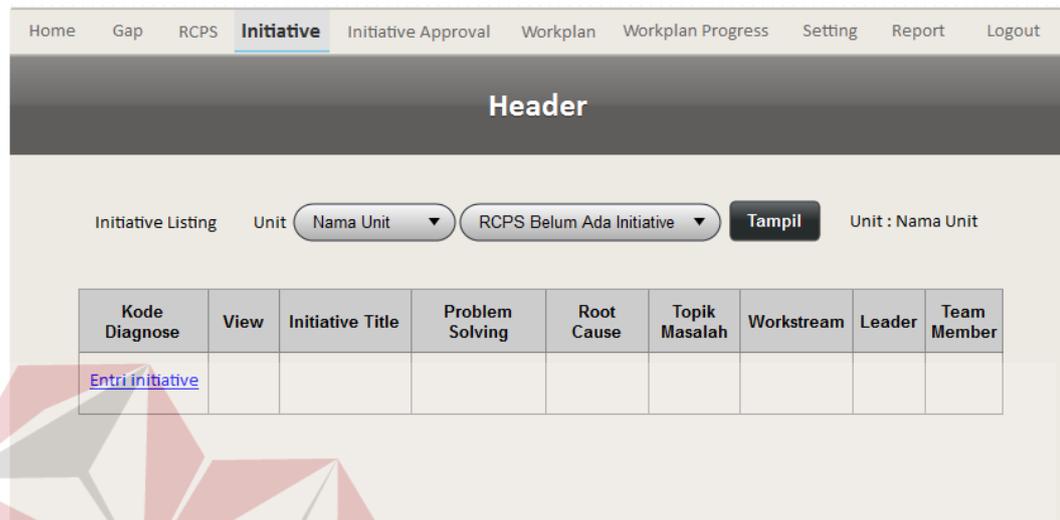
Pada gambar di atas juga terdapat link *Unggah RCPS*. Link *Unggah RCPS* adalah link ke halaman unggah. *Unggah* dapat dilakukan lebih dari sekali, sebanyak RCPS yang ditemukan. Misalnya, *gap/Diagnose* awal adalah gangguan listrik. *Root cause* bisa jadi lebih dari satu, yaitu karena binatang yang merusak kabel, karena usia peralatan, karena hujan/petir, atau karena isolator yang kotor, dan masih ada lagi alasan yang lain. Untuk itu, *Unggah RCPS* ini diperlukan, agar masing-masing sebab tadi menjadi jelas. Rancangan tampilan *Unggah RCPS* ini dapat dilihat pada Gambar 3.37.

Gambar 3. 37 Tampilan Unggah RCPS

#### 3.4.4 Rancangan Tampilan *Initiative*

Tampilan *initiative* menampilkan *initiative* . Pada tampilan terdapat *combo-button* yang terdiri dari dua pilihan, yaitu: RCPS belum ada *initiative* dan

RCPS sudah ada *initiative* . RCPS belum ada *initiative* artinya RCPS tersebut belum memiliki *initiative* . Untuk itu, di kolom Kode *Diagnose* diberi link untuk entri *initiative*. Rancangan tampilan dapat dilihat pada Gambar 3.38.



Gambar 3. 38 Tampilan *Initiative*

Gambar 3. 39 Tampilan Entri *Initiative*

Pada halaman entri *initiative* dibutuhkan masukan seperti judul *initiative*, nama-nama yang termasuk sebagai tim member serta *leader* dalam *gap* dengan kode *Diagnose* yang terpilih. Rancangan tampilan dapat dilihat pada Gambar 3.39.

### 3.4.5 Rancangan Tampilan *Initiative Approval*

Tampilan *initiative approval* adalah tampilan untuk manajer. terdiri dari Kode *initiative* (kode yang diperoleh karena suatu *gap* telah melalui tahap RCPS dan tahap *Initiative* sebelumnya), judul *initiative*, dan keputusan manajer (*Go/No-Go*). Rancangan tampilan *Initiative Approval* dapat dilihat pada Gambar



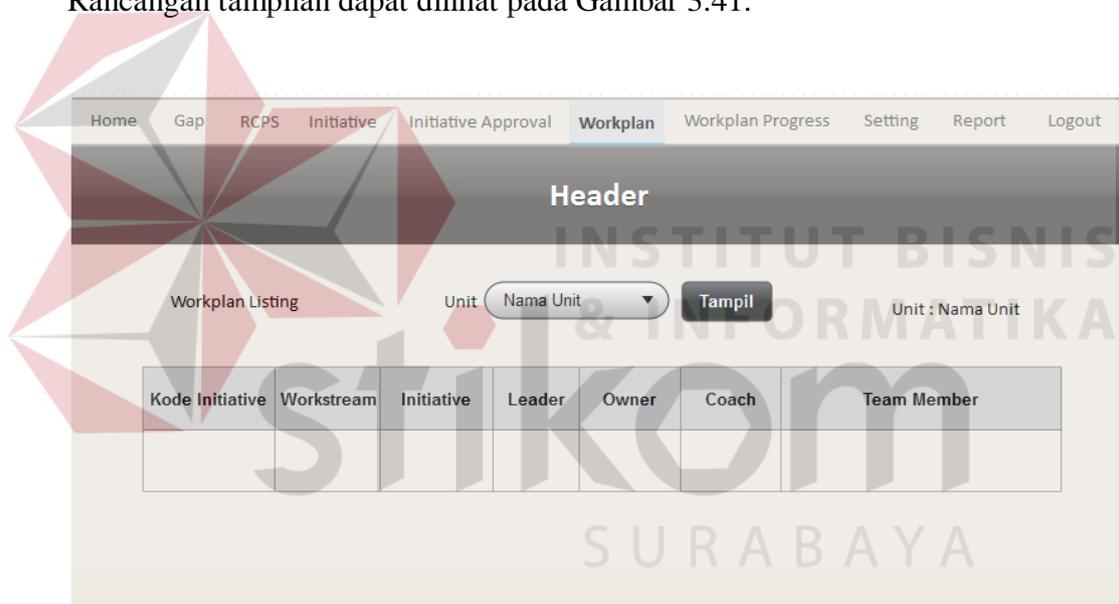
Gambar 3. 40 Tampilan Halaman Persetujuan *Initiative*

Keputusan manajer ini berpengaruh pada tahap selanjutnya. Karena *initiative* yang telah dipertimbangkan untuk *Go* akan maju dan direalisasikan menjadi *workplan* atau rangkaian aktivitas kerja. Jika *No-Go* maka *initiative* itu berakhir di tahap itu saja.

Keputusan manajer ini diperoleh tidak dengan keputusan sepihak oleh manajer saja, melainkan terdapat rapat untuk berdiskusi. Dalam aplikasi ini, rapat serta proses menyangkut persetujuan manajer, diwujudkan seperti tampilan ini serta fungsinya.

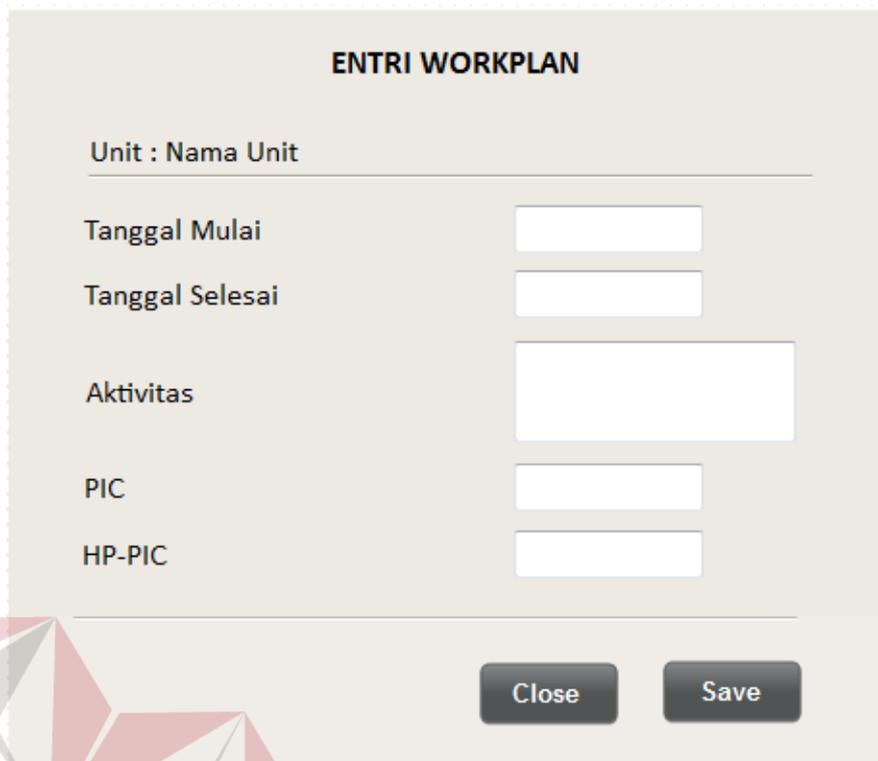
### 3.4.6 Rancangan Tampilan *Workplan*

Tampilan *workplan* akan muncul jika pengguna memilih menu *workplan*. *Initiative-Go* akan masuk ke dalam tahap ini, yaitu pembuatan *workplan*. Rancangan tampilan dapat dilihat pada Gambar 3.41.



Gambar 3. 41 Tampilan *Workplan*

Pada Gambar 3.41 terdapat *link* menuju halaman untuk memasukkan data-data *workplan*. Melalui *form* entri *workplan* tersebut, pengguna dapat menambahkan aktivitas *workplan* apa saja yang akan dilakukan serta waktu yang dibutuhkan untuk pengerjaan aktivitas tersebut. Waktu ditandai dengan tanggal mulai dan tanggal akhir. Rancangan tampilan entri *workplan* dapat dilihat pada Gambar 3.42.



**ENTRI WORKPLAN**

Unit : Nama Unit

---

Tanggal Mulai

Tanggal Selesai

Aktivitas

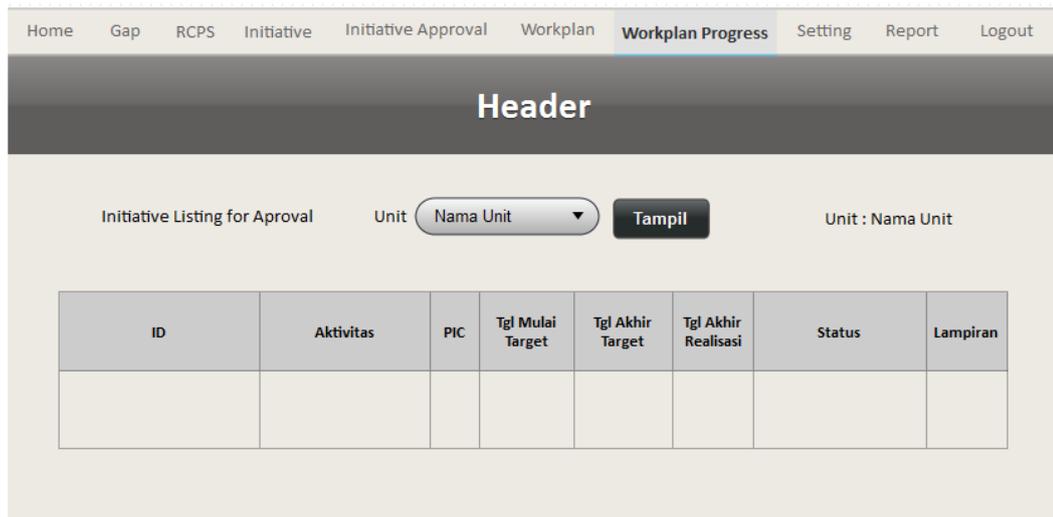
PIC

HP-PIC

---

Gambar 3. 42 Tampilan Entri *Workplan*

Gambar 3.43 menjelaskan tentang tampilan *workplan* yang telah dimasukkan. Tanggal mulai dan akhir aktivitas yang dimasukkan pada form entri *workplan* sebelumnya, akan menjadi salah satu patokan bagi perubahan status aktivitas *workplan* (“Belum dimulai” jika tanggal sekarang belum masuk tanggal mulai aktivitas, “Dalam Proses” jika tanggal sekarang berada di antara tanggal mulai dan akhir aktivitas, dan “Melebihi Deadline” jika tanggal sekarang melewati tanggal akhir aktivitas dan aktivitas tersebut belum terselesaikan). Status *workplan* ditandai dengan warna dan keterangan di sampingnya.



Home Gap RCPS Initiative Initiative Approval Workplan **Workplan Progress** Setting Report Logout

**Header**

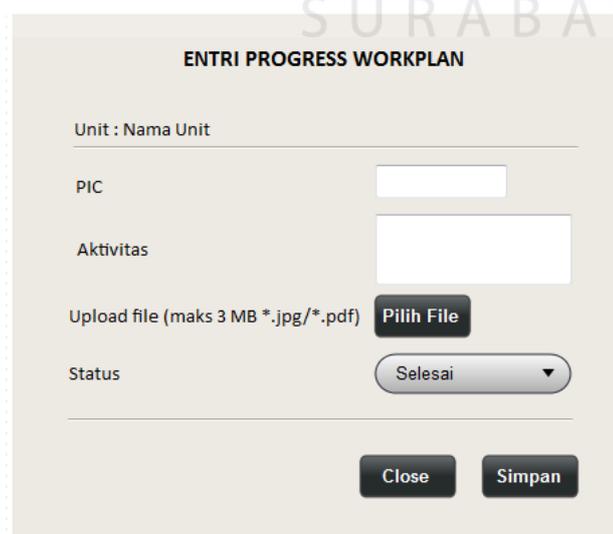
Initiative Listing for Approval Unit: Nama Unit  Unit : Nama Unit

ID	Aktivitas	PIC	Tgl Mulai Target	Tgl Akhir Target	Tgl Akhir Realisasi	Status	Lampiran

Gambar 3. 43 Tampilan *Workplan Progress*

### 3.4.7 Rancangan Tampilan Entri *Workplan Progress*

Tampilan ini akan muncul jika pengguna memilih menu *Workplan Progress*. Halaman entri ini digunakan untuk mengupdate status aktivitas *workplan* menjadi 'Selesai' atau 'Ditunda'. Combo-box status berisi pilihan apakah suatu aktivitas 'Selesai' atau 'Ditunda'. Rancangan tampilan dapat dilihat pada Gambar 3.44.



**ENTRI PROGRESS WORKPLAN**

Unit : Nama Unit

PIC

Aktivitas

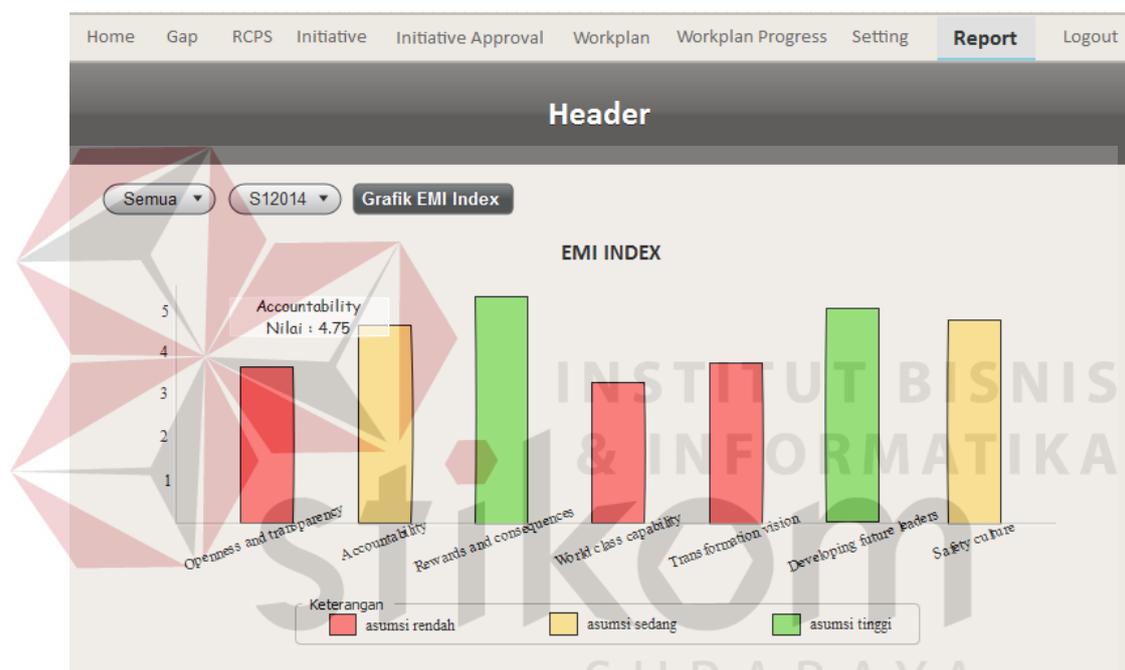
Upload file (maks 3 MB \*.jpg/\*.pdf)

Status

Gambar 3. 44 Tampilan Entri *Workplan Progress*

### 3.4.8 Rancangan Tampilan Grafik *EMI Survey*

Tampilan ini akan muncul jika pengguna memilih menu *report*, lalu mengklik button grafik *EMI Survey*. Grafik ini merupakan salah satu *output* dari pengisian *EMI Survey* yang dilakukan oleh responden sebelumnya. Dalam grafik terdapat nilai dari tujuh indikator *EMI Survey*. Rancangan tampilan dapat dilihat pada Gambar 3.45.



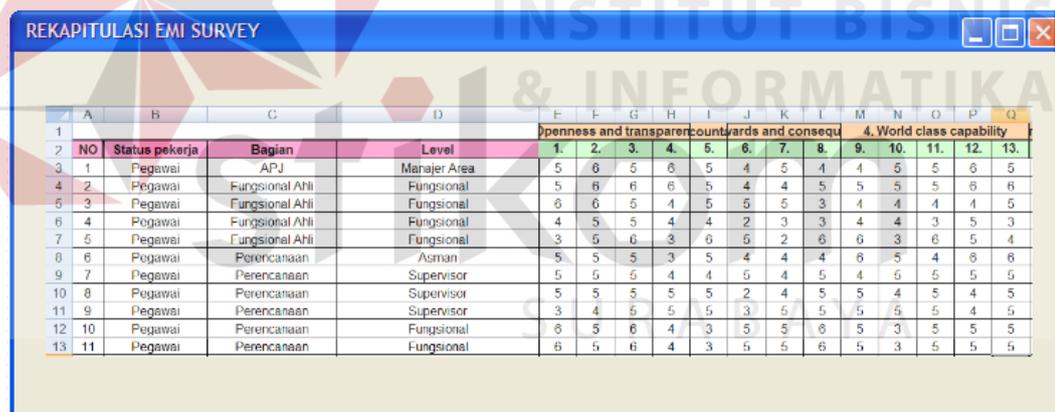
Gambar 3. 45 Tampilan Grafik *EMI Survey*

Terdapat tiga warna batang grafik, yaitu merah yang menandakan bahwa asumsi responden terhadap sebuah indikator rendah; kuning yang menandakan bahwa asumsi responden terhadap sebuah indicator sedang; dan hijau yang menandakan bahwa asumsi responden terhadap sebuah indikator tinggi. Setiap batang pada grafik akan memunculkan warna sesuai dengan nilainya. Untuk nilai indikator 0 hingga 4,5, batang grafik akan berwarna merah; untuk nilai indikator lebih dari 4,5 hingga kurang dari 5, batang grafik akan berwarna kuning; dan

untuk nilai indikator lebih dari atau sama dengan 5, batang grafik akan berwarna hijau. Pewarnaan tiap batang grafik sesuai nilainya ini dapat digunakan untuk mempermudah melihat indikator mana yang memiliki asumsi rendah. Sebab tiga indikator dengan asumsi paling rendah akan dibahas lebih lanjut dengan cara dijadikan bahan *gap*.

### 3.4.9 Rancangan Tampilan Rekapitulasi *EMI Survey*

Untuk mendapatkan rekapitulasi *EMI Survey*, pengguna harus ke menu *report*, kemudian mengklik button rekapitulasi. Untuk selanjutnya, rekapitulasi *EMI Survey* akan muncul sebagai unduhan dalam format *.xlsx*. Rancangan tampilan dapat dilihat pada Gambar 3.46.

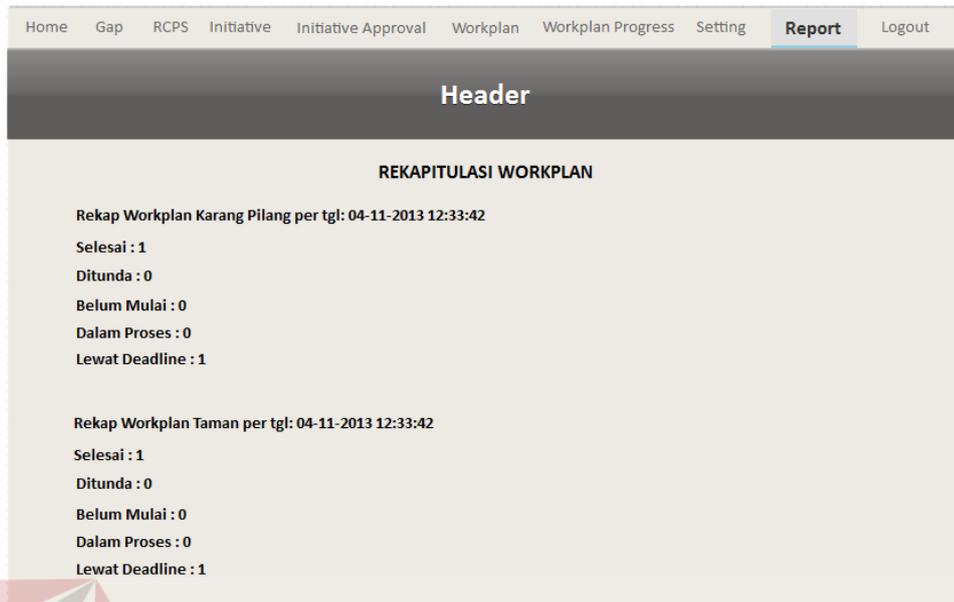


REKAPITULASI EMI SURVEY				EMI SURVEY INDICATORS												
NO	Status pekerja	Bagian	Level	1. Openness and transparency	2. Accountability	3. Customer focus	4. Innovation	5. Quality	6. Cost efficiency	7. Risk management	8. Environmental friendliness	9. World class capability	10. Customer focus	11. Innovation	12. Quality	13. Risk management
1	Pegawai	APJ	Manajer Area	5	6	5	6	5	4	5	4	4	5	5	6	5
2	Pegawai	Fungsional Ahli	Fungsional	5	6	6	6	5	4	4	5	5	5	5	6	6
3	Pegawai	Fungsional Ahli	Fungsional	6	6	5	4	5	5	5	3	4	4	4	4	5
4	Pegawai	Fungsional Ahli	Fungsional	4	5	5	4	4	2	3	3	4	4	3	5	3
5	Pegawai	Fungsional Ahli	Fungsional	3	5	6	3	6	5	2	6	6	3	6	5	4
6	Pegawai	Perencanaan	Asman	5	5	5	3	5	4	4	4	8	5	4	6	6
7	Pegawai	Perencanaan	Supervisor	5	5	5	4	4	5	4	5	4	5	5	5	5
8	Pegawai	Perencanaan	Supervisor	5	5	5	5	5	2	4	5	5	4	5	4	5
9	Pegawai	Perencanaan	Supervisor	3	4	5	5	5	3	5	5	5	5	5	4	5
10	Pegawai	Perencanaan	Fungsional	8	5	6	4	3	5	5	8	5	3	5	5	5
11	Pegawai	Perencanaan	Fungsional	6	5	6	4	3	5	5	6	5	3	5	5	5

Gambar 3. 46 Tampilan Rekapitulasi *EMI Survey*

### 3.4.10 Rancangan Tampilan Rekapitulasi *Workplan*

Tampilan ini akan muncul jika pengguna memilih menu *report*, lalu mengklik button Notifikasi. Tampilan di atas menjelaskan tentang rekapitulasi *workplan* dari masing-masing rayon dan tingkat bagian, juga PLN SBB secara keseluruhan. Rancangan tampilan dapat dilihat pada Gambar 3.47.



Gambar 3. 47 Tampilan Rekapitulasi *Workplan*

### 3.4.11 Rancangan Tampilan Notifikasi via SMS

Tampilan di bawah ini menjelaskan tentang notifikasi via SMS. Rekapitulasi *workplan*, selain ditampilkan di dalam web, juga dikirimkan melalui SMS. Rancangan tampilan notifikasi via SMS dapat dilihat pada Gambar 3.48.



Gambar 3. 48 Tampilan Notifikasi via SMS