

**RANCANG BANGUN SISTEM INFORMASI PROJECT TRACKING
BERBASIS WEB MENGGUNAKAN PHP 5 DAN MYSQL 5.6.21
DI DIVISI TEKNIK SIPIL
PADA PT PELABUHAN TANJUNG PRIOK**

TUGAS AKHIR

Diajukan Untuk Memenuhi Sebagian Syarat Penyelesaian Jenjang Diploma Empat (D-4)
Program Studi Sistem Informasi pada Politeknik STMI Jakarta

DISUSUN OLEH

Saputra Ismail

1412001



**POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA
JAKARTA
2016**

POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA

LEMBAR PENGESAHAN

Judul Tugas Akhir : RANCANG BANGUN SISTEM INFORMASI
BERBASIS WEB MENGGUNAKAN PHP 5 DAN
MYSQL 5.6.21 DI DIVISI TEKNIK SIPIL PADA PT
PELABUHAN TANJUNG PRIOK.

Disusun Oleh :
Nama : Saputra Ismail
Nim : 1412001
Program Studi : Sistem Informasi

Telah Diuji Oleh Tim Penguji Sidang Tugas Akhir Program Studi Sistem
Informasi Politeknik STMI Jakarta Kementerian Perindustrian Republik Indonesia
pada hari Jum'at tanggal 4 November 2016.

Jakarta, 21 November 2016

Dosen Pembimbing

Ketua Penguji

Fifi L. Hadianastuti, S.Kom, M.Kes

NIP. 19731016 2005 02 2001

Dosen Penguji

Ulil Hamida, ST, MT

NIP. 19810327 2005 02 2001

Dosen Penguji

Noveriza Yuliasari, S.Si, MT

NIP. 19781121 2009 01 2003

Dr. Ridzky Kramanandita, S.Kom, MT

NIP. 19740302 2002 12 1001

POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA

LEMBAR PERSETUJUAN DOSEN PEMBIMBING

Judul Tugas Akhir : RANCANG BANGUN SISTEM INFORMASI
BERBASIS WEB MENGGUNAKAN PHP 5 DAN
MYSQL 5.6.21 DI DIVISI TEKNIK SIPIL PADA PT
PELABUHAN TANJUNG PRIOK.

Disusun Oleh :

Nama : Saputra Ismail

Nim : 1412001

Program Studi : Sistem Informasi

Tanggal Seminar : 25 Oktober 2016

Tanggal Sidang : 4 November 2016

Tanggal Lulus : 4 November 2016

Jakarta, 21 November 2016

Menyetujui
Dosen Pembimbing

Ahlan Ismono, S.Kom
NIP. 19790107 2006 04 1002

POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA

LEMBAR PERSETUJUAN ASISTEN PEMBIMBING

Judul Tugas Akhir : RANCANG BANGUN SISTEM INFORMASI
BERBASIS WEB MENGGUNAKAN PHP 5 DAN
MYSQL 5.6.21 DI DIVISI TEKNIK SIPIL PADA PT
PELABUHAN TANJUNG PRIOK.

Disusun Oleh :
Nama : Saputra Ismail
Nim : 1412001
Program Studi : Sistem Informasi
Tanggal Seminar : 25 Oktober 2016
Tanggal Sidang : 4 November 2016
Tanggal Lulus : 4 November 2016

Jakarta, 21 November 2016

Menyetujui
Asisten Pembimbing

Fifi L. Hadianastuti, S.Kom, M.Kes
NIP. 19731016 2005 02 2001

**LEMBAR PERSETUJUAN ASISTEN DOSEN PEMBIMBING
POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA**

TANDA PERSETUJUAN ASISTEN PEMBIMBING

Judul Tugas Akhir : RANCANG BANGUN SISTEM INFORMASI
PROJECT TRACKING BERBASIS WEB
MENGUNAKAN PHP 5 DAN MYSQL 5.6.21 DI
DIVISI TEKNIK SIPIL PADA PT PELABUHAN
TANJUNG PRIOK.

Disusun Oleh

Nama : Saputra Ismail
Nim : 1412001
Program Studi : Sistem Informasi

Telah diperiksa dan disetujui untuk diajukan dalam seminar dan sidang Tugas Akhir Politeknik STMI Jakarta.

Jakarta, Oktober 2016

Asisten Pembimbing

Fifi L. Hadianastuti, S.Kom, M.Kes
NIP. 19731016 2005 02 2001

LEMBAR PERSETUJUAN DOSEN PEMBIMBING
POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA

TANDA PERSETUJUAN DOSEN PEMBIMBING

Judul Tugas Akhir : RANCANG BANGUN SISTEM INFORMASI
PROJECT TRACKING BERBASIS WEB
MENGUNAKAN PHP 5 DAN MYSQL 5.6.21 DI
DIVISI TEKNIK SIPIL PADA PT PELABUHAN
TANJUNG PRIOK.

Disusun Oleh

Nama : Saputra Ismail
Nim : 1412001
Program Studi : Sistem Informasi

Telah diperiksa dan disetujui untuk diajukan dalam seminar dan sidang Tugas Akhir Politeknik STMI Jakarta.

Jakarta, Oktober 2016

Dosen Pembimbing

Ahlan Ismono, S.Kom
NIP. 19790107 2006 04 1002

LEMBAR BIMBINGAN PENYUSUNAN TUGAS AKHIR

Nama : Saputra Ismail
 NIM : 1412001
 Judul TA : Rancang Bangun Sistem Informasi *Project Tracking* Berbasis Web
 Menggunakan PHP 5 dan MySQL 5.6.26 di Divisi Teknik Sipil pada
 PT Pelabuhan Tanjung Priok
 Pembimbing : Ahlan Ismono, S.Kom
 Asisten Pembimbing : Fifi L. Hadianastuti, S.Kom, M.Kes

| Tanggal | BAB | Keterangan | Paraf |
|------------|------------|--|-------|
| 15/04/2016 | I | Bimbingan latar belakang, permasalahan dan tujuan. | |
| 18/04/2016 | I dan II | Bimbingan latar belakang dan landasan teori. | |
| 21/04/2016 | I dan II | Bimbingan latar belakang dan landasan teori. | |
| 29/04/2016 | I dan II | Bimbingan latar belakang dan landasan teori. | |
| 10/05/2016 | III dan IV | Bimbingan kerangka penelitian dan sistem berjalan. | |
| 18/05/2016 | III dan IV | Bimbingan kerangka penelitian dan sistem berjalan. | |
| 24/05/2016 | IV | Bimbingan sistem berjalan, <i>usecase</i> dan <i>flowmap</i> | |
| 19/09/2016 | IV - VI | Bimbingan sistem berjalan dan sistem usulan | |
| 28/09/2016 | I - VI | Bimbingan bab I – VI dan demo program | |
| 10/10/2016 | I – VI | Bimbingan bab I – VI dan demo program | |
| 17/10/2016 | I – VI | Bimbingan bab I – VI dan demo program | |
| 18/10/2016 | I - VI | Bimbingan Bab I – VI dan lampiran | |

Mengetahui,
 Ka Prodi
 Sistem Informasi Industri Otomotif

Pembimbing

Drs. Jacob Saragih, MM
 NIP : 19540428 1986 03 1002

Ahlan Ismono, S.Kom
 NIP : 19790107 2006 04 1002

LEMBAR BIMBINGAN PENYUSUNAN TUGAS AKHIR

Nama : Saputra Ismail
 NIM : 1412001
 Judul TA : Rancang Bangun Sistem Informasi *Project Tracking* Berbasis Web Menggunakan PHP 5 dan MySQL 5.6.26 di Divisi Teknik Sipil pada PT Pelabuhan Tanjung Priok
 Pembimbing : Ahlan Ismono, S.Kom
 Asisten Pembimbing : Fifi L. Hadianastuti, S.Kom, M.Kes

| Tanggal | BAB | Keterangan | Paraf |
|------------|------------|--|-------|
| 15/04/2016 | I | Bimbingan latar belakang, permasalahan dan tujuan. | |
| 18/04/2016 | I dan II | Bimbingan latar belakang dan landasan teori. | |
| 21/04/2016 | I dan II | Bimbingan latar belakang dan landasan teori. | |
| 29/04/2016 | I dan II | Bimbingan latar belakang dan landasan teori. | |
| 10/05/2016 | III dan IV | Bimbingan kerangka penelitian dan sistem berjalan. | |
| 18/05/2016 | III dan IV | Bimbingan kerangka penelitian dan sistem berjalan. | |
| 24/05/2016 | IV | Bimbingan sistem berjalan, <i>usecase</i> dan <i>flowmap</i> | |
| 19/09/2016 | IV - VI | Bimbingan sistem berjalan dan sistem usulan | |
| 28/09/2016 | I - VI | Bimbingan bab I – VI dan demo program | |
| 10/10/2016 | I – VI | Bimbingan bab I – VI dan demo program | |
| 17/10/2016 | I – VI | Bimbingan bab I – VI dan demo program | |
| 18/10/2016 | I – VI | Bimbingan Bab I – VI dan lampiran | |

Mengetahui,
 Ka Prodi
 Sistem Informasi Industri Otomotif

Asisten Pembimbing

Drs. Jacob Saragih, MM
 NIP : 19540428 1986 03 1002

Fifi L. Hadianastuti, S.Kom, M.Kes
 NIP : 19731016 2005 02 2001

LEMBAR PERNYATAAN KEASLIAN

Saya yang bertanda tangan di bawah ini:

Nama : Saputra Ismail

Nim : 1412001

Berstatus sebagai mahasiswa Program Studi Sistem Informasi di Politeknik STMI Jakarta Kementerian Perindustrian RI. Dengan ini menyatakan bahwa hasil karya Tugas Akhir yang saya buat dengan judul:

**“RANCANG BANGUN SISTEM INFORMASI PROJECT TRACKING
BERBASIS WEB MENGGUNAKAN PHP 5 DAN MYSQL 5.6.21 DI DIVISI
TEKNIK SIPIL PADA PT PELABUHAN TANJUNG PRIOK”**

- **Dibuat** dan diselesaikan sendiri, dengan menggunakan literatur hasil kuliah, survei lapangan, dibantu oleh dosen pembimbing maupun asisten dosen pembimbing, serta buku-buku maupun jurnal-jurnal ilmiah yang menjadi bahan acuan yang tertera dalam referensi karya Tugas Akhir ini.
- **Bukan** merupakan hasil duplikasi karya tulis yang sudah dipublikasikan atau yang pernah dipakai sebelumnya untuk mendapatkan gelar sarjana di Universitas/Perguruan Tinggi lain, kecuali yang telah disebutkan sumbernya dan dicantumkan pada referensi karya Tugas Akhir ini.
- **Bukan** merupakan karya tulis hasil terjemahan dari kumpulan buku atau jurnal acuan yang tertera dalam referensi karya Tugas Akhir ini.

Jika terbukti saya tidak memenuhi apa yang telah dinyatakan di atas, maka saya bersedia karya Tugas Akhir ini dibatalkan.

Jakarta, 21 November 2016

Yang Membuat Pernyataan,

Saputra Ismail

ABSTRAK

PT Pelabuhan Tanjung Priok adalah perusahaan yang bergerak di bidang jasa kepelabuhanan yang kegiatannya adalah memberikan jasa kepelabuhanan kepada pengguna jasa ekspor dan impor dengan fasilitas dan infrastruktur berskala internasional. Dalam melakukan aktivitasnya, PT Pelabuhan Tanjung Priok terbagi ke dalam beberapa divisi. Salah satu divisi yang dimiliki PT Pelabuhan Tanjung Priok adalah divisi teknik sipil yang bertugas dalam perencanaan pembagunan sekaligus pengawas pekerjaan pembangunan fasilitas dan infrastruktur di Pelabuhan Tanjung Priok. Baiknya pengawasan di dalam suatu proyek akan mendapatkan hasil yang diharapkan. Di beberapa bagian pengawasan pada divisi teknik sipil masih menggunakan cara manual, misalnya dalam memantau *progress* pekerjaan masih menggunakan Microsoft Excel, sehingga menyulitkan pengawas lain bila ingin melihat *progress* pekerjaan tersebut. Lalu kegiatan pengawas selama pengawasan pekerjaan tidak tersimpan dengan baik. Dengan dibuatnya perbaikan Sistem Informasi *Project Tracking* diharapkan dapat meningkatkan efektivitas dan efisiensi proses tersebut. Aplikasi sistem *Project Tracking* ini dikembangkan dengan metodologi *prototype evolutioner* dengan model sistem UML, model data dengan kamus data, dan alat bantu perancangan dengan HIPO dan *flowchart*. Sedangkan untuk *engine* program menggunakan bahasa pemrograman PHP dan *database* menggunakan MySQL. Aplikasi ini diharapkan dapat membantu para pengawas di Divisi Teknik Sipil PT Pelabuhan Tanjung Priok dalam memantau *progress* seluruh pekerjaan di wilayah Pelabuhan Tanjung Priok, memudahkan para pengawas dalam mengumpulkan data untuk membuat addendum bila ada pekerjaan yang membutuhkan Addendum, dan informasi mengenai pekerjaan dapat dilihat dalam aplikasi sistem *Project Tracking*. Untuk penerapan sistem baru terhadap sistem lama, sebaiknya dilakukan secara bertahap dan diperlukan sosialisasi kepada bagian terkait yang akan menggunakan sistem ini. Pemeliharaan aplikasi dilakukan secara berkala, sehingga aplikasi berjalan dengan baik. Tersedianya sumber daya manusia dalam bidang IT yang khusus menangani sistem ini, jika suatu saat terjadi kesalahan atau *error* pada sistem.

Kata kunci : Sistem Informasi, *Project Tracking*, Prototipe Evolutioner, *Unified Modeling Language* (UML), PHP 5, MySQL 5.6.21.

KATA PENGANTAR

Puji syukur penulis panjatkan bagi Tuhan Yang Maha Esa, yang atas berkat dan kasih-Nya sehingga penulis mampu menyelesaikan Tugas Akhir dengan judul **“Rancang Bangun Sistem Informasi *Project Tracking* Berbasis Web Menggunakan PHP 5 dan MySQL 5.6.21 di Divisi Teknik Sipil pada PT Pelabuhan Tanjung Priok”**.

Penulisan Tugas Akhir ini bertujuan untuk memenuhi sebagian syarat penyelesaian jenjang D-4 pada program studi Sistem Informasi pada Politeknik STMI Jakarta.

Terselesaikannya tugas akhir ini tidak terlepas dari bantuan moril maupun materil, bimbingan, dan petunjuk dari berbagai pihak. Oleh karena itu, tak salah kiranya bila penulis mengungkapkan rasa terima kasih yang sebesar-besarnya dan penghargaan kepada:

1. Bapak Dr. Mustofa, ST, MT, Selaku Direktur Politeknik STMI Jakarta.
2. Bapak Drs. Jacob Saragih, MM, Selaku Ketua Jurusan Sistem Informasi Industri Otomotif.
3. Bapak Ahlan Ismono, S.Kom, Selaku dosen pembimbing yang telah memberikan arahan, bimbingan serta penjelasan dalam penyusunan Tugas Akhir.
4. Ibu Fifi L. Hadianastuti, S.Kom, M.Kes, Selaku asisten dosen pembimbing yang telah memberikan arahan, bimbingan serta penjelasan dalam penyusunan Tugas Akhir.
5. Bapak Jumadi dan Ibu Darnani selaku Orang Tua penulis yang selalu mendoakan, memberikan motivasi dan pengorbanannya baik segi moril, materi kepada penulis sehingga Tugas Akhir ini dapat diselesaikan dengan baik.

6. Keluarga Jumadi/Darnani, Aldy Tri Prayoga. Selaku saudara penulis yang selalu memberikan motivasi dan dukungan moril serta doa sehingga Tugas Akhir ini dapat diselesaikan dengan baik.
7. Ibu Isora Dewanti dan Ibu Gita Winda Rahayu selaku pembimbing di PT Pelabuhan Tanjung Priok serta seluruh pekerja khususnya di Divisi Teknik Sipil yang telah membimbing dan memberikan informasi yang berguna kepada penulis dalam penyusunan Tugas Akhir.
8. Seluruh dosen Politeknik STMI Jakarta yang telah memberikan ilmunya guna menambah pengetahuan dan pengalaman yang sangat bermanfaat.
9. Seluruh teman-teman jurusan Sistem Informasi angkatan 2012, khususnya SA21, yang telah memberikan semangat dan motivasi kepada penulis sehingga Tugas Akhir ini dapat diselesaikan dengan baik.
10. Serta semua pihak yang telah membantu penulis dan yang tidak dapat disebutkan satu per satu.

Semoga Tuhan Yang Maha Esa membalas kebaikan dan ketulusan semua pihak yang telah membantu dan penulis berharap semoga Tugas Akhir ini dapat memberikan manfaat dan informasi yang berguna bagi pembaca. Terima kasih.

Jakarta, Oktober 2016

Penulis,
Saputra Ismail

DAFTAR ISI

| | Halaman |
|--|----------|
| HALAMAN JUDUL | |
| LEMBAR PERSETUJUAN DOSEN PEMBIMBING | |
| LEMBAR PERSETUJUAN ASISTEN PEMBIMBING | |
| LEMBAR BIMBINGAN DENGAN DOSEN PEMBIMBING | |
| LEMBAR BIMBINGAN DENGAN ASISTEN PEMBIMBING | |
| LEMBAR PERNYATAAN KEASLIAN | |
| ABSTRAK | vii |
| KATA PENGANTAR | viii |
| DAFTAR ISI | x |
| DAFTAR GAMBAR | xiv |
| DAFTAR TABEL | xvi |
| BAB I PENDAHULUAN | 1 |
| 1.1. Latar Belakang | 1 |
| 1.2. Pokok Permasalahan | 3 |
| 1.3. Tujuan Penelitian | 3 |
| 1.4. Batasan Masalah | 3 |
| 1.5. Manfaat Tugas Akhir | 4 |
| 1.6. Sistematika Penulisan | 4 |
| BAB II LANDASAN TEORI | 6 |
| 2.1. Konsep Dasar Sistem | 6 |
| 2.1.1 Pengertian Sistem | 6 |
| 2.1.2 Karakteristik Sistem | 7 |
| 2.1.3 Klasifikasi Sistem | 10 |
| 2.2 Konsep Dasar Informasi | 11 |
| 2.2.1 Pengertian Informasi | 11 |
| 2.2.2 Siklus Informasi | 11 |
| 2.2.3 Kualitas Informasi | 12 |

| | | |
|-------|---|----|
| 2.3 | Konsep Dasar Sistem Informasi | 13 |
| 2.3.1 | Pengertian Sistem Informasi | 13 |
| 2.3.2 | Komponen Sistem Informasi | 14 |
| 2.4 | Basis Data | 16 |
| 2.4.1 | Pengertian Basis Data | 16 |
| 2.4.2 | Istilah yang Berhubungan dengan Basis Data | 16 |
| 2.4.3 | Komponen Sistem Basis Data | 17 |
| 2.4.4 | Tujuan dan Manfaat Basis Data | 18 |
| 2.4.5 | Pengguna Basis Data | 19 |
| 2.5 | Konsep Dasar Proyek | 19 |
| 2.5.1 | Pengertian Proyek | 19 |
| 2.5.2 | Area Keilmuan Proyek | 20 |
| 2.6 | Konsep Dasar <i>Tracking</i> (Pengawasan) | 22 |
| 2.6.1 | Pengertian Pengawasan | 22 |
| 2.6.2 | Prinsip Pengawasan | 23 |
| 2.6.3 | Tujuan dan Fungsi Pengawasan | 24 |
| 2.6.4 | Jenis Pengawasan | 24 |
| 2.6.5 | Kriteria Pengawasan | 26 |
| 2.6.6 | Cara Pelaksanaan Pengawasan | 27 |
| 2.6.7 | Langkah-Langkah Pengawasan | 27 |
| 2.6.8 | Perencanaan Pengawasan | 28 |
| 2.6.9 | Pemanfaatan Hasil Pengawasan | 28 |
| 2.7 | <i>System Development Life Cycle</i> (SDLC) | 29 |
| 2.7.1 | Model Protipe Secara Umum | 30 |
| 2.7.2 | <i>Evolutionary Prototype</i> | 32 |
| 2.8 | <i>Hypertext Preprocessor</i> (PHP) | 33 |
| 2.8.1 | Kelebihan dan Kelemahan PHP | 34 |
| 2.8.2 | PHP 5 | 34 |
| 2.9 | <i>MySQL</i> | 35 |
| 2.9.1 | Tipe Data <i>MySQL</i> | 36 |
| 2.9.2 | <i>MySQL</i> 5.6.21 | 39 |

| | | |
|----------------|---|-----------|
| 2.10 | <i>Unified Modelling Language (UML)</i> | 40 |
| 2.10.1 | <i>Use Case Diagram</i> | 42 |
| 2.10.2 | <i>Activity Diagram</i> | 43 |
| 2.10.3 | <i>Sequence Diagram</i> | 44 |
| 2.10.4 | <i>Class Diagram</i> | 44 |
| 2.10.5 | <i>Component Diagram</i> | 46 |
| 2.10.6 | <i>Deployment Diagram</i> | 47 |
| 2.11 | Bagan Alir (<i>Flowchart</i>) | 47 |
| BAB III | METODOLOGI PENELITIAN | 50 |
| 3.1. | Metodologi Penelitian | 50 |
| 3.2. | Jenis dan Sumber Data | 50 |
| 3.3. | Metode Pengumpulan Data | 51 |
| 3.4. | Metode yang Dilakukan | 52 |
| 3.5. | Kerangka Penelitian | 52 |
| BAB IV | PENGUMPULAN DAN PENGOLAHAN DATA | 56 |
| 4.1. | Informasi Umum Perusahaan | 56 |
| 4.1.1 | Profil Perusahaan | 56 |
| 4.1.2 | Visi dan Misi Perusahaan | 57 |
| 4.2. | Struktur Organisasi Perusahaan..... | 57 |
| 4.3. | Progress Pekerjaan | 60 |
| 4.3.1 | Berita Acara (BA) Mulai Pekerjaan | 60 |
| 4.3.2 | Laporan Mingguan | 61 |
| 4.3.3 | Monitoring Laporan Mingguan | 62 |
| 4.3.4 | <i>Form</i> Progress Pekerjaan | 63 |
| 4.3.5 | Rekap Laporan | 63 |
| 4.4. | Addendum Pekerjaan | 65 |
| 4.5. | Sistem Berjalan | 66 |
| 4.6. | <i>Use Case Diagram</i> Sistem Saat Ini | 67 |
| BAB V | ANALISIS DAN PEMBAHASAN | 72 |
| 5.1. | Analisis Kebutuhan Sistem | 72 |
| 5.2. | Prosedur <i>Project Tracking</i> Usulan | 73 |

| | | |
|-----------------------|--|------------|
| 5.3. | Analisis dan Perancangan Sistem Usulan | 75 |
| 5.3.1. | <i>Use Case Diagram</i> | 75 |
| 5.3.2. | <i>Activity Diagram</i> | 81 |
| 5.3.3. | <i>Sequence Diagram</i> | 89 |
| 5.3.4. | <i>Deployment Diagram</i> | 96 |
| 5.3.5. | <i>Class Diagram</i> | 96 |
| 5.4. | Perancangan Basis Data | 98 |
| 5.5. | HIPO | 103 |
| 5.6. | <i>Flowchart</i> | 104 |
| 5.7. | Perancangan <i>Interface</i> | 105 |
| 5.8. | Implementasi Sistem | 105 |
| 5.8.1 | Kebutuhan Perangkat Lunak | 109 |
| 5.8.2 | Kebutuhan Perangkat Keras | 109 |
| BAB VI | KESIMPULAN DAN SARAN | 110 |
| 6.1. | Kesimpulan | 110 |
| 6.2. | Saran | 110 |
| DAFTAR PUSTAKA | | 111 |

DAFTAR GAMBAR

| | Halaman |
|--------------|--|
| Gambar II.1 | Karakteristik sistem 7 |
| Gambar II.2 | Siklus Informasi 12 |
| Gambar II.3 | Blok Sistem informasi yang saling berinteraksi 14 |
| Gambar II.4 | Batasan yang mempengaruhi proyek 20 |
| Gambar II.5 | Ilustrasi Model <i>Prototype</i> 31 |
| Gambar II.6 | <i>Evolutionary Prototype Model</i> 33 |
| Gambar II.7 | Klasifikasi Diagram UML 41 |
| Gambar III.1 | Kerangka Penelitian 53 |
| Gambar IV.1 | Struktur Organisasi Keseluruhan PT PTP 58 |
| Gambar IV.2 | Struktur Organisasi Divisi Teknik Sipil 59 |
| Gambar IV.3 | Berita Acara Mulai Pekerjaan 60 |
| Gambar IV.4 | <i>Progress</i> Pekerjaan Laporan Mingguan 61 |
| Gambar IV.5 | Monitoring Laporan Mingguan 62 |
| Gambar IV.6 | <i>Form Progress</i> Pekerjaan 63 |
| Gambar IV.7 | Rekap Laporan <i>Progress</i> Pekerjaan 64 |
| Gambar IV.8 | Berita Acara Addendum Pekerjaan 65 |
| Gambar IV.9 | <i>Flowmap</i> Sistem Monitoring <i>Progress</i> Pekerjaan 66 |
| Gambar IV.10 | <i>Use Case Monitoring Sistem Berjalan</i> 68 |
| Gambar V.1 | <i>Flowmap</i> Sistem <i>Project Tracking</i> Usulan 74 |
| Gambar V.2 | <i>Use Case Diagram</i> Sistem <i>Project Tracking</i> Usulan 76 |
| Gambar V.3 | <i>Activity Diagram Login</i> 81 |
| Gambar V.4 | <i>Activity Diagram</i> Mengelola Data Master 82 |
| Gambar V.5 | <i>Activity Diagram</i> Mengelola Data Pengawas 83 |
| Gambar V.6 | <i>Activity Diagram</i> Membuat Pekerjaan Baru 84 |
| Gambar V.7 | <i>Activity Diagram Input Progress</i> Rencana Kerja 85 |
| Gambar V.8 | <i>Activity Diagram Input Monitoring Progress</i> Pekerjaan 86 |
| Gambar V.9 | <i>Activity Diagram</i> Menginput Data Kegiatan 87 |

| | | |
|-------------|--|-----|
| Gambar V.10 | <i>Activity Diagram</i> Membuat Addendum | 88 |
| Gambar V.11 | <i>Activity Diagram</i> Print Laporan | 89 |
| Gambar V.12 | <i>Sequence Diagram</i> Proses Login | 90 |
| Gambar V.13 | <i>Sequence Diagram</i> Mengelola Data Master | 91 |
| Gambar V.14 | <i>Sequence Diagram</i> Mengelola Data Pengawas | 92 |
| Gambar V.15 | <i>Sequence Diagram</i> Membuat Pekerjaan Baru | 93 |
| Gambar V.16 | <i>Sequence Diagram</i> Input Rencana Progress Pekerjaan | 93 |
| Gambar V.17 | <i>Sequence Diagram</i> Monitoring Progress Pekerjaan | 94 |
| Gambar V.18 | <i>Sequence Diagram</i> Menginput Data Kegiatan | 95 |
| Gambar V.19 | <i>Sequence Diagram</i> Membuat Addendum | 95 |
| Gambar V.20 | <i>Sequence Diagram</i> Print Laporan | 96 |
| Gambar V.21 | <i>Deployment Diagram</i> | 96 |
| Gambar V.22 | <i>Class Diagram</i> | 97 |
| Gambar V.23 | HIPO <i>Project Tracking</i> | 103 |
| Gambar V.24 | <i>Flowchat</i> Sistem <i>Project Tracking</i> | 104 |
| Gambar V.25 | Rancangan <i>Form Login</i> | 105 |
| Gambar V.26 | Rancangan Daftar Menu | 105 |
| Gambar V.27 | Rancangan Tampilan Data Pekerjaan | 106 |
| Gambar V.28 | Rancangan <i>Form</i> Tambah Data Pekerjaan | 106 |
| Gambar V.29 | Rancangan <i>Form</i> Edit Data Pekerjaan | 107 |
| Gambar V.30 | Rancangan Tampilan Monitoring Pekerjaan | 107 |
| Gambar V.31 | Rancangan Tampilan Data Pengawas | 108 |
| Gambar V.32 | Rancangan Tampilan Rekap Laporan | 108 |

DAFTAR TABEL

| | | Halaman |
|-------------|--|---------|
| Tabel II.1 | Tipe Data di MySQL | 36 |
| Tabel II.2 | Simbol-Simbol <i>Use Case Diagram</i> | 42 |
| Tabel II.3 | Simbol-Simbol <i>Activity Diagram</i> | 43 |
| Tabel II.4 | Simbol-Simbol <i>Sequence Diagram</i> | 44 |
| Tabel II.5 | Simbol-Simbol <i>Class Diagram</i> | 45 |
| Tabel II.6 | Simbol-Simbol <i>Component Diagram</i> | 46 |
| Tabel II.7 | Simbol-Simbol <i>Deployment Diagram</i> | 47 |
| Tabel II.8 | Simbol-Simbol <i>Flowchart</i> | 48 |
| Tabel IV.1 | <i>Use Case Description</i> Membuat BA Mulai Kerja | 68 |
| Tabel IV.2 | <i>Use Case Description</i> Melaksanakan Pekerjaan | 68 |
| Tabel IV.3 | <i>Use Case Description</i> Membuat Laporan Mingguan | 69 |
| Tabel IV.4 | <i>Use Case Description</i> Mengevaluasi Laporan | 69 |
| Tabel IV.5 | <i>Use Case Description</i> Membuat Monitoring Laporan | 69 |
| Tabel IV.6 | <i>Use Case Description</i> Mengisi Form Progress | 70 |
| Tabel IV.7 | <i>Use Case Description</i> Melaksanakan Rapat Mingguan | 70 |
| Tabel IV.8 | <i>Use Case Description</i> Melaksanakan Rapat Pembuatan Add . | 70 |
| Tabel IV.9 | <i>Use Case Description</i> Membuat Addendum | 70 |
| Tabel IV.10 | <i>Use Case Description</i> Menjilid Laporan | 71 |
| Tabel IV.11 | <i>Use Case Description</i> Rekap Laporan | 71 |
| Tabel IV.12 | <i>Use Case Description</i> Terima Rekap Laporan | 71 |
| Tabel V.1 | Kebutuhan Sistem | 72 |
| Tabel V.2 | Definisi Aktor <i>Use Case Diagram</i> yang Diusulkan | 76 |
| Tabel V.3 | <i>Use Case Description</i> Login | 77 |
| Tabel V.4 | <i>Use Case Description</i> Mengelola Data Master | 77 |
| Tabel V.5 | <i>Use Case Description</i> Membuat Pekerjaan Baru | 78 |

| | | |
|------------|---|-----|
| Tabel V.6 | <i>Use Case Description</i> Membuat BA Mulai Kerja | 78 |
| Tabel V.7 | <i>Use Case Description Input</i> Rencana <i>Progress</i> Pekerjaan | 78 |
| Tabel V.8 | <i>Use Case Description</i> Evaluasi Laporan Mingguan | 79 |
| Tabel V.9 | <i>Use Case Description</i> Input Monitoring Laporan | 79 |
| Tabel V.10 | <i>Use Case Description</i> Melaksanakan Rapat Mingguan | 79 |
| Tabel V.11 | <i>Use Case Description</i> Melaksanakan Rapat Pembuatan Add .. | 79 |
| Tabel V.12 | <i>Use Case Description</i> Upload Notulen Rapat | 80 |
| Tabel V.13 | <i>Use Case Description</i> Membuat Addendum | 80 |
| Tabel V.14 | <i>Use Case Description</i> Print Rekap Laporan | 80 |
| Tabel V.15 | Tabel tb_pekerjaan | 98 |
| Tabel V.16 | Tabel tb_kegiatan | 99 |
| Tabel V.17 | Tabel tb_bagian | 99 |
| Tabel V.18 | Tabel tb_jabatan | 100 |
| Tabel V.19 | Tabel tb_pengawas | 100 |
| Tabel V.20 | Tabel tb_progress | 100 |
| Tabel V.21 | Tabel tb_addendum | 101 |
| Tabel V.22 | Tabel tb_progadd | 101 |
| Tabel V.23 | Tabel tb_kontraktor | 102 |
| Tabel V.24 | Tabel tb_status | 102 |
| Tabel V.25 | Tabel tb_lokasi | 103 |
| Tabel V.26 | Tabel Pembagian Hak Akses | 104 |

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan dalam bidang teknologi informasi pada saat ini telah membawa manfaat yang sangat penting bagi kemajuan kehidupan suatu bangsa dalam sektor industri dan jasa. Kemajuan dalam hal teknologi informasi ini dapat menyebabkan pertumbuhan ekonomi yang semakin tinggi dan produktifitas industri dan jasa yang semakin meningkat. Perkembangan ini juga dapat dilihat dengan berkembangnya pelabuhan-pelabuhan di Indonesia.

Di dalam jurnal maritim pelabuhan adalah infrastruktur transportasi yang memiliki peran sangat penting di dalam struktur ekonomi, mobilitas barang, jasa dan manusia. Saat ini hampir 90%, volume perdagangan dunia diangkut oleh transportasi laut. Menjadikannya moda yang tak terpisahkan dari perdagangan global. Pelabuhan sebagai simpul dalam sistem pelayaran harus menjamin kelancaran aktivitas logistik dan kontribusi arus material, produk dan informasi di dalam rantai supply yang efektif dan efisien.

Pelabuhan Tanjung Priok adalah salah satu pelabuhan terbesar di Indonesia yang dikelola oleh pemerintah melalui Badan Usaha Milik Negara (BUMN) yaitu PT Pelabuhan Indonesia II (Persero). Pelabuhan Tanjung Priok mempunyai peranan penting untuk menunjang perekonomian, industri dan perdagangan, serta sebagai sentra distribusi barang ke berbagai wilayah di Indonesia. Hal ini membuat pelabuhan Tanjung Priok harus mampu meningkatkan kinerja serta pelayanan agar terus mendorong pertumbuhan ekonomi Indonesia.

Untuk melakukan pelayanan pelabuhan yang baik dan bertaraf internasional, harus ditunjang dengan pembangunan-pembangunan infrastruktur pelabuhan yang mendukung kegiatan dipelabuhan tersebut. Pembangunan pelabuhan melibatkan pihak swasta dan kontraktor-kontraktor BUMN.

Divisi teknik sipil merupakan divisi yang bertanggung jawab dalam pembangunan pelabuhan dan perawatan fasilitas-fasilitas yang berada di dalam pelabuhan. Divisi teknik sipil memiliki 2 sub divisi yaitu sub divisi estimasi & desain dan penilikan. Sub divisi estimasi & desain itu sendiri bertugas untuk melakukan perencanaan pembangunan, baik itu pembangunan gedung, dermaga, lapangan penumpukan maupun fasilitas-fasilitas yang terkait dengan kepelabuhanan. Perencanaannya meliputi gambar rencana pembangunan dan Rencana Anggaran Biaya (RAB). Sub divisi penilikan bertugas untuk mengawasi jalannya proyek pembangunan yang dilakukan kontraktor, pekerjaan mengacu kepada perencanaan yang sudah direncanakan di sub divisi estimasi & desain. Pengawasan dilakukan agar volume pekerjaan, material pekerjaan, dan waktu pekerjaan sesuai dengan yang sudah direncanakan. Cakupan pekerjaan pengawasan yaitu mengawasi jalannya proyek dari awal dimulainya pekerjaan, hingga pekerjaan selesai. Cakupan yang dimaksud meliputi aspek teknis dan non teknis. Teknis disini merupakan kecocokan antara spesifikasi material pekerjaan rencana dengan pelaksanaan dilapangan, dan cara pengerjaannya. Aspek nonteknis sendiri meliputi berkas-berkas administrasi.

Masalah yang sering dihadapi dalam pengawasan adalah kurangnya pengontrolan laporan *progress* pekerjaan, baik laporan mingguan maupun laporan bulanan. Pengontrolan laporan meliputi pengumpulan dan pengecekan laporan. Akibat dari laporan mingguan yang tidak terkontrol dengan baik, bila nanti ditemukan pekerjaan yang pelaksanaannya tidak sesuai dengan rencana, sehingga pekerjaan tersebut membutuhkan *addendum* (pekerjaan tambah atau pekerjaan kurang), ini akan menyulitkan pengawas untuk membuat *addendum* pekerjaan. Laporan mingguan juga digunakan sebagai syarat dalam penagihan yang dimana jumlah penagihan mengacu kepada nilai dari pekerjaan tersebut. Dalam hal ini pengawas harus melihat buku kontrak ketika kontraktor ingin melakukan penagihan, ini menjadi salah satu yang memperlambat pekerjaan dari pengawas.

1.2 Pokok Permasalahan

Permasalahan yang terjadi pada PT Pelabuhan Tanjung Priok adalah sebagai berikut:

1. Laporan mingguan yang tidak terkontrol dengan baik, sehingga menyulitkan pengawas bila ada pekerjaan yang membutuhkan addendum.
2. Sulitnya memantau seluruh *progress* pekerjaan, karena terlalu banyak pekerjaan pembangunan.
3. Pengawas kesulitan dalam menginformasikan nilai kontrak karena harus mencari dan melihat buku kontrak, ketika kontraktor ingin melakukan penagihan.

1.3 Tujuan Penelitian

Adapun tujuan dari penelitian Tugas Akhir ini pada Divisi Teknik Sipil di PT Pelabuhan Tanjung Priok membuat aplikasi *Project Tracking* yang memiliki fungsi sebagai berikut:

1. Membantu pengawas dalam mengontrol laporan mingguan dan mempermudah pengawas dalam membuat addendum pekerjaan.
2. Memudahkan pengawas dalam memantau seluruh *progress* pekerjaan untuk laporan kepada atasan langsung (*manager*).
3. Membantu pengawas dalam menginformasikan nilai kontrak suatu pekerjaan bila ada kontraktor yang ingin melakukan penagihan.

1.4 Batasan Masalah

Agar dalam penulisan Tugas Akhir ini lebih fokus dan lebih terarah, maka perlu diadakan batasan-batasan sebagai berikut:

1. Pengamatan dilakukan pada Divisi Teknik khususnya Sub Divisi Penilikan di PT Pelabuhan Tanjung Priok.
2. Pengamatan ini dilakukan dalam kurun waktu 2 (dua) bulan, sejak April 2014 – Juni 2014
3. Pengamatan dilakukan hanya sebatas menangani masalah dalam *progress* pekerjaan dan laporan *progress* pekerjaan yang masih manual.

1.5 Manfaat Tugas Akhir

Adapun manfaat yang dapat diperoleh dari penelitian Tugas Akhir ini adalah sebagai berikut:

1. Bagi penulis
 - a. Untuk memberikan pengalaman kepada penulis dalam menganalisis suatu sistem dan diharapkan dapat memberikan suatu solusi permasalahan.
 - b. Untuk memberikan kemampuan dalam menggambarkan suatu literatur sistem secara jelas terhadap masalah yang diamati.
2. Bagi perusahaan
Mendapatkan saran yang diharapkan mampu menyelesaikan permasalahan yang terjadi.
3. Bagi pembaca
Dengan pembuatan Sistem Informasi *Project Tracking* ini, pembaca dapat memahami ilmu-ilmu yang telah didapat oleh penulis selama di bangku kuliah, serta menambah wawasan dan pengetahuan penulis.

1.6 Sistematika Penulisan

Agar lebih mempermudah perumusan dan pemecahan masalah yang akan dibahas pada penelitian Tugas Akhir ini, maka penulis menguraikan tahapan-tahapan dalam penyusunan laporan ini sebagai berikut:

BAB I PENDAHULUAN

Bab ini memuat latar belakang penelitian, pokok permasalahan, tujuan penelitian, batasan masalah, manfaat tugas akhir dan sistematika penulisan.

BAB II LANDASAN TEORI

Bab ini membahas tentang pengertian sistem, jenis sistem, pengertian informasi, jenis-jenis informasi, metodologi pengembangan sistem, *Personal Home Page (PHP)*, *MySQL*, *Flowchart* dan *Unified Modelling Language (UML)*.

BAB III METODOLOGI PENELITIAN

Bab ini membahas tentang jenis dan sumber data, metode pengumpulan data, serta langkah-langkah yang akan dilakukan dalam perumusan dan pemecahan masalah termasuk metodologi pengembangan sistem yang digunakan.

BAB IV PENGUMPULAN DAN PENGOLAHAN DATA

Bab ini akan menguraikan tentang hasil pengamatan dalam kerja lapangan yang telah dilakukan, seperti analisis sistem yang berjalan diperusahaan, bentuk dokumen yang terlibat, pembuatan flowchart, dan pembuatan UML.

BAB V ANALISIS DAN PEMBAHASAN

Bab ini berisi analisis data dan analisis permasalahan yang ada di lapangan serta pembahasannya dengan menggunakan *Personal Home Page* (PHP), MySQL, *Flowchart* dan *Unified Modelling Language* (UML).

BAB VI KESIMPULAN DAN SARAN

Bab ini berisikan kesimpulan dari hasil penelitian yang dilakukan, serta mengemukakan saran-saran yang diperlukan perusahaan dan peneliti selanjutnya.

BAB II

LANDASAN TEORI

2.1 Konsep Dasar Sistem

2.1.1 Pengertian Sistem

Kata sistem sendiri berasal dari bahasa Latin “*systēma*” dan bahasa Yunani “*sustēma*” adalah suatu kesatuan yang terdiri komponen atau elemen yang dihubungkan bersama untuk memudahkan aliran informasi, materi atau energi. Istilah sistem merupakan istilah dari bahasa Yunani, *system* yang artinya adalah himpunan bagian atau unsur yang saling berhubungan secara teratur untuk mencapai tujuan bersama. Untuk beberapa sistem yang diintisarikan dari beberapa sumber referensi dari buku dan website, mendefinisikan bahwa sistem :

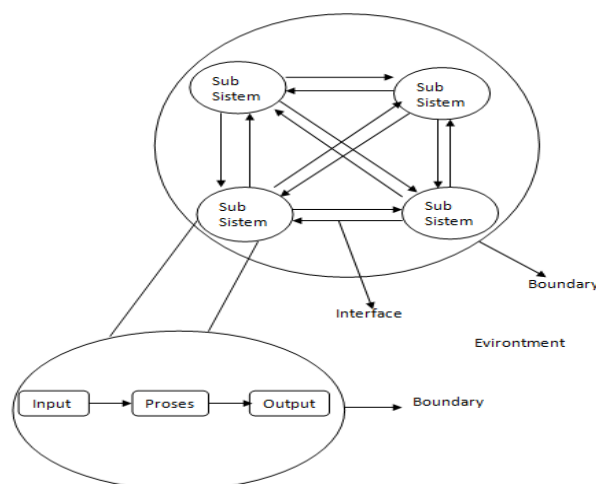
1. Menurut Indrajit (2001) “Mengemukakan bahwa sistem mengandung arti kumpulan-kumpulan dari komponen-komponen yang dimiliki unsur keterkaitan antara satu dengan lainnya”.
2. Menurut Jogiyanto (2005) “Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. sistem ini menggambarkan suatu kejadian-kejadian dan kesatuan yang nyata adalah suatu objek nyata, seperti tempat, benda, dan orang-orang yang betul-betul ada dan terjadi”.
3. Menurut Murdick (1991) “Sistem adalah seperangkat elemen yang membentuk kumpulan atau prosedur-prosedur / bagan-bagan pengolahan yang mencari suatu tujuan bagian atau tujuan bersama dengan mengoperasikan data dan atau barang pada waktu rujukan tertentu untuk menghasilkan informasi dan/atau energi dan/atau barang”.
4. Menurut Davis (1991) “Sistem secara fisik adalah kumpulan dari elemen-elemen yang beroperasi bersama-sama untuk menyelesaikan suatu sasaran”.

5. Menurut Sidharta (1995), “Sistem adalah himpunan dari bagian-bagian yang saling berhubungan yang secara bersama mencapai tujuan”.

Suatu sistem dapat terdiri dari sistem-sistem bagian (*subsystem*). Sebagai contoh, sistem komputer dapat terdiri dari subsistem perangkat keras (*hardware*) dan perangkat lunak (*software*). Masing-masing subsistem dapat terdiri dari subsistem-subsistem yang lebih kecil atau terdiri dari komponen-komponen. Subsistem perangkat keras (*hardware*) dapat terdiri dari alat masukan, alat pemroses, alat keluaran dan simpanan luar. Subsistem-subsistem saling berinteraksi dan saling berhubungan membentuk satu kesatuan sehingga tujuan atau sasaran sistem tersebut dapat tercapai. Interaksi dari subsistem-subsistem sedemikian rupa sehingga dicapai suatu kesatuan yang terpadu atau terintegrasi. (Jogiyanto, 2005)

2.1.2 Karakteristik Sistem

Menurut Jogiyanto (2005) Karakteristik system adalah system yang mempunyai komponen (*components*), batas sistem (*boundary*), lingkungan sistem (*environments*), penghubung (*interface*), masukan (*input*), keluaran (*output*), pengolah (*process*), dan sasaran (*objectives*), seperti yang dapat dilihat pada gambar 2.1 :



Gambar II.1 Karakteristik Sistem

(Sumber : Jogiyanto, 2005)

1. Komponen Sistem (*components*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk satu kesatuan. komponen-komponen sistem atau elemen-elemen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem. Setiap subsistem mempunyai sifat-sifat dari sistem untuk menjalankan suatu fungsi tertentu mempengaruhi proses sistem secara keseluruhan.

2. Batasan Sistem (*boundary*)

Batasan sistem (*boundary*) merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. batasan suatu sistem menunjukkan ruang lingkup dari sistem tersebut.

3. Lingkungan Sistem (*environments*)

Lingkungan luar (*environment*) dari suatu sistem adalah apapun diluar batas sistem yang mempengaruhi operasi. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut. Lingkungan luar yang menguntungkan berupa energi dari sistem dan dengan demikian harus tetap dijaga dan dipelihara. sedang lingkungan luar yang merugikan harus ditahan dan dikendalikan, kalau tidak maka akan mengganggu kelangsungan hidup dari sistem.

4. Penghubung (*interface*)

Penghubung (*interface*) merupakan media penghubung antara satu subsistem dengan subsistem yang lainnya. melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lainnya. Dengan penghubung satu subsistem dapat berintegrasi dengan subsistem yang lainnya membentuk satu kesatuan. Sedangkan menurut Burch dan Grundnitski (dalam Jogiyanto, 2005) desain sistem dapat didefinisikan sebagai penggambaran,

perencanaan, dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah kedalam suatu kesatuan yang utuh dan berfungsi. Desain sistem menentukan bagaimana suatu sistem akan menyelesaikan tahap ini menyangkut mengkonfigurasi dari komponen-komponen perangkat lunak dan perangkat keras dari suatu sistem sehingga setelah instalasi dari sistem akan benar-benar memuaskan rancang bangun yang telah ditetapkan pada akhir tahap analisis sistem. (Jogiyanto, 2005)

5. Masukan (*input*)

Masukan (*input*) sistem adalah energi yang masukan kedalam sistem. Masukan dapat berupa masukan perawatan (*maintenance input*), dan masukan sinyal (*signal input*). *Maintenance input* adalah energi yang dimasukkan supaya tersebut dapat beroperasi. *Signal input* adalah energi yang diproses untuk didapatkan keluaran. Sebagai contoh di dalam komputernya dan data adalah *signal input* untuk diolah menjadi informasi.

6. Keluaran (*output*)

Keluaran (*output*) sistem adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Misalnya untuk sistem komputer, panas yang dihasilkan adalah keluaran yang tidak berguna dan merupakan hasil sisa pembuangan, sedang informasi adalah keluaran yang dibutuhkan.

7. Pengolah (*process*)

Suatu sistem dapat mempunyai suatu bagian pengolah yang akan merubah masukan menjadi keluaran. Suatu sistem produksi akan mengolah masukan berupa bahan baku dan bahan-bahan yang lain menjadi keluaran berupa barang jadi.

8. Sasaran (*objectives*)

Sebuah sistem sudah tentu mempunyai sasaran ataupun tujuan. Dengan adanya sasaran sistem, maka kita dapat menentukan masukan yang dibutuhkan sistem dan keluaran apa yang akan dihasilkan sistem

tersebut dapat dikatakan berhasil apabila mencapai/mengenai sasaran atau pun tujuan.

2.1.3 Klasifikasi Sistem

Sistem dapat diklasifikasikan dari beberapa sudut pandang, diantaranya adalah sebagai berikut :

1. Sistem diklasifikasikan sebagai sistem abstrak (*abstract system*) dan sistem fisik (*physical system*). Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik, misalnya sistem teologia yaitu sistem yang berupa pemikiran-pemikiran hubungan antara manusia dan Tuhan. Sistem fisik merupakan sistem yang ada secara fisik, misalnya sistem komputer, sistem akuntansi, dan sistem produksi.
2. Sistem diklasifikasikan sebagai sistem alamiah (*natural system*) dan sistem buatan manusia (*human made system*). Sistem alamiah adalah sistem yang terjadi melalui proses alam dan tidak dibuat manusia, misalnya sistem perputaran bumi. Sistem buatan manusia adalah sistem yang dirancang oleh manusia dan melibatkan interaksi antara manusia dengan mesin, misalnya sistem informasi akuntansi.
3. Sistem diklasifikasikan sebagai sistem tertentu (*deterministic system*) dan sistem tak tentu (*probabilistic system*). Sistem tertentu beroperasi dengan tingkah laku yang sudah dapat diprediksi. Interaksi diantara bagian-bagiannya dapat dideteksi dengan pasti sehingga keluarannya dapat diramalkan. Sistem komputer adalah contoh dari sistem tertentu yang tingkah lakunya dapat dipastikan melalui program yang dijelankannya. Sistem tak tentu adalah sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilitas.
4. Sistem diklasifikasikan sebagai sistem tertutup (*closed system*) dan sistem terbuka (*open system*). Sistem tertutup adalah system yang tidak berhubungan dan tidak terpengaruh dengan lingkungan luarnya. Sistem ini bekerja secara otomatis tanpa adanya turut campur tangan dari

pihak diluarnya. Pada kenyataannya tidak ada sistem yang benar-benar tertutup, yang ada hanyalah *relatively closed system*. Sistem terbuka adalah sistem yang berhubungan dan terpengaruh dengan lingkungan luarnya. Sistem ini menerima masukan dan menghasilkan keluaran untuk lingkungan luar atau subsistem yang lainnya, sehingga harus memiliki sistem pengendalian yang baik. (Jogiyanto, 2005)

2.2 Konsep Dasar Informasi

2.2.1 Pengertian Informasi

Menurut Mcleod (2004), Informasi adalah data yang telah diolah menjadi bentuk yang memiliki arti bagi si penerima dan bermanfaat bagi pengambilan keputusan saat ini atau mendatang. Menurut Sutabri (2005) Informasi adalah data yang telah diklasifikasikan atau diolah atau diinterpretasikan untuk digunakan dalam proses pengambilan keputusan. Kadir (2002), McFadden, dkk (1999) mendefinisikan informasi sebagai data yang telah diproses sedemikian rupa sehingga meningkatkan pengetahuan seseorang yang menggunakan data tersebut. Menurut Bodnar (2000), “Informasi adalah data yang diolah sehingga dapat dijadikan dasar untuk mengambil keputusan yang tepat”.

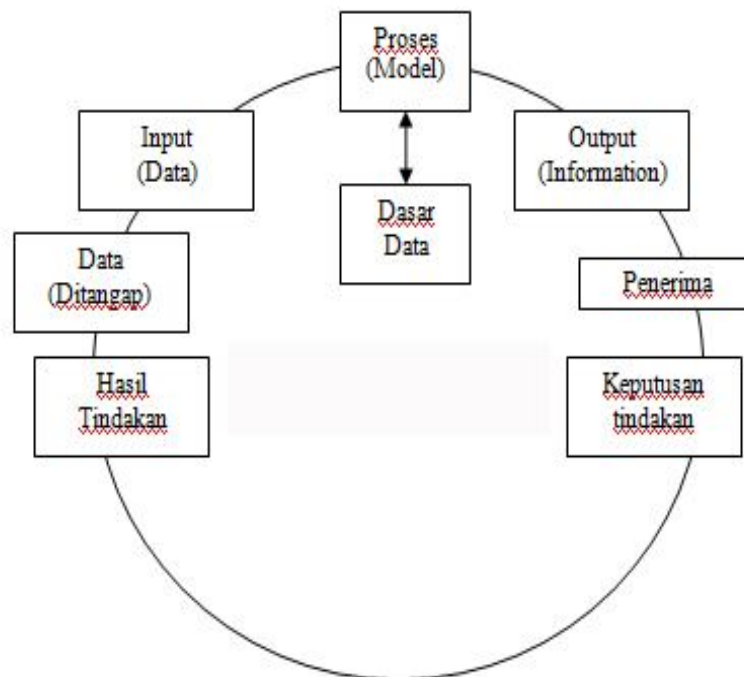
2.2.2 Siklus Informasi

Secara umum informasi dapat didefinisikan sebagai hasil dari pengolahan data dalam suatu bentuk yang lebih berguna dan lebih berarti bagi penerimanya yang menggambarkan suatu kejadian-kejadian yang nyata yang digunakan untuk pengambilan keputusan.

Sumber dari informasi adalah data. Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata. Kejadian-kejadian adalah sesuatu yang terjadi pada saat tertentu. Di dalam dunia bisnis, kejadian-kejadian yang sering terjadi adalah transaksi perubahan dari suatu nilai yang disebut transaksi. Kesatuan nyata adalah berupa suatu obyek nyata seperti tempat, benda dan orang yang betul-betul ada dan terjadi.

Data merupakan bentuk yang masih mentah, belum dapat bercerita banyak sehingga perlu diolah lebih lanjut. Data diolah melalui suatu metode untuk menghasilkan informasi. Data dapat berbentuk simbol-simbol semacam huruf, angka, bentuk suara, sinyal, gambar, dsb.

Data yang diolah melalui suatu model menjadi informasi, penerima kemudian menerima informasi tersebut, membuat suatu keputusan dan melakukan tindakan, yang berarti menghasilkan suatu tindakan yang lain yang akan membuat sejumlah data kembali. Data tersebut akan ditangkap sebagai input, diproses kembali lewat suatu model dan seterusnya membentuk suatu siklus. Siklus informasi ini dapat digambarkan sebagai berikut :



Gambar II.2 Siklus Informasi

(Sumber: Sutabri, 2005)

2.2.3 Kualitas informasi

Kualitas informasi sangat dipengaruhi oleh 3 hal, yaitu:

1. Relevan (*Relevancy*)

Informasi tersebut mempunyai manfaat untuk pemakainya. Relevansi informasi untuk tiap-tiap orang satu dengan yang lainnya berbeda, misalnya informasi mengenai sebab-musabab kerusakan mesin produksi

kepada akuntan perusahaan adalah kurang relevan dan akan lebih relevan bila ditujukan kepada ahli teknik perusahaan. Sebaliknya informasi mengenai harga pokok produksi untuk ahli teknik merupakan informasi yang kurang relevan, tetapi relevan untuk akuntan.

2. Akurat (*Accuracy*)

Informasi harus bebas dari kesalahan-kesalahan dan tidak bias atau menyesatkan. Akurat juga berarti informasi harus jelas mencerminkan maksudnya. Informasi harus akurat karena dari sumber informasi sampai ke penerima informasi kemungkinan banyak terjadi gangguan (*noise*) yang dapat merubah atau merusak informasi tersebut.

3. Tepat Pada Waktu (*timeliness*)

Informasi yang datang pada penerima tidak boleh terlambat, informasi yang sudah usang tidak akan mempunyai nilai lagi, karena informasi merupakan landasan didalam pengambilan keputusan. Bila pengambilan keputusan terlambat, maka dapat berakibat fatal untuk organisasi. Dewasa ini mahalnya nilai informasi disebabkan harus cepatnya informasi tersebut didapat, sehingga diperlukan teknologi-teknologi mutakhir untuk mendapatkan, mengolah dan mengirimkannya.

2.3 Konsep Dasar Sistem Informasi

2.3.1 Pengertian Sistem Informasi

Menurut Gelinas, Oram, dan Wiggins (1990) Sistem informasi adalah suatu sistem buatan manusia yang secara umum terdiri atas sekumpulan komponen berbasis komputer dan manual yang dibuat untuk menghimpun, menyimpan, dan mengelola data serta menyediakan informasi keluaran kepada pemakai.

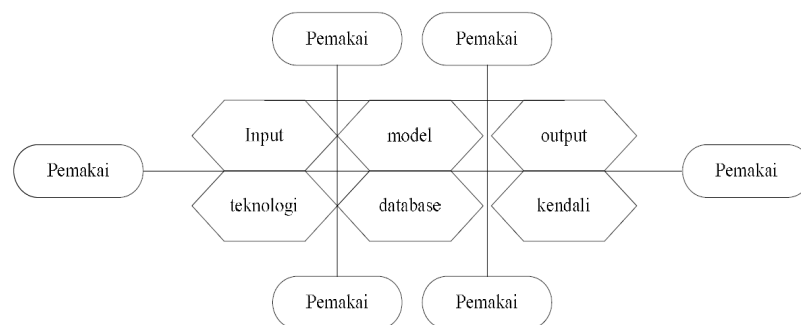
Menurut Nash (2003) yang diterjemahkan oleh La Midjan, menyatakan bahwa Sistem Informasi adalah kombinasi dari manusia, fasilitas atau alat teknologi, media, prosedur dan pengendalian yang bermaksud menata jaringan komunikasi yang penting, proses atas transaksi-transaksi tertentu dan rutin,

membantu manajemen dan pemakai intern dan ekstern dan menyediakan dasar pengambilan keputusan yang tepat. Menurut Wilkinson (1992) sistem informasi adalah suatu kerangka kerja dengan sumber daya (manusia dan komputer) yang dikoordinasikan untuk mengubah masukan (data) menjadi keluaran (informasi) guna mencapai sasaran perusahaan.

Jadi, Sistem Informasi adalah suatu sistem atau proses kumpulan fakta yang di lihat, yang kemudian di catat dalam suatu data, kemudian data tersebut diolah menjadi suatu informasi yang bermanfaat bagi yang membutuhkan.

2.3.2 Komponen Sistem Informasi

Sistem informasi dapat terdiri dari beberapa komponen yang disebut blok bangunan (*building block*), yaitu blok masukan (*input block*), blok model (*modelblock*), blok keluaran (*output block*), dan blok teknologi (*technology block*), blok dasar data (*database block*), dan blok kendali (*control block*). Sebagai suatu sistem, keenam blok tersebut masing-masing saling berinteraksi satu dengan yang lainnya membentuk satu kesatuan untuk mencapai sasarnya.



Gambar II.3 Blok sistem informasi yang saling berinteraksi

(Sumber: Jogiyanto, 2005)

Keterangan gambar blok sistem informasi yang berinteraksi :

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, dapat berupa dokumen-dokumen dasar.

2. Blok Model

Blok ini terdiri dari kombinasi prosedur, logika dan model matematika yang akan memanipulasi data input dan data yang tersimpan di dasar data dengan cara 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*) dari pekerjaan sistem informasi. Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian dari sistem keseluruhan. Teknologi terdiri dari 2 bagian utama, yaitu perangkat lunak (*software*) dan perangkat keras (*hardware*).

5. Blok Basis Data

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

6. Blok Kendali

Agar sistem informasi dapat berjalan sesuai dengan yang diinginkan, maka perlu adanya pengendalian-pengendalian di dalamnya. Banyak hal yang dapat merusak sistem informasi, seperti misalnya bencana alam, api, temperatur, debu, air, kecurangan-kecurangan, kegagalan-

kegagalan sistem itu sendiri, kesalahan-kesalahan, ketidakefisienan dan sabotase. Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah ataupun terlanjur terjadi kesalahan-kesalahan dapat langsung cepat diatasi.

2.4 Basis Data

2.4.1 Pengertian Basis Data

Basis data adalah kumpulan tabel-tabel yang mempunyai kaitan antara satu tabel dengan tabel lainnya sehingga membentuk suatu bangunan data untuk menginformasikan suatu perusahaan atau instansi dalam batasan tertentu.

Menurut “Stephens dan Plew (2000), adalah mekanisme yang digunakan untuk menyimpan informasi atau data. Informasi adalah sesuatu yang kita gunakan sehari-hari untuk berbagai alasan. Dengan basisdata, pengguna dapat menyimpan data secara terorganisasi. Setelah data disimpan, informasi harus mudah diambil. Kriteria dapat digunakan untuk mengambil informasi. Cara data disimpan dalam basisdata menentukan seberapa mudah mencari informasi berdasarkan banyak kriteria. Data pun harus mudah ditambahkan kedalam basisdata, dimodifikasi, dan dihapus”.

Menurut “Siberschatz, dkk. (2002) mendefinisikan basisdata sebagai kumpulan data berisi informasi yang sesuai untuk sebuah perusahaan. System manajemen basisdata (DBMS) adalah kumpulan data yang saling berhubungan dan kumpulan program untuk mengakses data. Tujuan utama system manajemen basisdata adalah menyediakan cara menyimpan dan mengambil informasi basisdata secara mudah dan efisien”.

Menurut “Ramakrishnan dan Gehrke (2003) basis data sebagai kumpulan data, umumnya mendeskripsikan aktivitas satu organisasi atau lebih yang berhubungan”.

2.4.2 Istilah yang berhubungan dengan Basis data

Beberapa hal yang termaksud unsur-unsur dari basis data adalah sebagai berikut:

- a. **Entitas (Tabel)**, Entitas adalah orang, tempat, kejadian atau konsep yang informasinya direkam. Pada bidang kesehatan Entity adalah Pasien, Dokter, Kamar.
- b. **Field**, Setiap entity mempunyai atribut atau sebutan untuk mewakili suatu entity. Seorang siswa dapat dilihat dari atributnya misalnya, NIM, Nama_siswa, Alamat.
- c. **Record, Record** adalah kumpulan isi elemen data (atribut) yang saling berhubungan menginformasikan tentang suatu *entity* secara lengkap. Contoh Kumpulan atribut NIP, Nama, dan alamat berisikan “01001245566”, Sanusi, Jl. Hati suci No 2 Kupang.
- d. **Data Value**, Merupakan data aktual atau informasi yang disimpan di tiap data elemen. Isi atribut disebut nilai data.
- e. **Kunci Elemen Data (Key Data Element)**, Tanda pengenal yang secara unik mengidentifikasi entitas dari suatu kumpulan entitas. Contoh Entitas Mahasiswa yang mempunyai atribut-atribut npm, nama, alamat, tanggal lahir menggunakan Kunci Elemen Data npm.

2.4.3 Komponen Sistem Basis Data

Basis data merupakan sistem yang terdiri atas kumpulan file atau tabel yang saling berhubungan dan Database Management System (DBMS) yang memungkinkan beberapa pemakai untuk mengakses dan manipulasi file-file tersebut (Fathansyah,1999). Dalam Sistem Basis data memiliki beberapa komponen yaitu:

- a. **Perangkat Keras (Hardware)**, Perangkat keras yang biasanya terdapat dalam sistem basis data adalah memori sekunder hardisk.
- b. **Sistem Operasi (Operating System)**, Sistem Operasi (*Operating System*) merupakan program yang mengaktifkan atau mengfungsikan sistem komputer, mengendalikan seluruh sumber daya (*resource*) dan melakukan operasi-operasi dalam komputer. Sistem Operasi yang banyak digunakan seperti: MS-DOS, MS-Windows 95 MS Windows NT, dan Unix.

- c. **Basis data (*Database*)**, Sebuah basis data (*Database*) dapat memiliki beberapa basis data. Setiap basis data dapat berisi atau memiliki sejumlah objek basis data seperti file atau tabel. Database
- d. ***Management System (DBMS)***, Pengolahan basis data secara fisik tidak dilakukan oleh pemakai secara langsung, tetapi ditangani oleh sebuah perangkat lunak yang disebut DBMS yang menentukan bagaimana data disimpan, diubah dan diambil kembali.
- e. **Pemakai (*User*)**, Bagi pemakai dapat berinteraksi dengan basis data dan memanipulasi data dalam program yang ditulis dalam bahasa pemrograman.

2.4.4 Tujuan dan Manfaat Basis Data

Tujuan utama dalam pengolahan data dalam sebuah basis data adalah agar kita dapat memperoleh data yang kita cari dengan mudah dan cepat (Fathansyah,1999). Pemanfaatan basis data dilakukan dengan tujuan yaitu:

- a. **Kecepatan dan kemudahan (*Speed*)**, Pemanfaatan Database memungkinkan kita untuk dapat menyimpan data atau melakukan perubahan (manipulasi) dan menampilkan kembali data tersebut dengan cepat dan mudah, dari pada kita menyimpan data secara manual.
- b. **Efisien ruang penyimpanan (*Space*)**, Dengan Database penggunaan ruang penyimpanan data dapat dilakukan karena kita dapat melakukan penekanan jumlah pengulangan data dengan menerapkan sejumlah pengkodean .
- c. **Keakuratan (*Acuracy*)**, Pemanfatan pengkodean atau pembentukan relasi antar data dengan penerapan aturan atau batasan tipe data dapat diterapkan dalam Database yang berguna untuk menentukan ketidakakuratan pemasukan atau penyimpanan.
- d. **Keamanan (*Security*)**, Dalam sejumlah system pengelola database tidak menerapkan aspek keamanan dalam penggunaan database. Tetapi untuk sistem yang besar dan serius, aspek keamanan juga dapat diterapkan. Dengan begitu kita dapat menentukan siapa yang boleh menggunakan

database dan menentukan jenis operasi-operasi apa saja yang boleh dilakukan.

- e. **Terpeliharanya keselarasan data (*Consitant*)**, Apabila ada perubahan data pada aplikasi yang berbeda maka secara otomatis perubahan itu berlaku untuk keseluruhan.
- f. **Data dapat dipakai secara bersama (*Shared*)**, Data dapat dipakai secara bersama-sama oleh beberapa program aplikasi (secara batch maupun on-line) pada saat bersamaan.
- g. **Dapat diterapkan standarisasi (*standardization*)**, Dengan adanya pengontrolan yang terpusat maka DBA dapat menerapkan standarisasi data yang disimpan sehingga memudahkan pemakaian, pengiriman maupun pertukaran data.

2.4.5 Pengguna Basis Data

a. *System Engineer*

Tenaga ahli yang bertanggung jawab atas pemasangan Sistem Basis Data, dan juga mengadakan peningkatan dan melaporkan kesalahan dari sistem tersebut kepada pihak penjual

b. *Database Administrator (DBA)*

Tenaga ahli yang mempunyai tugas untuk mengontrol sistem basis data secara keseluruhan, meramalkan kebutuhan akan sistem basis data, merencanakannya dan mengaturnya. Adapun tugas dari DBA adalah :

1. Mengontrol DBMS dan software-software
2. Memonitor siapa yang mengakses basis data
3. Mengatur pemakaian basis data
4. Memeriksa *security, integrity, recovery dan concurrency*

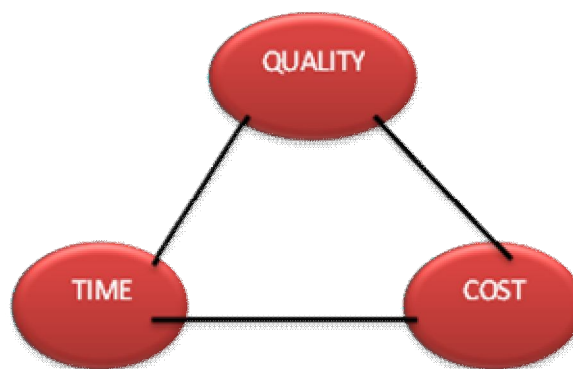
2.5 Konsep Dasar Proyek

2.5.1 Pengertian Proyek

Hal pertama yang perlu kita pahami tentu saja pengertian mengenai proyek itu sendiri. Terdapat beberapa macam pengertian proyek menurut beberapa ahli

yang berbeda-beda. Menurut Suharto (1999), proyek adalah suatu kegiatan sementara (yang berlangsung sementara) dengan alokasi jangka waktu tertentu dengan alokasi sumber daya yang ada dan dimasukkan untuk melakukan fungsi yang telah ditetapkan. Sedangkan menurut PMBOK (*Project Management Body of Knowledge*), *a project is temporary endeavour under taken to create a unique product or service.*

Dalam implementasinya, proyek memiliki tiga buah batasan (*constraints*) yang saling mempengaruhi, yaitu *quality*, *time* dan *cost*. Ketiga batasan ini memiliki pengaruh yang sama kuat terhadap jalannya proyek, sehingga bila digambarkan akan membentuk suatu hubungan segitiga seperti gambar berikut :



Gambar II.4 Batasan yang Mempengaruhi Proyek
(Sumber: Yudhis, 2004)

Agar proyek bisa dilaksanakan dengan baik dan tepat waktu dalam penyelesaiannya, maka seorang manajer proyek harus bisa menyeimbangkan *budget*, *resources*, *schedule*, *scope*, dan *quality* dalam pelaksanaannya. Di dalam melaksanakan proyek, seorang manajer juga menyeimbangkan *knowledge*, yang meliputi *technical aspect* (hampir semua dapat dilakukan ataupun dihitung) dan *social cultural aspect*.

2.5.2 Area Keilmuan Proyek

Menurut PMBOK (*Project Management Body of Knowledge*), terdapat sembilan area keilmuan manajemen proyek. Sembilan area tersebut dibagi ke dalam tiga bagian, yaitu :

- a. Frame : project integration management*
- b. Core : project scope, time, cost, quality*
- c. Supporting : procurement management, HRD management, Risk Management, Communication, Integration Management .*

Scope Management

Salah satu inti (*core*) dari area keilmuan proyek di atas adalah project scope. Untuk memmanajemen *scope* dengan baik, maka harus diperhatikan lima hal berikut, yaitu :

- a. Scope planning*
- b. Scope definition*
- c. Create WBS*
- d. Scope verification*
- e. Scope control*

Time Management

Inti yang lain dari manajemen proyek adalah manajemen waktu. Manajemen waktu yang baik harus dapat melaksanakan tahap-tahap berikut ini :

- a. Activity definition*
- b. Activity Sequencing*
- c. Resource Estimating*
- d. Duration Estimating*
- e. Schedule Development & Control.*

Cost Management

Dalam melakukan manajemen biaya, seorang manajer proyek yang baik harus memperhatikan tiga hal berikut :

1. Cost Estimating

Terdapat tiga cara dalam melakukan estimasi biaya proyek, yaitu :

- a. Estimasi Kasar oleh Pemilik proyek (Parametric estimation, accuracy 15-30%)*
- b. Estimasi Pendahuluan (oleh konsultan perencana – Bottom up estimation, accuracy 10-15%)*
- c. Estimasi (oleh kontraktor Bottom up estimation 5-10%)*

2. *Cost Budgeting*

Pada umumnya, terdapat tiga sumber dana dalam melakukan pembiayaan proyek, yaitu :

- a. Dana Pemerintah (APBN)
- b. Dana Swasta (*Private*)
- c. Kerjasama Pemerintah dan Swasta (*Public Private Partnership –PPP*)

Quality Management

Proyek yang baik tentu harus memperhatikan kualitas dari hasil proyeknya. Agar kualitas proyek yang dihasilkan memang layak, maka perlu dilakukan quality management yang meliputi tiga hal berikut :

1. *Quality Assurance*

Sebuah proyek perlu dijamin kualitasnya. Pihak-pihak yang melakukan penjaminan kualitas proyek adalah : PPK (pejabat pembuat komitmen) – Tim Teknis PU – Konsultan Perencana – Unit Layanan Pengadaan (ULP)

2. *Quality Planning*

Quality planning yang baik harus memperhatikan waktu, biaya, cacat produksi, design/ kontrak, partnering dan juga rantai pasok.

3. *Quality Control*

Dilakukan dengan cara melakukan pemeriksaan terhadap produk yang dihasilkan dari proyek, apakah telah memenuhi standar kualitas yang ditetapkan atau belum.

2.6 Konsep Dasar *Tracking* (Pengawasan)

2.6.1 Pengertian Pengawasan

Pengertian dan definisi pengawasan telah banyak dan panjang lebar dibicarakan para ahli. Berikut pengertian pengawasan (pengawasan) menurut para ahli :

1. Siagian (1970) : mengemukakan bahwa pengawasan sebagai proses pengamatan dari pada pelaksanaan seluruh kegiatan organisasi untuk menjalin agar semua pekerjaan yang sedang dilaksanakan berjalan sesuai dengan rencana yang telah di tentukan sebelumnya .

2. Handoko (1995) : mendefinisikan pengawasan sebagai proses untuk menjamin bahwa tujuan-tujuan organisasi dan manajemen tercapai. Hal ini berkaitan dengan cara-cara membuat kegiatan-kegiatan sesuai dengan yang direncanakan.
3. Sarwoto (1983) : menjelaskan pengawasan adalah kegiatan manajer yang mengusahakan agar pekerjaan-pekerjaan terlaksana sesuai dengan rencana yang ditetapkan dan atau hasil yang dikehendaki.
4. Soekarno (1968) : mendefinisikan pengawasan sebagai suatu proses yang menentukan tentang apa yang harus dikerjakan, agar apa yang diselenggarakan sejalan dengan rencana.

2.6.2 Prinsip Pengawasan

Prinsip pengawasan sangat diperlukan oleh seorang pimpinan atau manajer dalam membandingkan rencana dengan pelaksanaan adalah sebagai berikut :

- a. Prinsip perencanaan merupakan suatu standar atau alat pengukur dari pada suatu pekerjaan yang. Rencana menjadi petunjuk apakah sesuatu pelaksanaan pekerjaan berhasil atau tidak.
- b. Prinsip wewenang merupakan suatu kegiatan pemimpin dalam memberikan kepercayaan kepada bawahan dalam melakukan sistem pengawasan. Wewenang dan instruksi-instruksi yang jelas harus dapat diberikan kepada bawahan, karena berdasarkan pelimpahan wewenang dapat diketahui apakah bawahan sudah melaksanakan tugas-tugasnya dengan baik.
- c. Prinsip tercapainya tujuan. Pengawasan harus ditunjukan kearah tercapainya tujuan, yaitu dengan mengadakan perbaikan (koreksi) untuk menghindarkan penyimpangan-penyimpangan dari rencana yang disusun sebelumnya.
- d. Prinsip efisiensi. Pengawasan dikatakan efisien apabila dapat menghindarkan penyimpangan dari perencanaan, sehingga tidak menimbulkan hal-hal lain yang diluar dugaan.
- e. Prinsip tanggung jawab. Pelaksanaan pengawasan yang efektif dan efisien menurut tanggung jawab penuh dari seorang pimpinan atau manajer terhadap pelaksanaan rencana organisasi.

- f. Prinsip masa depan. Kegiatan pengawasan yang efektif dan efisien harus ditunjukkan kearah pencegahan penyimpangan perencanaan yang akan terjadi baik pada waktu sekarang maupun pada masa yang akan datang.
- g. Prinsip pengawasan langsung. Teknik pengawasan yang paling efektif adalah mengusahakan adanya manajer bawahan yang berkualitas baik. pengawasan itu dilakukan oleh manajer atas dasar bahwa manusia itu sering berbuat salah.
- h. Prinsip penyesuaian dengan organisasi. Pengawasan yang dilakukan hendaknya sesuai dengan struktur organisasi. Manajer dan bawahannya merupakan sarana untuk melaksanakan rencana. Dengan demikian pengawasan yang efektif harus disesuaikan dengan besarnya wewenang manajer, sehingga mencerminkan struktur organisasi.
- i. Prinsip pengawasan individual. Pengawasan harus sesuai dengan kebutuhan manajer. Teknik pengawasan harus ditunjukkan terhadap kebutuhan-kebutuhan akan informasi setiap manajer. Ruang lingkup informasi yang dibutuhkan itu berbeda satu sama lain, tergantung pada tingkat dan tugas manajer.
- j. Prinsip standar. Pengawasan efektif dan efisien dalam organisasi memerlukan standar yang tepat, dan akan dipergunakan sebagai acuan atau alat ukur pelaksanaan dan tujuan yang dicapai.

2.6.3 Tujuan dan Fungsi Pengawasan

- a. Pemeriksaan
- b. Pengujian dan penilaian
- c. Pengurusan
- d. Peninjauan
- e. Pengamatan dan pemantauan
- f. Kunjungan staf
- g. Pembinaan yang dilakukan oleh pimpinan
- h. Pengendalian
- i. Penertiban
- j. Mengusahakan suatu struktur yang terorganisir

- k. Mengusahakan supervise
- l. Mengusahakan informasi yang akurat
- m. Pencapaian hasil
- n. Meningkatkan keterampilan kerja
- o. Mendapatkan atau memperoleh umpan balik

2.6.4 Jenis Pengawasan

Dalam pandangan para ahli terhadap perbedaan-perbedaan yang berhubungan dengan jenis pengawasan sesuai dengan pemahaman mereka masing-masing, diantaranya adalah sebagai berikut :

a. Pengawasan *Preventif*

Usaha-usaha pengawasan yang dilakukan pimpinan terhadap pekerjaan yang dilakukan pegawai dilihat sebelum terjadinya penyimpangan-penyimpangan dalam pelaksanaan pekerjaan sesuai dengan rencana yang telah ditetapkan sebelumnya dapat dilakukan dengan pengawasan *preventif*.

b. Pengawasan *Represif*

Pengawasan yang dilakukan pada akhir kegiatan dikenal dengan pengawasan represif. pengawasan yang dilakukan setelah pekerjaan atau kegiatan dilaksanakan.

c. Pengawasan Langsung

Jenis pengawasan berikutnya adalah pengawasan langsung atau dapat juga dikatakan sebagai kegiatan pengawasan. Pengawasan langsung adalah pengawasan yang dilakukan dengan cara mengunjungi dan melakukan pemeriksaan ditempat terjadinya pelaksanaan pekerjaan, apakah berhubungan dengan pekerjaan yang dilakukan pengawas dan pemimpin organisasi.

d. Pengawasan tidak langsung

Pengawasan tidak langsung dilakukan pemimpin dengan melihat dokumen-dokumen, tanpa langsung melihat ke lapangan tempat dilaksanakannya pekerjaan.

e. Pengawasan Formal

Pengawasan formal sebagai pengawasan resmi oleh lembaga-lembaga pengawasan maupun oleh aparat pengawasan yang mempunyai legalitas tugas dalam bidang pengawasan.

f. Pengawasan Non-formal

Pengawasan nonformal sebagai pengawasan yang dilakukan masyarakat berfungsi sebagai social control. Pengawasan nonformal adalah pengawasan yang dilakukan oleh masyarakat, baik langsung maupun tidak langsung.

g. Pengawasan Administratif

Pengawasan Administratif sebagai kegiatan yang melihat pekerjaan dari ketatalaksanaan pelaksanaan program kerja organisasi atau perusahaan. Pengawasan administratif adalah pengawasan yang menilai perbuatan keseluruhan dari organisasi atau bidang-bidang bagaiannya.

h. Pengawasan Operatif

Pengawasan operatif adalah mengukur efisiensi perbuatan dari waktu ke waktu yang ditunjukkan pada bidang-bidang yang memerlukan tindakan pembetulan dan perbaikan.

i. Pengawasan Intern

Kegiatan yang dilakukan oleh orang yang berada dalam organisasi dikenal dengan pengawasan intern atau pengawasan maupun pimpinan orang tersebut.

j. Pengawasan ekstren

Pengawasan ekstren atau disebut juga dengan pengawasan dari masyarakat ataupun dari pejabat pengawasan fungsional diluar organisasi. Pengawasan ekstren merupakan pengawasan yang dilakukan oleh orang-orang yang ada diluar organisasi.

2.6.5 Kriteria Pengawasan

Kriteria yang dipakai sebagai dasar pengawasan adalah yang berkaitan dengan hal-hal sbb :

- a. Estimasi hasil pekerjaan, sampai seberapa jauh pelaksanaan kegiatan pada saat pengawasan dilakukan apakah pelaksanaan tersebut sesuai dengan rencana yang ditetapkan.
- b. Estimasi penggunaan dana yang telah dikeluarkan, sampai seberapa besar dana yang telah dialokasikan dan apakah pengeluaran dana tersebut sesuai dengan rencana yang telah ditetapkan.
- c. Estimasi pengeluaran tiap periode kegiatan. Umumnya lama kegiatan ini adalah tiap bulan. Jadi berapa besar pengeluaran tiap bulannya. Dari data ini dapat dilihat apakah pengeluaran tersebut sesuai dengan rencana apakah jumlah pengeluaran tersebut sesuai dengan rencana dan apakah jumlah pengeluaran tersebut cukup rasional bila dibandingkan dengan volume pekerjaan.
- d. Estimasi penyusutan alat-alat yang dipakai. Sebab besar-kecilnya penyusutan akan mempengaruhi perhitungan kebutuhan biaya.
- e. Estimasi efisiensi alokasi sumber daya ; misalnya apakah sumber daya yang telah dilaksanakan dengan efisien atau apakah produktivitas tenaga kerja telah dicapai untuk tujuan efisiensi tersebut.

2.6.6 Cara Pelaksanaan Pengawasan

Cara pelaksanaan pengawasan dapat dilakukan melalui dua cara yaitu secara langsung dan cara tidak langsung. Kedua cara tersebut dilakukan dengan seperangkat kegiatan pengawasan yang sama yaitu kegiatan yang berkaitan dengan mengumpulkan, mencatat, mengolah informasi dan pelaksanaan.

a. Pengamatan Langsung

Pengertian pengamatan langsung adalah pengamatan yang dilakukan dengan cara mengamati langsung, dan dapat mengumpulkan secara bebas informasi yang dilakukan.

b. Pengamatan tidak langsung

Cara ini menghendaki petugas pengawasan tidak perlu terjun langsung ke lokasi, tetapi penggalan data dilakukan dengan cara mengirim seperangkat daftar isian untuk diisi oleh orang lain dilokasi penelitian. Cara tidak

langsung ini juga dapat dilakukan dengan mengumpulkan data melalui laporan-laporan yang dibuat oleh pimpinan (manajer).

2.6.7 Langkah-langkah Pengawasan

Pengawasan dilakukan dengan langkah-langkah sebagai berikut:

1. Menyusun Rancangan Pengawasan
 - a. Tujuan
 - b. Sasaran/Aspek yang akan dimonitor
 - c. Faktor Pendukung dan Penghambat
 - d. Pendekatan, Teknik, dan Instrumen
 - e. Waktu dan Jadwal Pengawasan
 - f. Biaya
2. Melaksanakan Pengawasan
3. Menyusun dan Melaporkan hasil kepada pihak pengelola/penyelenggaran program.

2.6.8 Perencanaan Pengawasan

Perencanaan pengawasan meliputi aspek-aspek sebagai berikut:

- a. Perumusan tujuan pemantauan, berisi informasi tentang apa yang diinginkan, untuk siapa, dan untuk kepentingan apa.
- b. Penetapan sasaran pemantauan, apa yang akan dijadikan sebagai objek pemantauan. Contoh: kesulitan belajar dan jenis-jenis kesalahan konsepsi matematika yang masih dialami para siswa.
- c. Penjabaran data yang dibutuhkan pemantauan, penjabaran dari sasaran. Contoh: guru perlu dapat memilah kesalahan karena kecerobohan atau ketidaktelitian dengan kesalahan karena kurang memahami makna dan cara penyelesaian soal.
- d. Penyiapan metode/alat pemantauan sesuai dengan sifat objek dan sumber atau jenis datanya. Contoh: guru menyiapkan tugas berupa soal yang harus dikerjakan secara mandiri oleh setiap siswa. Kertas atau buku yang berisi pekerjaan siswa akan menjadi sumber data utama. Dari pekerjaan itulah guru

akan mengidentifikasi bagian mana dari bahan ajar matematika yang baru saja diajarkan, tetapi masih banyak belum dipahami, jenis kesalahan apa yang pada umumnya masih dilakukan oleh siswa.

- e. Perancangan analisis data pemantauan dan pemaknaannya dengan berorientasi pada tujuan pemantauan. Contoh: dalam analisis nantinya akan dilakukan pengelompokan jenis-jenis kesalahan untuk menjadi bahan pertimbangan dalam memberikan pembelajaran remedial, terutama pada sejumlah siswa yang memang masih mengalami kesulitan memahami pelajaran.

2.6.9 Pemanfaatan Hasil Pengawasan

Data yang telah terkumpul dari hasil pemantauan harus secepatnya diolah dan dimaknai sehingga dapat segera diketahui apakah tujuan pelaksanaan program tercapai atau tidak. Pemaknaan hasil pemantauan ini menjadi dasar untuk merumuskan langkah-langkah berikutnya dalam pelaksanaan program. Jika perlu perubahan, perubahan apa dan bagaimana rancangannya. Jika tidak ada hal mendasar yang memerlukan perubahan, mungkin masih dapat pula dirumuskan bagian mana dari rancangan program yang memerlukan perhatian lebih banyak sehingga aspek-aspek program yang sudah baik dapat menjadi lebih baik lagi.

2.7 *System Development Life Cycle (SDLC)*

System Development Life Cycle adalah proses mengembangkan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan orang untuk mengembangkan sistem-sistem perangkat lunak sebelumnya berdasarkan *best practice* atau cara-cara yang sudah teruji baik (Rosa dan Shalahuddin, 2011).

Menurut Rosa dan Shalahuddin (2011), tahapan-tahapan yang ada pada SDLC secara global adalah sebagai berikut:

1. Inisiasi (*initiation*)

Tahap ini biasanya ditandai dengan pembuatan proposal proyek perangkat lunak.

2. Pengembangan konsep sistem (*system concept development*)
Mendefinisikan lingkup konsep termasuk dokumen lingkup sistem analisis manfaat biaya, manajemen rencana dan pembelajaran kemudahan sistem.
3. Perencanaan (*planning*)
Mengembangkan rencana manajemen proyek dan dokumen perencanaan lainnya. Menyediakan dasar untuk mendapatkan sumber daya (*resources*) yang dibutuhkan untuk memperoleh solusi.
4. Analisis kebutuhan (*requirements analysis*)
Menganalisis kebutuhan pemakai sistem perangkat lunak (*user*) dan mengembangkan kebutuhan *user*. Membuat dokumen kebutuhan fungsional.
5. Desain (*design*)
Mentransformasikan kebutuhan detail menjadi kebutuhan yang sudah lengkap, dokumen desain sistem fokus pada bagaimana dapat memenuhi fungsi-fungsi yang dibutuhkan.
6. Pengembangan (*development*)
Mengonversi desain ke sistem informasi yang lengkap termasuk bagaimana memperoleh dan melakukan instalasi lingkungan sistem yang dibutuhkan, membuat basis data dan mempersiapkan prosedur kasus pengujian, mempersiapkan berkas atau file pengujian, pengodean, pengompilasian, memperbaiki dan membersihkan program serta peninjauan pengujian.
7. Integrasi dan pengujian (*integration and test*)
Mendemonstrasikan sistem perangkat lunak bahwa telah memenuhi kebutuhan yang dispesifikasikan pada dokumen kebutuhan fungsional. Dengan diarahkan oleh staf penjamin kualitas (*quality assurance*) dan *user* sehingga menghasilkan laporan analisis pengujian.
8. Implementasi (*implementation*)
Termasuk pada persiapan implementasi, implementasi perangkat lunak pada lingkungan produksi (lingkungan luar *user*) dan

menjalankan resolusi dari permasalahan yang teridentifikasi dari fase integrasi dan pengujian.

9. Operasi dan pemeliharaan (*operations and maintenance*)

Mendeskripsikan pekerjaan untuk mengoperasikan dan memelihara sistem informasi pada lingkungan produksi (lingkungan pada *user*), termasuk implementasi akhir dan masuk pada proses peninjauan.

10. Disposisi (*disposition*)

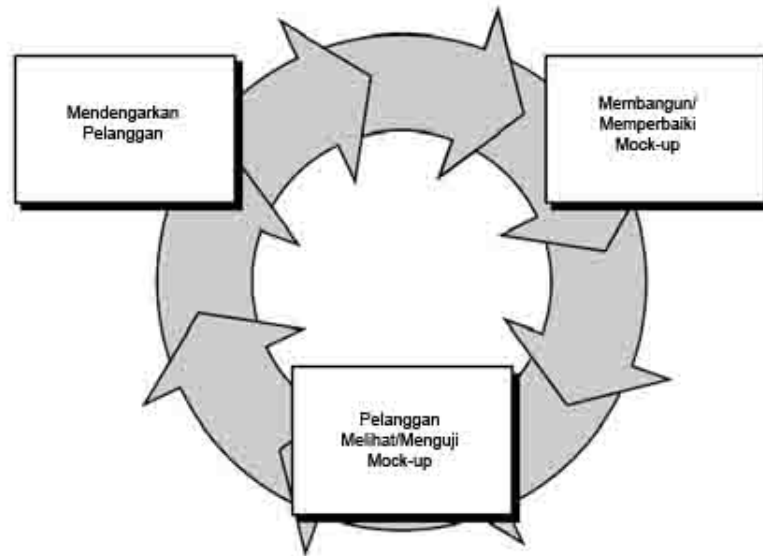
Mendeskripsikan aktifitas akhir dari pengembangan sistem dan membangun data yang sebenarnya sesuai dengan aktifitas *user*.

2.7.1 Model Prototipe Secara Umum

Model prototipe dapat digunakan untuk menyambungkan ketidakpahaman pelanggan mengenai hal teknis yang memperjelas spesifikasi kebutuhan yang diinginkan pelanggan kepada pengembang perangkat lunak (Rosa dan Shalahuddin, 2011).

Tahapan-tahapan pada model prototipe (*prototype model*) adalah sebagai berikut (Rosa dan Shalahuddin, 2011):

1. Mengumpulkan kebutuhan pelanggan terhadap perangkat lunak yang akan dibuat.
2. Membuat prototipe agar pelanggan lebih terbayang dengan apa yang sebenarnya diinginkan. Program prototipe biasanya merupakan program yang belum jadi.
3. Program prototipe selanjutnya dievaluasi oleh pelanggan atau *user* sampai ditemukan spesifikasi yang sesuai dengan keinginan pelanggan atau *user*.



Gambar II.5 Ilustrasi Model Prototipe

Sumber: Rosa dan Shalahuddin (2011)

Mock-up adalah sesuatu yang digunakan sebagai model desain yang digunakan untuk mengajar, demonstrasi, evaluasi desain, promosi atau keperluan lain. Sebuah *mock-up* disebut sebagai prototipe perangkat lunak jika menyediakan atau mampu mendemonstrasikan sebagian besar fungsi sistem perangkat lunak dan memungkinkan pengujian desain sistem perangkat lunak. Iterasi terjadi pada pembuatan prototipe sampai sesuai dengan keinginan pelanggan atau *user* (Rosa dan Shalahuddin, 2011).

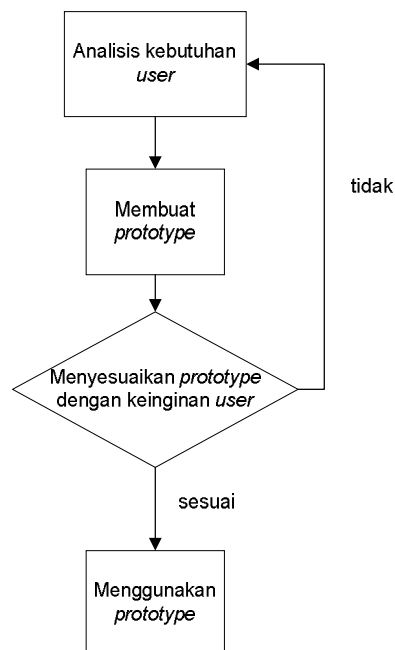
Kelemahan model prototipe adalah sebagai berikut (Rosa dan Shalahuddin, 2011):

1. *User* dapat sering mengubah-ubah atau menambah spesifikasi kebutuhan karena menganggap aplikasi sudah dengan cepat dikembangkan, karena adanya iterasi ini dapat menyebabkan pengembang banyak mengalah dengan *user* karena perubahan atau penambahan spesifikasi kebutuhan perangkat lunak.
2. Pengembang lebih sering mengambil kompromi dengan pelanggan untuk mendapatkan prototipe dengan waktu yang cepat sehingga pengembang lebih sering melakukan segala cara (tanpa idealis) guna menghasilkan prototipe untuk didemonstrasikan. Hal ini dapat

menyebabkan kualitas perangkat lunak yang kurang baik atau bahkan menyebabkan iteratif tanpa akhir.

2.7.2 *Evolutionary Prototype*

Evolutionary prototype yaitu, *prototype* yang secara terus menerus dikembangkan hingga *prototype* tersebut memenuhi fungsi dan prosedur yang dibutuhkan oleh sistem. Pada pendekatan evolusioner, suatu *prototype* dibangun berdasarkan pada kebutuhan dan pemahaman secara umum. *Prototype* kemudian diubah dan dievolusikan dari pada dibuang. *Prototype* yang dibuang biasanya digunakan dengan aspek sistem yang dimengerti secara luas dan dibangun atas kekuatan tahapan *evolutionary prototype* (McLeod, 2004).



Gambar II.6 *Evolutionary Prototype Model*

Sumber: McLeod (2004)

1. Analisis kebutuhan *user*, pengembang dan *user* atau pemilik sistem melakukan diskusi dimana *user* atau pemilik sistem menjelaskan kepada pengembang tentang kebutuhan sistem yang mereka inginkan.
2. Membuat *prototype*, pengembang membuat *prototype* dari sistem yang telah dijelaskan oleh *user* atau pemilik sistem.

3. Menyesuaikan *prototype* dengan keinginan *user* atau pemilik sistem, pengembang menanyakan kepada *user* atau pemilik sistem tentang *prototype* yang sudah dibuat, apakah sesuai atau tidak dengan kebutuhan sistem.
4. Menggunakan *prototype*, sistem mulai dikembangkan dengan *prototype* yang sudah dibuat.

2.8 *Hypertext Preprocessor (PHP)*

PHP adalah bahasa pemrograman web atau *scripting language* yang didesain untuk web. Awalnya PHP merupakan kependekan dari *Personal Home Page*. PHP dibuat pertama kali oleh Rasmus Lerdorf pada tahun 1994, yang pada awalnya dibuat untuk menghitung jumlah pengunjung pada *homepage*-nya. Selanjutnya diganti menjadi FI ("*Forms Interpreter*"). Sejak versi 3.0, nama bahasa ini diubah menjadi "*PHP: Hypertext Preprocessor*" dengan singkatannya "PHP". PHP versi terbaru adalah versi ke -5 (Septian, 2011).

2.8.1 Kelebihan dan Kelemahan PHP

Beberapa kelebihan PHP dari bahasa pemrograman *web*, antara lain:

1. Bahasa pemrograman PHP adalah sebuah bahasa *script* yang tidak melakukan sebuah kompilasi dalam penggunaannya.
2. *Web Server* yang mendukung PHP dapat ditemukan dimana-mana dari mulai *apache*, *IIS*, *Lighttpd*, hingga *Xitami* dengan konfigurasi yang relatif mudah.
3. Dalam sisi pengembangan lebih mudah, karena banyaknya milis-milis dan *developer* yang siap membantu dalam pengembangan.
4. Dalam sisi pemahaman, php adalah bahasa *scripting* yang paling mudah karena referensi yang banyak.

Beberapa kelemahan PHP dari bahasa pemrograman lain, antara lain:

1. Tidak detail untuk pengembangan skala besar
2. Tidak memiliki sistem pemrograman berorientasi objek yang sesungguhnya (sampai versi 4).

3. Tidak bisa memisahkan antara tampilan dengan *logic* dengan baik.
4. PHP memiliki kelemahan *security* tertentu apabila *programmer* tidak jeli dalam memperhatikan isi konfigurasi PHP.

2.8.2 PHP 5

PHP 5 merupakan kelanjutan dari evolusi yang berjalan pada PHP. Walaupun pada PHP 4 sudah banyak *library* yang ditambahkan, PHP 5 menawarkan peningkatan dari fungsionalitas dan penambahan beberapa fitur, antara lain:

1. Peningkatan dari kemampuan pemrograman berorientasi objek.
2. *Exception handling*, yang menstandarisasi logika atas pemberitahuan kesalahan pemrograman.
3. Peningkatan terhadap penanganan *string*.
4. Peningkatan dukungan terhadap *XML* dan *Web Service*, yang menggunakan *libxml2*.
5. Dukungan terhadap *SQLite database server*.
6. PHP versi 5 dapat membuat file swf dan applet java.

Salah satu kelebihan PHP5 adalah PHP5 dapat menangani OOP (*Object Oriented Programming*) lebih baik dari pendahulunya.

2.9 MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (*database management system*) atau DBMS yang *multithread*, *multiuser*, dengan sekitar 6 juta instalasi di seluruh dunia. *MySQL AB* membuat *MySQL* tersedia sebagai perangkat lunak gratis di bawah lisensi GNU *General Public Licence* (GPL), tetapi mereka juga menjual di bawah lisensi komersial untuk kasus-kasus dimana penggunaannya tidak cocok dengan penggunaan GPL (Solichin, 2010).

Fitur-fitur *MySQL* antara lain (Solichin, 2010):

1. *Relational Database System*

Seperti halnya *software database* lain yang ada di pasaran, *MySQL* termasuk RDBMS (*Relational DataBase Management System*).

2. Arsitektur *Client-Server*

MySQL memiliki arsitektur *client-server* dimana *server database MySQL* terinstal di *server*. *Client MySQL* dapat berada di komputer yang sama dengan *server* dan dapat juga di komputer lain yang berkomunikasi dengan server melalui jaringan bahkan internet.

3. Mengenal perintah SQL standar

SQL (*Structured Query Language*) merupakan suatu bahasa standar yang berlaku di hampir semua *software database*. *MySQL* mendukung SQL versi SQL:2003.

4. Mendukung *Sub Select*

Mulai versi 4.1 *MySQL* telah mendukung *select* dalam *select (sub select)*.

5. Mendukung *Views*

MySQL mendukung *views* sejak versi 5.0

6. Mendukung *Stored Prosedured (SP)*

MySQL mendukung SP sejak versi 5.0

7. Mendukung *Triggers*

MySQL mendukung *trigger* pada versi 5.0 namun masih terbatas. Pengembang *MySQL* berjanji akan meningkatkan kemampuan trigger pada versi 5.1.

8. Mendukung *replication*.

9. Mendukung transaksi.

10. Mendukung *foreign key*.

11. Tersedia fungsi GIS.

12. *Free* (bebas diunduh)

13. Stabil dan tangguh

14. Fleksibel dengan berbagai pemrograman

15. *Security* yang baik

16. Dukungan dari banyak komunitas

17. Perkembangan *software* yang cukup cepat.

2.9.1 Tipe Data MySQL

Tabel II.1 Tipe Data MySQL

| Tipe Data | Keterangan |
|-----------|--|
| Tinyint | Normal = -128 s/d 127, Unsigned = 0 s/d 255. Maksimal digit angka ditentukan pada parameter size didalam kurung. |
| Smallint | Normal = -32768 s/d 32767, Unsigned = 0 s/d 65535. Maksimal digit angka ditentukan pada parameter size didalam kurung. |
| Mediumint | Normal = -8388608 s/d 8388607, Unsigned = 0 s/d 16777215. Maksimal digit angka ditentukan pada parameter size didalam kurung. |
| Int | Normal = -2147483648 s/d 2147483647, Unsigned = 0 s/d 4294967295. Maksimal digit angka ditentukan pada parameter size didalam kurung. |
| Bigint | Normal = -9223372036854775808 s/d 9223372036854775807, Unsigned = 0 s/d 18446744073709551615. Maksimal digit angka ditentukan pada parameter size didalam kurung. |
| Float | Angka kecil dengan desimal. Maksimal digit angka ditentukan pada parameter size dan maksimal digit desimal ditentukan pada parameter d . |
| Double | Angka besar dengan desimal. Maksimal digit angka ditentukan pada parameter size dan maksimal digit desimal ditentukan pada parameter d . |
| Decimal | Tipe DOUBLE yang disimpan sebagai string, memungkinkan untuk poin desimal tetap. Maksimal digit angka ditentukan pada parameter size dan maksimal digit desimal ditentukan pada parameter d . |
| Char | <ul style="list-style-type: none"> - Menangani tipe data string tetap/fixed (bisa berisi huruf, angka dan special karakter). - Panjang string ditentukan dengan nilai parameter size |

Tabel II.1 Tipe Data *MySQL* (Lanjutan)

| Tipe Data | Keterangan |
|------------------|--|
| | <p>didalam kurung.</p> <ul style="list-style-type: none"> - Kelebihan karakter akan dipotong sesuai panjang yang telah ditentukan. - Dapat menyimpan sampai maksimal 255 karakter. |
| Varchar | <ul style="list-style-type: none"> - Menangani tipe data string variabel (bisa berisi huruf, angka dan special karakter). Panjang string ditentukan dengan nilai parameter size didalam kurung. - Kelebihan karakter akan dipotong sesuai panjang yang telah ditentukan. - Dapat menyimpan sampai maksimal 255 karakter. - Jika dimasukkan nilai yang lebih dari 255 akan otomatis di konvert ke tipe TEXT parameter size didalam kurung. |
| Tinytext | Menangani tipe data string dengan panjang maksimal 255 karakter. |
| Text | Menangani tipe data string dengan panjang maksimal 65.535 karakter. |
| Blob | Untuk BLOBs (Binary Large OBjects), mampu menangani sampai 65,535 bytes data |
| Mediumtext | Menangani tipe data string dengan panjang maksimal 16,777,215 karakter |
| Mediumblob | Untuk BLOBs (Binary Large OBjects). mampu menangani sampai 16,777,215 bytes data |
| Longtext | Menangani tipe data string dengan panjang maksimal 4,294,967,295 karakter |
| Longblob | Untuk BLOBs (Binary Large OBjects). mampu menangani sampai 4,294,967,295 bytes data |
| Enum(x,y,z,etc.) | Mengijinkan kita memasukkan beberapa nilai yang mungkin didalam sebuah daftar. Kita bisa membuat hingga 65535 nilai dalam daftar ENUM. Jika nilai yang dimasukkan yang tidak ada dalam daftar, maka nilai kosong |

Tabel II.1 Tipe Data *MySQL* (Lanjutan)

| Tipe Data | Keterangan |
|-----------|--|
| | akan disisipkan. Catatan: Nilai-nilai diurutkan berdasarkan urutan saat kita memasukkan mereka. Kita bisa memasukkan kemungkinan nilai dalam format ini: ENUM ('X', 'Y', 'Z') |
| Set | Mirip dengan ENUM, namun SET bisa berisi hingga 64 daftar item dan dapat menyimpan lebih dari satu pilihan. |
| Date | <ul style="list-style-type: none"> - Sebuah tanggal. - Format: YYYY-MM-DD <p>Note: Range yang disupport adalah dari '1000-01-01' sampai '9999-12-31'</p> |
| Datetime | <ul style="list-style-type: none"> - Sebuah kombinasi tanggal dan waktu. - Format: YYYY-MM-DD HH:MI:SS <p>Note: Range yang disupport adalah dari '1000-01-01 00:00:00' sampai '9999-12-31 23:59:59'</p> |
| Timestamp | <ul style="list-style-type: none"> - Sebuah Timestamp - Nilai TIMESTAMP disimpan sebagai detik angka sejak masa Unix ('1970-01-01 00:00:00' UTC). - Format: YYYY-MM-DD HH:MI:SS <p>Note: Range yang disupport adalah dari '1970-01-01 00:00:01' UTC sampai '2038-01-09 03:14:07' UTC</p> |
| Time | <ul style="list-style-type: none"> - Sebuah waktu - Format: HH:MI:SS <p>Note: Range yang disupport adalah dari '-838:59:59' sampai '838:59:59'</p> |
| Year | <ul style="list-style-type: none"> - Sebuah tahun dengan format dua digit atau empat digit. <p>Note: Nilai yang diijinkan untuk format 4 digit adalah 1901 to 2155. Nilai yang diijinkan pada format 2 digit adalah 70 to 69, merupakan representasi dari tahun 1970 sampai 2069</p> |

Sumber : nulis-ilmu.com (2016)

2.9.2 MySQL 5.6.21

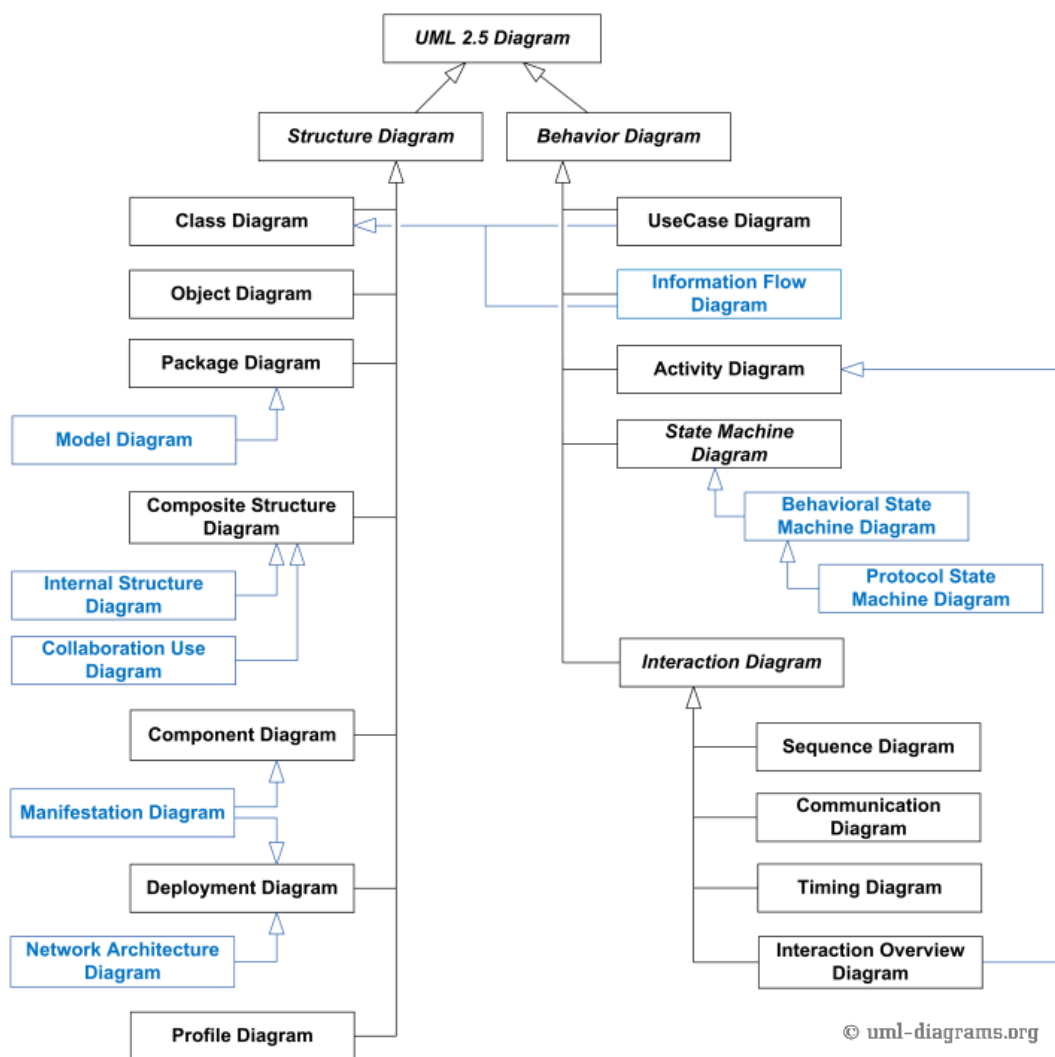
Pada mysql versi 5.6.21 memiliki beberapa tambahan atau perubahan fungsi dan memperbaiki fungsi-fungsi yang memiliki *bugs*, beberapa diantaranya sebagai berikut:

- Replikasi: Variabel *simplified_binlog_gtid_recovery* baru dapat digunakan untuk mengubah cara file log biner mencari GTIDs sebelumnya selama pemulihan, mempercepat proses ketika sejumlah besar file log biner yang sudah ada.
- Internal, tipe data spasial seperti Geometri direpresentasikan sebagai nilai-nilai BLOB, sehingga ketika dijalankan dengan opsi `--hex-blob`, `mysqldump` sekarang menampilkan nilai-nilai tata ruang di hex.
- InnoDB; *Partitioning*: Sejumlah besar tabel InnoDB dipartisi bisa mengkonsumsi banyak memori ketika digunakan di MySQL 5.6 atau 5.5 dari memori yang digunakan oleh tabel yang sama digunakan dalam rilis sebelumnya dari MySQL Server.
- InnoDB : Sebuah *ALTER TABLE ... ADD* operasi *FOREIGN KEY* bisa menyebabkan kesalahan serius.
- InnoDB : Dalam membangun debug , operasi *INSERT* mempengaruhi tabel dikompresi akan menaikkan pernyataan terkait sync. (sumber: dev.mysql.com)

2.10 Unified Modeling Language (UML)

Unified Modeling Language (UML) adalah representasi grafis parsial (tampilan) dari model sistem yang masih dalam tahap desain, tahap implementasi ataupun sistem yang sudah ada. Diagram UML terhubung dengan garis-garis (jalur atau arus) yang mewakili unsur-unsur dalam model UML dari sistem yang dirancang. (uml-diagram.org, 2016)

Pada UML 2.5 terdiri dari 11 macam diagram yang dikelompokkan dalam 2 kategori. Pembagian kategori tersebut dapat dilihat pada gambar berikut ini (uml-diagram.org, 2016):



Gambar II.7 Klasifikasi Diagram UML

Sumber: uml-diagram.org


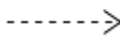



Berikut ini penjelasan singkat dari pembagian kategori tersebut (uml-diagram.org, 2016):

1. *Structure diagrams*, yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan
2. *Behavior diagrams*, yaitu kumpulan diagram yang digunakan untuk menggambarkan proses sistem atau rangkaian perubahan yang terjadi pada sebuah sistem.

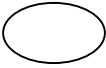
2.10.1 Use Case Diagram

Use case adalah Menggambarkan seperangkat tindakan (kasus penggunaan) bahwa beberapa sistem atau sistem (subjek) harus atau dapat melakukan kerja sama dengan satu atau lebih pengguna eksternal dari sistem (aktor) untuk memberikan beberapa hasil yang dapat diamati dan berharga untuk para aktor atau pemangku kepentingan lainnya dari system (uml-diagram.org, 2016).

Tabel II.2 Simbol-simbol *Use Case Diagram*

| Simbol | Nama | Keterangan |
|---|--------------------|---|
|  | <i>Actor</i> | Seorang aktor yang menentukan peran yang dimainkan oleh entitas eksternal yang berinteraksi dengan subjek (misalnya, dengan bertukar sinyal dan data), pengguna manusia dari sistem yang dirancang, beberapa sistem lain atau perangkat keras menggunakan layanan dari subjek . |
|  | <i>Include</i> | Mencakup hubungan diantara 2 <i>use cases</i> atau lebih yang digunakan untuk menunjukkan bahwa perilaku use case termasuk kedalam perilaku use case dasar. |
|  | <i>Extend</i> | Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan. |
|  | <i>Association</i> | menghubungkan antara objek satu dengan objek lainnya. |
|  | <i>Subject</i> | Subjek (kadang-kadang disebut batas sistem) disajikan oleh persegi panjang dengan nama subjek , terkait kata kunci dan stereotip di sudut kiri atas. Berlaku untuk subjek yang terletak di dalam persegi panjang dan aktor - di luar batas sistem. |

Tabel II.2 Simbol-simbol *Use Case Diagram* (Lanjutan)





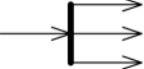

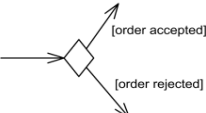
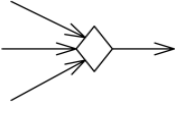
| Simbol | Nama | Keterangan |
|---|-----------------|--|
|  | <i>Use Case</i> | Deskripsi dari urutan aksi-aksi yang ditampilkan sistem. |

Sumber: uml-diagram.org (2016)

2.10.2 Activity Diagram

Activity diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem (Rosa dan Shalahuddin, 2011).

Tabel II.3 Simbol-simbol *Activity Diagram*

| Simbol | Nama | Keterangan |
|---|----------------------------|---|
|  | <i>Activity</i> | Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain |
|  | <i>Action</i> | <i>State</i> dari sistem yang mencerminkan eksekusi dari suatu aksi |
|  | <i>Initial Node</i> | Bagaimana objek dibentuk atau diawali. |
|  | <i>Activity Final Node</i> | Bagaimana objek dibentuk dan dihancurkan |
|  | <i>Fork Node</i> | Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran |
|  | <i>Join Node</i> | Beberapa aliran yang pada tahap tertentu berubah menjadi satu aliran |
|  | <i>Decision Node</i> | <i>node</i> kontrol yang menerima token pada satu atau dua sisi masuk dan memilih salah satu ujung keluar dari satu atau lebih arus keluar. |
|  | <i>Merge Node</i> | Menggabungkan <i>node-node</i> kendali yang menyatukan beberapa alternatif yang masuk mengalir untuk menerima aliran |

Tabel II.3 Simbol-simbol *Activity Diagram* (Lanjutan)

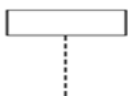


| Simbol | Nama | Keterangan |
|--------|------|----------------|
| | | keluar tunggal |

Sumber: uml-diagram.org (2016)

2.10.3 *Sequence Diagram*

Sequence diagram digunakan untuk menggambarkan perilaku sebuah skenario. Diagram ini menunjukkan sejumlah contoh objek dan pesan yang diletakkan diantara obyek-obyek ini di dalam *use case*. Objek diletakkan di dekat bagian atas diagram dengan urutan dari kiri ke kanan. Setiap *participant* terhubung dengan garis titik-titik yang disebut *lifeline*. Sepanjang *line* ada kotak yang disebut *activation* (Munawar, 2005).

Tabel II.4 Simbol-simbol *Sequence Diagram*

| Simbol | Nama | Keterangan |
|---|-----------------|--|
|  | <i>LifeLine</i> | Objek <i>entity</i> , antarmuka yang saling berinteraksi. |
|  | <i>Message</i> | Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi |
|  | <i>Message</i> | Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi |



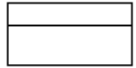


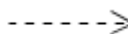
Sumber: Munawar (2005)

2.10.4 *Class Diagram*


Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem (Rosa dan Shalahuddin, 2011). *Class diagram* adalah sebuah class yang menggambarkan struktur dan penjelasan class, paket, dan objek serta hubungan satu sama lain seperti containment, pewarisan, asosiasi, dan lain-lain. Class diagram juga menjelaskan hubungan antar class dalam sebuah sistem yang sedang dibuat dan bagaimana caranya agar mereka saling berkolaborasi untuk mencapai sebuah tujuan.

Class juga memiliki 3 area pokok (utama) yaitu : nama, atribut, dan operasi. Nama berfungsi untuk member identitas pada sebuah kelas, atribut fungsinya adalah untuk member karakteristik pada data yang dimiliki suatu objek di dalam kelas, sedangkan operasi fungsinya adalah memberikan sebuah fungsi ke sebuah objek. Dalam mendefinisikan metode yang ada di dalam kelas harus diperhatikan yang namanya *Cohesion* dan *Coupling*, *Cohesion* adalah ukuran keterkaitan sebuah instruksi di sebuah metode, *Coupling* adalah ukuran keterkaitan antar metode. Di dalam class diagram terdapat hubungan antar kelas secara konseptual, yang disebut Relasi antar *Class*, di UML disediakan macam-macam relasi antar *Class*, diantaranya: Asosiasi (Hubungan statis antar kelas), *Agregasi* (hubungan dari keseluruhan objek), *Generalisasi* (relasi beberapa subkelas ke super kelas), *Dependency* (keterhubungan tiap kelas).

Tabel II.5 Simbol-simbol *Class Diagram*

| Simbol | Nama | Keterangan |
|---|-------------------------|--|
|  | <i>Generalization</i> | Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>). |
|  | <i>Nary Association</i> | Upaya untuk menghindari asosiasi dengan lebih dari 2 objek. |
|  | <i>Class</i> | Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama. |
|  | <i>Collaboration</i> | Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor |
|  | <i>Realization</i> | Operasi yang benar-benar dilakukan oleh suatu objek. |
|  | <i>Dependency</i> | Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri |

Tabel II.5 Simbol-simbol *Class Diagram* (Lanjutan)

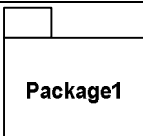

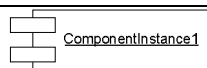


| Simbol | Nama | Keterangan |
|---|--------------------|---|
|  | <i>Association</i> | Apa yang menghubungkan antara objek satu dengan objek lainnya |

Sumber: Munawar (2005)

2.10.5 Component Diagram

Component diagram mengandung *component*, *interface* dan *relationship*. Hal yang penting pada *component* adalah *component* mewakili potongan-potongan yang independen yang bisa dipesan dan diperbaharui sewaktu-waktu. *Component* dihubungkan melalui *interface* yang diimplementasikan (Munawar, 2005).

Tabel II.6 Simbol-simbol *Component Diagram*


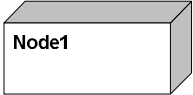
| Simbol | Nama | Keterangan |
|---|-------------------|---|
|  | <i>Package</i> | Merupakan sebuah kumpulan dari satu atau lebih komponen |
|  | <i>Link</i> | Relasi antar objek |
|  | <i>Component</i> | Komponen system |
|  | <i>Dependency</i> | Hubungan suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>). |
|  | <i>Interface</i> | Sebagai antarmuka komponen |

Sumber: Rosa dan Shalahuddin (2011)

2.10.6 Deployment Diagram

Deployment Diagram menyediakan gambaran bagaimana sistem secara fisik akan terlihat. Sistem terdiri dari *node-node* di mana setiap *node* diwakili oleh sebuah kubus dan garis yang menghubungkan antara kubus tersebut menunjukkan hubungan antara kedua *node* tersebut (Munawar, 2005).

Tabel II.7 Simbol-simbol *Deployment Diagram*

| Simbol | Nama | Keterangan |
|--|-------------------|--|
|  | <i>Package</i> | Merupakan sebuah kumpulan dari satu atau lebih komponen |
|  | <i>Link</i> | Relasi antar objek |
|  | <i>Dependency</i> | Hubungan pada suatu elemen mandiri (independent) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (independent). |
|  | <i>Node</i> | Perangkat keras dan perangkat lunak |

Sumber: Rosa dan Shalahuddin (2011)







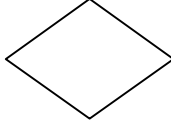
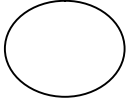
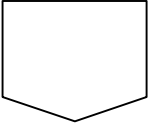
2.11 Bagan Alir (*Flowchart*)

Bagan alir adalah bagan yang menunjukkan alir didalam program atau prosedur sistem secara logika. Menurut Jogiyanto (2005), terdapat 5 macam bagan alir, yaitu:

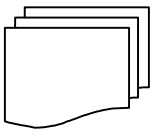
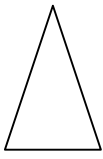
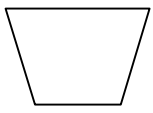
1. Bagan alir sistem (*systems flow chart*)
2. Bagan alir dokumen (*document flow chart*)
3. Bagan alir skematik (*schematic flow chart*)
4. Bagan alir program (*program flow chart*)
5. Bagan alir proses (*process flow chart*)

Berikut ini adalah simbol yang biasa digunakan dalam *flow chart*:

Tabel II.8 Simbol-Simbol *Flow Chart*

| Simbol | Nama | Fungsi |
|---|---|--|
|  | Terminator | Awal dan akhir dari suatu proses. |
|  | Garis Akhir (<i>Front Line</i>) | Arus dari suatu proses |
|  | <i>Preparation</i> | Proses inisialisasi awal |
|  | Proses | Proses pengolahan data |
|  | Input/Output Data | Mewakili data masukan atau keluaran. |
|  | <i>Predefined Process</i> (Sub Proses) | Permulaan sub proses |
|  | <i>Decision</i> | Perbandingan pernyataan, penyeleksian data yang memberikan pilihan untuk langkah selanjutnya |
|  | <i>On Page Connector</i> | Penghubung bagian-bagian <i>flowchart</i> yang berada pada satu halaman |
|  | <i>Off Page Connector</i> | Penghubung bagian-bagian <i>flowchart</i> yang berada pada halaman berbeda |

Tabel II.8 Simbol-Simbol *Flow Chart* (Lanjutan)

| Simbol | Nama | Fungsi |
|---|-----------------|---|
|  | Dokumen Rangkap | Menggambarkan dokumen asli dan tembusannya |
|  | Arsip Permanen | Menunjukkan tempat penyimpanan dokumen secara permanen yang tidak akan diproses lagi |
|  | Proses Manual | Untuk menggambarkan kegiatan manual seperti : menerima <i>order</i> , mengisi formulir, membandingkan, dll. |

Sumber: Jogiyanto (2005)

BAB III

METODOLOGI PENELITIAN

3.1 Metodologi Penelitian

Metodologi penelitian adalah tata cara bagaimana suatu penelitian dilaksanakan atau cara yang ditempuh sehubungan dengan penelitian yang dilakukan, yang memiliki langkah-langkah sistematis Hasan (2002). Metodologi penelitian juga dikenal sebagai metode ilmiah dalam mencari, mengembangkan dan menguji suatu kebenaran pengetahuan.

Model *prototype evolutioner* adalah metode pengembangan sistem yang digunakan penulis dalam penelitian ini, karena penulis memulai penelitian ini dengan mengumpulkan kebutuhan yang diperlukan sistem atau perangkat lunak yang akan dibuat. Pada penelitian ini untuk mengatasi masalah yang ada pada sistem, diputuskan untuk membuat sistem informasi *project tracking*. Sistem informasi *project tracking* akan dibuat menjadi lebih terkomputerisasi dan berbasis web menggunakan bahasa pemrograman PHP. Pengembangan sistem dilakukan dengan menggunakan metode *prototype evolutioner*. Metode *prototype* ini terdiri dari tahap identifikasi kebutuhan pengguna, pembuatan *prototype*, evaluasi *prototype* sistem baru hingga didapatkan sistem yang memang dikehendaki pengguna, dan penggunaan *prototype*.

3.2 Jenis dan Sumber Data

Data yang diperlukan dalam penelitian ini ada 2 jenis, yaitu data primer dan data sekunder. Sumber dari data-data ini berasal dari tempat yang diamati oleh penulis di divisi teknik sipil khususnya penilikan pada PT Pelabuhan Tanjung Priok.

1. Data Primer

Data primer adalah data yang diperoleh secara langsung dari perusahaan dimana pengumpulan data atau informasi dilakukan langsung dari objek yang diteliti. Data-data tersebut adalah laporan-laporan mingguan pekerjaan, rekap

laporan pekerjaan, berita acara pekerjaan, dan informasi yang ada didalam buku perjanjian pekerjaan atau buku kontrak pekerjaan.

2. Data Sekunder

Data sekunder adalah data yang diperoleh dari data yang tersedia dan telah terlebih dahulu dikumpulkan dan dilaporkan oleh orang lain, buku-buku dan kajian ilmiah dari berbagai sumber yang berkaitan dengan penelitian. Data tersebut adalah informasi umum perusahaan, profil perusahaan dan struktur organisasi perusahaan.

3.3 Metode Pengumpulan Data

Metode pengumpulan data adalah cara yang dapat digunakan oleh peneliti untuk mengumpulkan data (Arikunto, 2007). Penelitian ini menggunakan beberapa teknik pengumpulan data, yaitu:

1. Studi lapangan

Studi lapangan adalah usaha melakukan pengumpulan data secara langsung pada objek yang dilakukan dengan cara sebagai berikut:

- a. Observasi, tahap ini dilakukan secara langsung di divisi teknik sipil PT Pelabuhan Tanjung Priok dengan mengamati proses pengawasan *progress* pekerjaan.
- b. Wawancara, yaitu mencari data yang dibutuhkan secara langsung melalui memberikan pertanyaan spontan terhadap segala hal yang diperlukan pada penyusunan tugas akhir ini. Wawancara ini dilakukan kepada pengawas dan pegawai di sub divisi penilikan.

2. Studi kepustakaan

Studi kepustakaan dilakukan dengan membaca buku dan literatur dalam lingkup perkuliahan maupun di luar lingkup perkuliahan yang berhubungan dengan judul dan permasalahan sehingga dapat menunjang dalam penulisan tugas akhir ini. Studi kepustakaan yang dilakukan adalah dengan menggunakan buku yang dimiliki, buku yang dipinjam dari perpustakaan dan mencari data yang diperlukan melalui internet.

3.4 Metode yang Dilakukan

Tahapan-tahapan dari metode yang dilakukan dalam pengembangan sistem *prototype evolutioner* adalah sebagai berikut:

1. Pengumpulan kebutuhan

User dan pengembang bersama-sama mendefinisikan format seluruh perangkat lunak, mengidentifikasi semua kebutuhan, dan garis besar sistem yang akan dibuat.

2. Membangun prototype

Membangun prototyping dengan membuat perancangan sementara yang berfokus pada penyajian kepada *user*, misalnya dengan membuat input dan format output.

3. Evaluasi protoptype

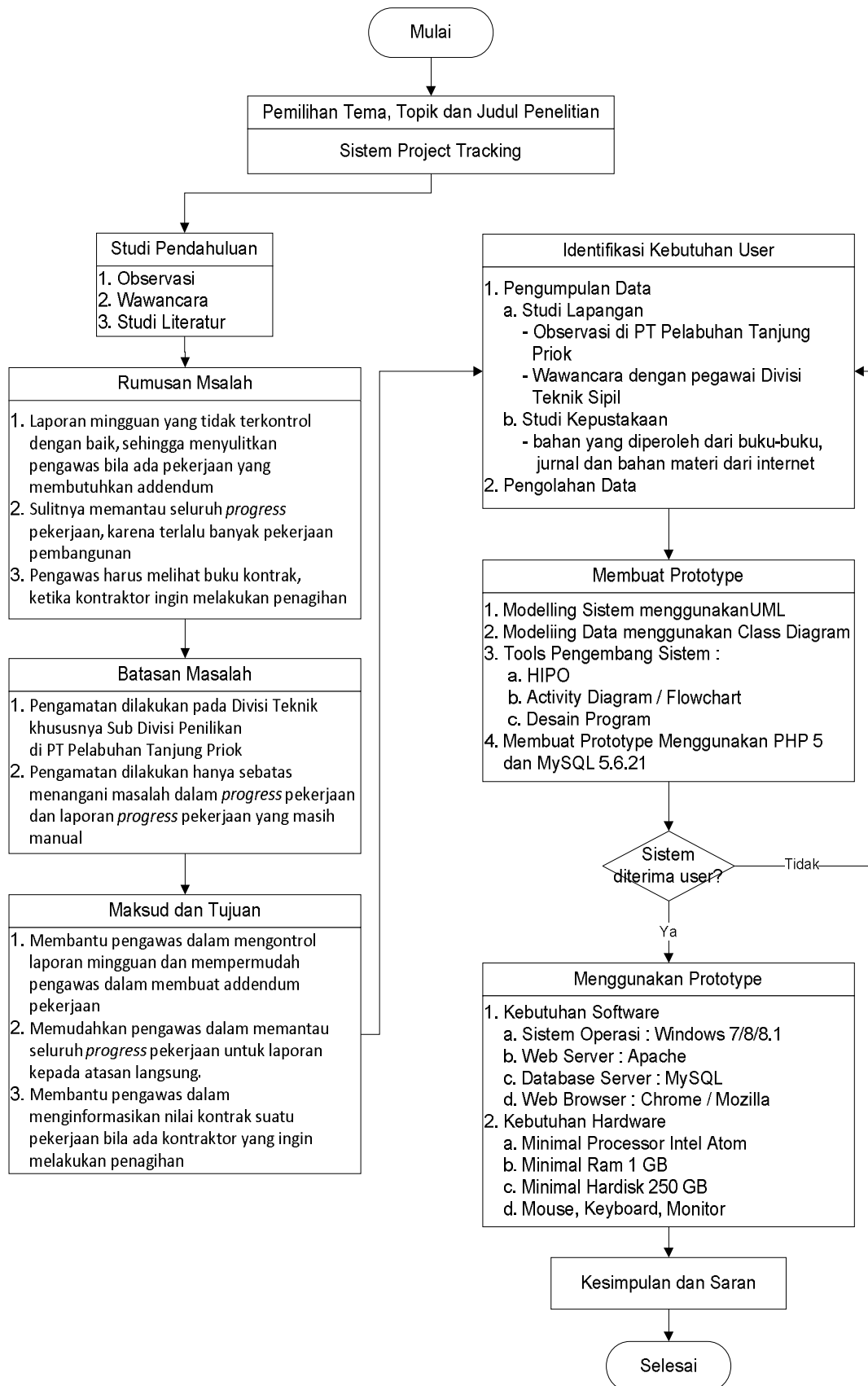
Evaluasi ini dilakukan oleh *user* apakah prototype yang sudah dibangun sesuai dengan keinginann *user*. Jika sudah sesuai maka langkah mengkodekan sistem akan diambil, jika tidak *prototype* direvisi dengan mengulangi langkah pengumpulan kebutuhan, membangun *prototype* dan evaluasi *prototype*.

4. Menguji sistem

Setelah sistem sudah menjadi suatu perangkat lunak yang siap pakai, harus diuji dahulu sebelum digunakan.

3.5 Kerangka Penelitian

Gambar III.1. berikut adalah kerangka penelitian yang menggambarkan bagaimana pemecahan masalah dari awal sampai akhir penelitian berdasarkan hasil keseluruhan penelitian di PT Pelabuhan Tanjung Priok. Berikut ini adalah *flowchart* kerangka pemecahan masalah:



Gambar III.1. Kerangka Penelitian.

Dalam pemecahan masalah yang dibahas dalam tugas akhir ini, penulis melakukan langkah-langkah atau tahapan dalam pemecahan masalah yang ada pada gambar III.1. sesuai dengan metodologi penelitian yang penulis gunakan dalam pemecahan masalah tersebut. Penjelasan langkah-langkah atau tahapan dalam pemecahan masalah tersebut adalah sebagai berikut:

1. **Pemilihan Tema, Topik dan Judul Penelitian**
Merupakan langkah awal penelitian, yaitu dengan menentukan tema dan topik sistem yang akan dirancang atau dikembangkan, kemudian membuat judul dari topik tersebut.
2. **Studi Pendahuluan**
Studi pendahuluan yang penulis lakukan yaitu dengan membaca buku literatur, *browsing* internet serta sumber-sumber lain dalam lingkup perkuliahan maupun di luar lingkup perkuliahan yang berhubungan dengan judul dan permasalahan tugas akhir. Studi pendahuluan ini dimaksudkan untuk dapat mengetahui suatu gambaran yang jelas mengenai kondisi dan situasi PT Pelabuhan Tanjung Priok pada saat ini, serta untuk mengetahui masalah yang sedang dihadapi. Langkah-langkah yang dilakukan dalam studi pendahuluan yang bertujuan untuk mendapatkan informasi adalah dengan cara observasi dan melakukan wawancara terhadap pihak-pihak terkait yang berada di PT Pelabuhan Tanjung Priok.
3. **Rumusan Masalah**
Dilakukan identifikasi masalah yang sering terjadi pada sistem yang berjalan, dalam penelitian ini dilakukan melalui studi pustaka dan observasi secara langsung pada sistem yang berjalan.
4. **Batasan Masalah**
Batasan masalah yaitu dilakukan agar penelitian yang dilakukan lebih terfokus dan tidak melenceng kemana-mana.
5. **Maksud dan Tujuan**
Tujuan dari penelitian ini yaitu membuat aplikasi *Project Tracking* berbasis *web* yang diharapkan dapat memberikan masukan dan memberikan

pemecahan masalah kepada PT Pelabuhan Tanjung Priok, serta sebagai referensi bagi peneliti lain yang melakukan penelitian serupa.

6. Identifikasi Kebutuhan *User*

Identifikasi kebutuhan *user* merupakan langkah lanjutan dari pengolahan data. Dengan monitoring progress pekerjaan yang berjalan saat ini dan menggunakan dokumen – dokumen yang ada seperti berita acara ataupun laporan mingguan dapat teridentifikasi apa yang sebenarnya *user* inginkan dari pembuatan aplikasi ini.

7. Membuat *Prototype*

Melakukan perancangan *prototype*, *modelling* sistem menggunakan UML, *modelling* data menggunakan *class diagram* dan *tools* pengembang sistem menggunakan *HIPO diagram*, *activity diagram* / *flowchart*, mendesain program dan membuat *prototype* menggunakan PHP 5 dan MySQL 5.6.21.

8. Menggunakan *Prototype*

Memutuskan untuk menggunakan sistem operasional yang telah divalidasi menggunakan metode *prototype*.

9. Kesimpulan dan Saran

Mengambil kesimpulan dari hasil analisis sistem berjalan, sistem yang diusulkan dan aplikasi yang dirancang serta memberikan saran kepada PT Pelabuhan Tanjung Priok dan kepada pengembang atau peneliti selanjutnya.

BAB IV

PENGUMPULAN DAN PENGOLAHAN DATA

4.1 Informasi Umum Perusahaan

PT. Pelabuhan Tanjung Priok beroperasi penuh menjadi anak perusahaan ke 14 dari IPC PT Pelabuhan Indonesia II (Pesero), sesuai surat Keputusan Bersama Direksi Pelabuhan Indonesia II (persero) dan Direksi Multi Terminal Indonesia Nomor HK.56/28/5/4/PI.II-14 dan Nomor HK.476/1/18/MTI-2014 tentang Organisasi PT Pelabuhan Tanjung Priok tanggal 28 Mei 2014 dan Surat Keputusan Menteri Perhubungan RI No KP 818 TAHUN 2014 Tanggal 29/09/14 tentang pembentukan Badan Usaha Pelabuhan PT Pelabuhan Tanjung Priok.

PT Pelabuhan Tanjung Priok lahir dengan sebuah keyakinan besar bahwa PT Pelabuhan Tanjung Priok akan mampu menciptakan pola kerja serta kualitas pelayanan kepelabuhanan secara fleksibel, cepat dan berfokus pada penekanan biaya operasi. PT Pelabuhan Tanjung Priok akan bergerak lebih cepat, efektif dan efisien dalam melayani kebutuhan para pelanggan atau pengguna jasa kepelabuhanan, semakin singkat waktu yang diperlukan dalam memproses layanan kepelabuhanan, semakin rendah pula biaya yang harus dikeluarkan. PT Pelabuhan Tanjung Priok mengintensifkan komunikasi dengan pelaku usaha dan asosiasi pengguna jasa pelabuhan untuk senantiasa menjalankan program modernisasi dan efisiensi layanan jasa kepelabuhanan.

4.1.1 Profil Perusahaan

| | |
|----------------|--|
| Nama | : PT Pelabuhan Tanjung Priok |
| Kantor Direksi | : Jalan Raya Pelabuhan No.9 Tanjung Priok, Jakarta |
| Telepon | : 021 - 4301080 |
| Faks | : 021 – 4372947 |
| Email | : priok@indonesiaport.co.id |
| Website | : www.priokport.co.id |

4.1.2 Visi dan Misi Perusahaan

1. Visi

Menjadi hub nasional terdepan yang melayani pertumbuhan ekonomi Indonesia.

2. Misi

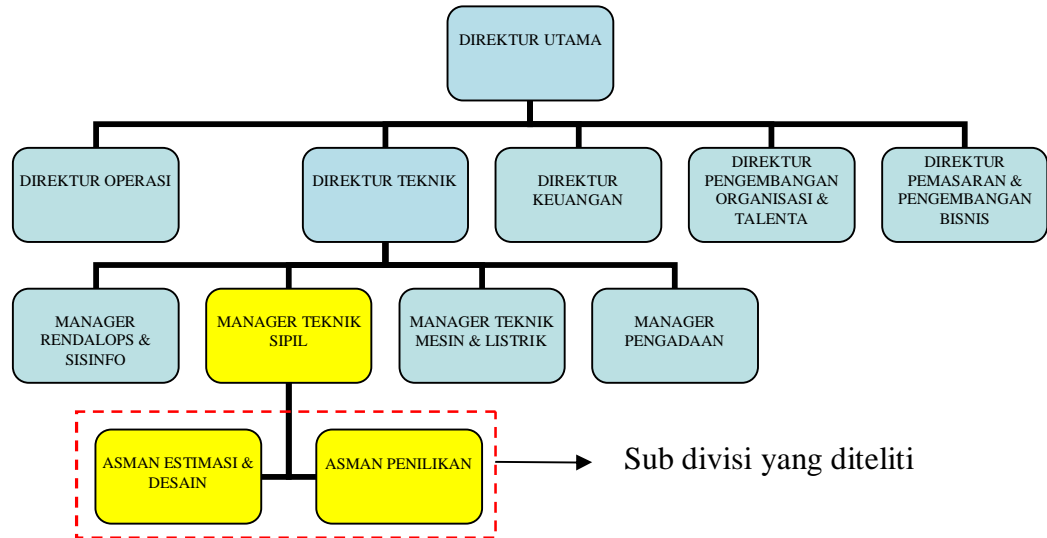
- a. Menjamin terlaksananya pelayanan yang produktif aman, mengedepankan keselamatan, dan ramah lingkungan.
- b. Menciptakan nilai tambah bagi pelanggan dan mitra bisnis.
- c. Menyelenggarakan budaya perusahaan yang bersih, menyenangkan untuk bekerja dan memberikan kesempatan yang sama untuk berkembang.
- d. Menyelenggarakan tata kelola perusahaan yang transparan, akuntabel, bertanggung jawab, berintegritas, dan adil.

4.2 Struktur Organisasi Perusahaan

Struktur organisasi adalah suatu kerangka atau susunan dasar tata laksana suatu badan organisasi dari suatu perusahaan. Struktur organisasi yang baik harus bersifat fleksibel sehingga memungkinkan untuk diadakan perubahan atau perbaikan untuk menunjang keberhasilan perusahaan.

Selain bersifat fleksibel, struktur organisasi juga harus dapat menggambarkan dengan jelas wewenang dan tanggung jawab yang ada dalam perusahaan, agar tidak terjadi kerancuan di antara fungsi-fungsi dari tiap bagian. Sehingga tiap-tiap bagian tersebut dapat mengoptimalkan kemampuan masing-masing demi tercapainya kemajuan perusahaan. Adapun struktur organisasi dari PT Pelabuhan Tanjung Priok adalah sebagai berikut:

Pada gambar IV.1 merupakan struktur organisasi keseluruhan dari PT Pelabuhan Tanjung Priok, berikut struktur organisasi dari Divisi Teknik Sipil



Gambar IV.2 Struktur Organisasi Divisi Teknik Sipil

Sumber : *Corporate Secretary & Hubungan Eksternal PT. PTP 2016*

A. Manager Teknik Sipil

Bertanggung jawab dalam pembangunan dan pemeliharaan fasilitas – fasilitas yang berada di Pelabuhan Tanjung Priok. Manager Teknik Sipil membawahi Asisten Manager Estimasi & Desain dan Asisten Manager Penilikan.

B. Asisten Manager Estimasi & Desain

Melakukan perencanaan pembangunan, baik itu pembangunan gedung, dermaga, lapangan penumpukan maupun fasilitas – fasilitas yang terkait dengan kepelabuhanan. Perencanaannya meliputi gambar rencana pembangunan dan Rencana Anggaran Biaya (RAB).

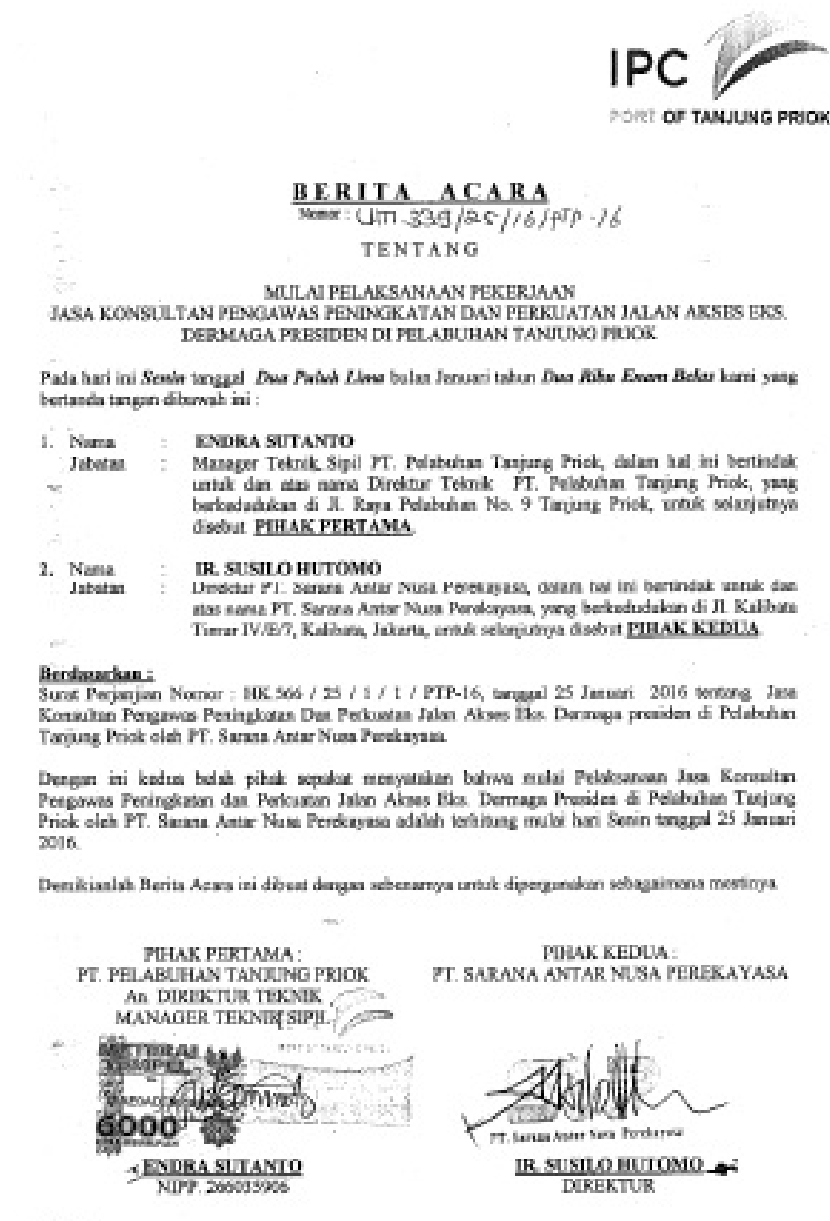
C. Asisten Manager Penilikan

Mengawasi jalannya proyek pembangunan yang dilakukan kontraktor, pekerjaan mengacu kepada perencanaan yang sudah direncanakan di bagian estimasi & desain.

4.3 Progress Pekerjaan

4.3.1 Berita Acara (BA) Mulai Pekerjaan

Berita acara mulai pekerjaan diterbitkan ketika pihak pertama dan pihak kedua telah setuju untuk waktu dimulainya pekerjaan. Berita acara ini menjadi dasar kontraktor untuk memulai pekerjaannya.



Gambar IV.3 Berita Acara Mulai Pekerjaan
Sumber : Sub Divisi Penilikan 2016

4.3.2 Laporan mingguan

Pekerjaan pengawasan dilakukan ketika kontraktor sudah memulai pekerjaannya yang didasari oleh surat berita acara mulai pekerjaan. Pengawasan dilakukan dengan maksud pekerjaan yang dilakukan sesuai dengan perencanaan yang sudah dilakukan baik itu dari volume pekerjaan, biaya ataupun waktu pekerjaan. Kontraktor melaporkan pekerjaannya melalui laporan mingguan yang diberikan setiap minggunya, laporan tersebut berisi kegiatan mereka selama satu minggu penuh dan progress fisik pekerjaan tersebut. Berikut contoh *progress* pekerjaan laporan mingguan dan monitoring laporan mingguan.

| NO | URAIAN PEKERJAAN | KONTRAK | | | KEMAJUAN PEKERJAAN | | | | | | | | | | | | |
|----|---|---------|----------|--------|----------------------|---------------|--------|---------|---------------|--------|-------------------------------|---------------|---------|--------|---|---|--------|
| | | SAT | VOL | BOBOT | PRESTASI MINGGU LALU | | | | | | PRESTASI SID AKHIR MINGGU INI | | | | | | |
| | | | | | VOLUME | PERCENT FISIK | BOBOT | VOLUME | PERCENT FISIK | BOBOT | VOLUME | PERCENT FISIK | BOBOT | | | | |
| | - Tee reduser 6" x 2 1/2" HDPE | unit | 7.0000 | 0.0152 | - | - | - | - | - | - | - | - | - | - | - | - | - |
| | - Beud M 16 x 50mm HDG | unit | 56.0000 | 0.0025 | 39.2000 | 70.0000 | 0.0018 | - | - | - | - | 39.2000 | 70.0000 | 0.0018 | - | - | - |
| | - Coupling House Machine dia 2 1/2" | unit | 7.0000 | 0.0033 | 4.9000 | 70.0000 | 0.0023 | - | - | - | - | 4.9000 | 70.0000 | 0.0023 | - | - | - |
| 4 | Pengadaan dan Pemasangan Outback Pipe HDPE 18" dia. 200 x 400 x 100 berulir clamp pipa T= 3 mm dia. 200 mm termasuk dyabolt + bandulan karet + angkur + | buah | 101.8000 | 0.0774 | 112.7000 | 70.0000 | 0.0542 | - | - | - | - | 112.7000 | 70.0000 | 0.0542 | - | - | - |
| 5 | Pengadaan & Pemasangan Gate Valve F100 KITZ 18" x SMO dia. 18" | unit | 3.0000 | 0.1400 | - | - | - | 2.1000 | 70.0000 | 0.0980 | - | 2.1000 | 70.0000 | 0.0980 | - | - | - |
| 6 | Pengadaan & Pemasangan Siku Flange dia 18" | unit | 6.0000 | 0.0121 | - | - | - | 4.2000 | 70.0000 | 0.0084 | - | 4.2000 | 70.0000 | 0.0084 | - | - | - |
| 7 | Pengadaan dan Pemasangan Rubber Gasket T= 3 mm dia. 200 mm | buah | 3.0000 | 0.0022 | - | - | - | 2.1000 | 70.0000 | 0.0015 | - | 2.1000 | 70.0000 | 0.0015 | - | - | - |
| 8 | Pengadaan dan Pemasangan Bolt & Nuts + Ring 3/4 HDG | buah | 48.0000 | 0.0011 | - | - | - | 33.6000 | 70.0000 | 0.0008 | - | 33.6000 | 70.0000 | 0.0008 | - | - | - |
| 9 | Pengadaan dan Pemasangan Grounding Rod | ls | 1.0000 | 0.0255 | - | - | - | 0.7000 | 70.0000 | 0.0175 | - | 0.7000 | 70.0000 | 0.0175 | - | - | - |
| 10 | Pemasangan Bending Plate di pipi cable drum (Stainless steel, N-6%) | buah | 3.0000 | 0.0077 | - | - | - | - | - | - | - | - | - | - | - | - | - |
| 11 | Test Commissioning dan Apat banti | ls | 1.8000 | 0.0077 | - | - | - | - | - | - | - | - | - | - | - | - | - |
| | Sub Total I.C. | | | 0.7124 | | | 0.0176 | | | 0.1206 | | | | 0.2982 | | | 0.2982 |
| | GRAND TOTAL | | 100.0000 | | | | 0.2733 | | | 0.2982 | | | | 0.2982 | | | 0.2982 |

Koordinator Pengawasan PT. Pelabuhan Tanjung Priuk
I. M. Yudhan Santoso

Pengawas/ Inspector PT LAPRITS
1 Fakhriah

Mengetahui,
Staff Senior Penilikan
PT. PELABUHAN TANJUNGG PIRUK

Archandri Rahmawati
NPP. 28767688

Mengetahui,
Manajer Teknik Sipil
PT. PELABUHAN TANJUNGG PIRUK

Endra Subianto
NPP. 28655905

Dipaparkan,
Konsultan Pengawasan
PT. LAPRITS

Yuli Setiati Kurniawan
Site Engineer

Dipaparkan,
Kontrolor Pelaksanaan
PT. NINDIA WARYA (Pronosa)

Eko Subandono
Project Manager

Mengetahui,
Asisten Manajer Penilikan
PT. PELABUHAN TANJUNGG PIRUK

Eko Sugilo
NPP. 28410758

| ITEM PEKERJAAN | RENCANA SID MINGGU INI | REALISASI SID MINGGU INI | SELISIH / DEVIASI |
|----------------------------------|------------------------------|--------------------------------|-------------------------|
| PEKERJAAN PERSIAPAN | 1.1995 | 1.1449 | 0.0054 |
| PEKERJAAN DERAGAGA 212 | 50.0827 | 49.0861 | (0.3566) |
| PEKERJAAN DERAGAGA 213 | 44.1597 | 45.5243 | 1.3646 |
| PEKERJAAN MEKANIKAL & ELEKTRIKAL | 0.7134 | 0.2042 | (0.5092) |
| TOTAL | 96.0952 | 96.5595 | 0.4643 |

Gambar IV.4 *Progress* Pekerjaan Laporan mingguan
Sumber : Sub Divisi Penilikan 2016

4.3.3 Monitoring Laporan Mingguan

Monitoring laporan mingguan dibuat oleh para pengawas yang bersisi rencana progress pekerjaan yang kemudian dibandingkan dengan progress realisasi dan dari perbandingan tersebut menghasilkan deviasi progress. Dari deviasi progress tersebut kemudian dievaluasi lalu diambil keputusan selanjutnya.

The screenshot shows an Excel spreadsheet titled "Monitoring Laporan Mingguan 212-213 (Autosaved).xlsx". The spreadsheet contains the following data:

| No | Minggu ke - | Periode | Rencana Progress Kumulatif Awal | Rencana Progress Kumulatif Revisi | Progress Minggu Ini | Progress Kumulatif | Deviasi | Keterangan | Status |
|---|-------------|-----------------------------|---------------------------------|-----------------------------------|---------------------|--------------------|---------|--|----------------------|
| MONITORING LAPORAN MINGGUAN Pekerjaan : Perkuatan Dermaga 212 - 213 Pelabuhan Tanjung Priuk No Kontrak : HK. 566 / 10 / 8 / 1 / PTP - 15 Nilai Kontrak : 100.621.000.000,00 BA mulai kontraktor : UM. 339 / 18 / 8 / 19 / PTP - 15 | | | | | | | | | |
| 1 | 1 | 18 Aug 2015 - 23 Aug 2015 | 0,1103 | | 0,0668 | 0,0668 | -0,0435 | Notulen rapat ok | ok untuk diputar ttd |
| 2 | 2 | 24 Aug 2015-30 Aug 2015 | 0,2205 | | 0,2112 | 0,278 | 0,0575 | Notulen rapat ok | ok untuk diputar ttd |
| 3 | 3 | 31 Aug 2015 - 6 Sept 2015 | 0,3308 | | 0,4424 | 0,7204 | 0,3896 | Notulen rapat ok | ok untuk diputar ttd |
| 4 | 4 | 7 Sept 2015 - 13 Sept 2015 | 0,3815 | | 1,8588 | 2,5792 | 2,1977 | Notulen rapat ok | ok untuk diputar ttd |
| 5 | 5 | 14 Sept 2015 - 20 Sept 2015 | 2,0348 | | 3,7352 | 6,3144 | 4,2796 | Notulen rapat ok Koreksi hitungan glass flake di LK ok | ok untuk diputar ttd |
| 6 | 6 | 21 Sept 2015 - 27 Sept 2015 | 3,9395 | | 1,456 | 7,7704 | 3,8309 | Notulen rapat ok Koreksi hitungan glass flake di LK | ok untuk diputar ttd |
| 7 | 7 | 28 sept 2015 - 4 okt 2015 | 6,5153 | | 2,6312 | 10,4016 | 3,8863 | notulen rapat ok | ok untuk diputar ttd |
| 8 | 8 | 5 okt 2015 -11 okt 2015 | 9,1637 | 8,1957 | 4,2303 | 14,6319 | 6,4362 | beri warna spun pile no 66 ok Notulen rapat belum laporan K3 nama jabatan dan perusahaan di kurva S | putar ttd |

Gambar IV.5 Monitoring Laporan Mingguan
Sumber : Sub Divisi Penilikan 2016

4.3.4 Form Progress Pekerjaan

Form progress dicetak oleh bagian administrasi yang kemudian diberikan ke para pengawas, form tersebut berisi progress pekerjaan dalam 1 bulan. Form yang sudah diisi dikembalikan kepada bagian administrasi untuk pembuatan rekap laporan.



PROGRESS PEKERJAAN

| NO | NAMA PEKERJAAN | PROGRESS BULAN INI | PROGRESS KOMULATIF |
|----|---|--------------------|--------------------|
| 1 | Pembangunan jalan tambak dari jalan padaso kejalan pulau papung SID dan supervisi | 27,89 | 57,49 |
| 2 | Peningkatan dan pemeliharaan Dermaga 212 dan 213 termasuk Supervisi | 27,60 | 79,08 |
| 3 | Peningkatan dan pemeliharaan Jalan akses Ets. Dermaga Frejalen termasuk SID dan supervisi | 82,82 | 37,10 |
| 4 | Pelengkapan pelebaran dan Peningkatan jalan pulau papung | 83,77 | 53,27 |

Gambar IV.6 Form Progress Pekerjaan
Sumber : Sub Divisi Estimasi & Desain 2016

4.3.5 Rekap Laporan

Seluruh laporan mingguan pekerjaan yang sudah dikumpulkan dalam waktu 1 (satu) bulan kemudian direkap oleh bagian administrasi yang datanya diinformasikan oleh pengawas pekerjaan. Rekap laporan tersebut kemudian diberikan kepada atasan langsung. Berikut contoh rekap laporan pekerjaan:

Laporan Realisasi Pekerjaan Divisi Teknik Sipil 2016

| NO | URAIAN PEKERJAAN | NOMOR KONTRAK | KONTRAKTOR | NILAI KONTRAK | REALISASI | | KET. POSISI SAAT INI |
|----|---|--|---------------------|---------------|-----------------|----------------|---|
| | | | | | S.d Minggu lalu | S.d Minggu ini | |
| | | | | | PROGRESS | | |
| | | | | % | % | | |
| 1 | Pekerjaan Perbaikan Gudang Pombo termasuk SID, DED, dan Supervisi | | | | | | Desain selesai tetapi belum dapat berjalan menunggu kebijakan Kantor Pusat terhadap investasi Cabang Tanjung Priok. Selain itu buku RKAP Cabang belum diterima. |
| 2 | Pembangunan Lapangan Penumpukan ex. Kantor Adhiguna termasuk SID, DED, dan Supervisi | | | | | | Desain selesai tetapi belum dapat berjalan menunggu kebijakan Kantor Pusat terhadap investasi Cabang Tanjung Priok. Selain itu buku RKAP Cabang belum diterima. |
| 3 | Peningkatan lapangan penumpukan ex. Gudang 002 dan 004 termasuk SID dan supervisi | | | | | | Desain selesai tetapi belum dapat berjalan menunggu kebijakan Kantor Pusat terhadap investasi Cabang Tanjung Priok. Selain itu buku RKAP Cabang belum diterima. |
| 4 | Pembangunan Lapangan sisi Barat Pos I termasuk SID dan Supervisi, dibutuhkan dana Rp. 8,85 miliar, tahun 2011 dianggarkan Rp. 500 juta tidak terserap, tahun 2012 dianggarkan Rp. 7,86 miliar terserap Rp. 2,3 miliar, tahun 2013 dianggarkan Rp. 8,17 miliar tidak terserap, tahun 2014 dianggarkan Rp. 6,39 miliar tidak terserap, tahun 2015 dianggarkan Rp. 500 juta diperkirakan terserap Rp. 240,8 juta, tahun 2016 dianggarkan Rp. 6,24 Miliar | HK.566/118/C.Tpk-12, Tanggal 19 Januari 2012 | PT. Telemega Sebia | Kontrak | 6.620.000,00 | | Selesai |
| | | HK.566/318/C.Tpk-12, Tanggal 20 September 2012 | | Addendum I | 2.371.000,00 | | Selesai |
| | | HK.566/18/1/1/PTP-16 Tanggal 18 Januari 2016 | | Addendum | 5.328.089 | 80.44 | 81.92 |
| | | HK.566/22/1/1/PTP-16, Tanggal 22 Januari 2016 | | Addendum | 170.560 | 83.61 | 88.42 |
| | | | | | | 91.65 | 94.35 |
| 5 | Pembangunan lapangan penumpukan ex.Gedung TKBM disesuaikan menjadi Pembangunan lapangan penumpukan ex.Gedung TKBM termasuk relokasi Gerdu ST 8, dibutuhkan dana Rp. 1,29 miliar, tahun 2014 disesuaikan kebutuhan dana menjadi Rp. 5,8 miliar, tahun 2011 dianggarkan Rp 500 juta tidak terserap, tahun 2012 dianggarkan Rp. 1,25 miliar disesuaikan menjadi Rp. 50 juta tidak terserap, tahun 2013 dianggarkan Rp. 1,25 miliar tidak terserap, tahun 2014 dianggarkan Rp. 1,296 miliar terserap Rp. 94,8 juta, tahun 2015 dianggarkan Rp.5,67 miliar diperkirakan terserap Rp. 829 juta. Tahun 2016 dianggarkan Rp. 4,9 miliar | HK.566/17/2/1/PTP-15, tanggal 17 Februari 2015 | PT. Aye Saudan Jaya | | 780.064 | | Selesai (Tahap I) |
| | | HK.566/7/9/1/PTP_15 Tanggal 7 September 2015 | | Addendum I | 958.117 | | Addendum tambah Rp. 178.063 |
| | | | | | | 33.25 | 37.57 |

| NO | URAIAN PEKERJAAN | NOMOR KONTRAK | KONTRAKTOR | NILAI KONTRAK | REALISASI | | KET. POSISI SAAT INI |
|----|--|---|-----------------------------|---------------|-----------------|----------------|--|
| | | | | | S.d Minggu lalu | S.d Minggu ini | |
| | | | | | PROGRESS | | |
| | | | | % | % | | |
| | | HK.566/28/8/2/PTP-15 Tanggal 28 Agustus 2015 | | Addendum I | 149.413 | | Addendum tambah Rp. 7.528 |
| 6 | Pengadaan Kapal Tunda Kap. 2 x 1200 HP Tipe SRR dibutuhkan dana Rp. 45 Miliar, tahun 2012 dianggarkan Rp. 9 Miliar terserap Rp. 15,4 Miliar, tahun 2013 dianggarkan Rp. 36 Miliar terserap Rp. 20,21 Miliar, tahun 2014 dianggarkan Rp. 9,59 Miliar, terserap Rp. 606,5 Juta, tahun 2015 dianggarkan Rp. 4,46 miliar diperkirakan terserap Rp. 3,9 miliar. Tahun 2016 dianggarkan Rp. 544 juta | | | | 45.000.000 | | Pelaksanaan |
| 12 | Pengadaan Kapal Tunda Kap. 2 x 800 Tipe PFP dibutuhkan dana Rp. 15 Miliar, tahun 2012 dianggarkan Rp. 2 Miliar terserap Rp. 5,1 Miliar, tahun 2013 dianggarkan Rp. 11,5 Miliar terserap Rp. 6,97 Miliar tahun 2014 dianggarkan Rp. 95,26 juta terserap Rp.221 juta, tahun 2015 dianggarkan Rp. 1,49 miliar diperkirakan terserap Rp. 1,01 Miliar. Tahun 2016 dianggarkan Rp. 478 juta | | | | 15.000.000 | | Pelaksanaan |
| 7 | Pekerjaan Supply daya listrik termasuk tambahan daya dan jangkan ke kapal pelanggan di dermaga Pelabuhan Tanjung Priok disesuaikan menjadi Pekerjaan Supply daya listrik termasuk tambahan daya bentuk SID dan Supervisi dibutuhkan dana Rp. 145 Miliar disesuaikan menjadi Rp.185 miliar, tahun 2012 dianggarkan Rp. 1 Miliar tidak terserap, tahun 2013 dianggarkan Rp. 70 Miliar terserap Rp. 33,96 Miliar, tahun 2014 dianggarkan Rp.155 Miliar terserap Rp. 102 Miliar, tahun 2015 dianggarkan Rp. 35 miliar diperkirakan terserap Rp. 20 miliar. Tahun 2016 dianggarkan Rp. 19 Miliar. | HK.566/8/14/C.Tpk-13, Tanggal 16 Oktober 2013 | PT. Waskita Karya (Persero) | | 127.970.000 | - | 0 selesai |
| | | HK.566/1/14/C.Tpk-14, Tanggal 1 Oktober 2014 | | Addendum I | 136.044.910 | 24.50 | 27.32 |
| | | HK.566/24/7/3/C.Tpk-14, Tanggal 24 Juli 2014 | | Addendum II | | 56.13 | 58.46 |
| | | HK.566/31/10/22/C.Tpk-14, Tanggal 31 Oktober 2014 | | Addendum III | 135.393.011 | | |
| | | | | | | 77.59 | 83.11 |
| 8 | Pembangunan Jalan Tembus dan Jalan Nuaranre ke Jalan Industri 1 Termasuk Supervisi dibutuhkan dana Rp. 156,35 Miliar disesuaikan menjadi Rp. 85 miliar, tahun 2015 disesuaikan kembali menjadi Rp. 80 miliar, tahun 2013 dianggarkan Rp. 25 Miliar tidak terserap, tahun 2014 dianggarkan Rp. 25 Miliar terserap Rp. 2,75 miliar, tahun 2015 dianggarkan Rp. 75,5 miliar diperkirakan terserap Rp. 16,54 miliar, tahun 2016 dianggarkan Rp. 20 miliar | HK.566/30/4/5/PTP-15 tanggal 30 April 2015 | PT. Adhi Karya (Persero) | | 40.857.080,00 | | |
| | | HK.566/4/3/1/PTP-16, Tanggal 4 Maret 2016 | | Addendum | 13.018.044,00 | | 4.31 Selesai dan Addendum Kurang |
| | | HK.566/22/1/1/PTP-16, Tanggal 22 Januari 2016 | | Addendum | 170.560 | | 38,44 Selesai dan Addendum tambah Rp. 39.400 |

Gambar IV.7 Rekap Laporan Progress Pekerjaan
Sumber : Administrasi Teknik 2016

4.4 Addendum Pekerjaan (Kerja Tambah atau Kerja Kurang)

Addendum pekerjaan dilakukan ketika pekerjaan realisasinya tidak berjalan sesuai dengan perencanaan yang ada sebelumnya, baik itu dari volume ataupun dari waktu pekerjaan. Addendum dibuat berdasarkan laporan mingguan yang sudah dievaluasi. Pembuatan addendum melibatkan atasan, para pengawas dan kontraktor yang melakukan pekerjaan. Ketika addendum sudah dibuat dan disetujui dengan tandatangan atasan, para pengawas dan kontraktor kemudian terbit berita acara addendum, dari berita acara addendum tersebut kontraktor bisa melanjutkan kembali pekerjaannya, baik itu pekerjaan tambah ataupun pekerjaan kurang.



BERITA - ACARA
No. UM. 339 / 7 / 8 / 6 / PTP-16
Tentang

HASIL EVALUASI BERSAMA BIAYA KERJA TAMBAH KURANG DAN PERPANJANGAN WAKTU PELAKSANAAN ADDENDUM KE - 2 PADA PEKERJAAN PEMBANGUNAN LAPANGAN PENUMPUKAN EX. GUDANG 005, PENINGKATAN LAPANGAN PENUMPUKAN KETEL UAP, PERBAIKAN LAPANGAN PENUMPUKAN DI EX. INGGOM, PEMBANGUNAN LAPANGAN SISI BARAT POS 1, PEMBANGUNAN JALAN TEMBUS DARI JALAN NUSANTARA KE JALAN INDUSTRI 1 TAHAP 2 (LAPANGAN SISI UTARA JALAN TEMBUS) DI PELABUHAN TANJUNG PRIOK

Pada hari *Senin* tanggal *Tujuh* bulan *Maret* tahun *Dua Ribu Enam Belas*, Kami yang bertanda tangan di bawah ini :

| | |
|----------------------------|--|
| 1. Endra Sutanto | Manager Teknik Sipil |
| 2. Eko Sugito | Asman Penilikan |
| 3. Ika Oktania | Asman Estimasi dan Desain |
| 4. Andrianto Rahmadha | Pengawas Lapangan |
| 5. Agung Yudhantara | Pengawas Lapangan |
| 6. Muhammad Luthfi | Pengawas Lapangan |
| 7. Giita Winda Rahayu | Pengawas Lapangan |
| 8. Afid Nursyalam | Pengawas Lapangan |
| 9. Agung Prastio | Pengawas Lapangan |
| 10. Arie Rachmadi Djunaedi | Pengawas Lapangan |
| 11. Teguh Waluyo | Pengawas Lapangan |
| 12. Indrawan Dunar P. | Pengawas Lapangan |
| 13. Ferdy Hostiadi M. | Pengawas Lapangan |
| 14. Reza Ismail | Pengawas Lapangan |
| 15. Arief Rachman Hakim | Pengawas Lapangan |
| 17. Fatma Dewi Setyowati | Project Manager PT. Hutama Karya (Penero) |
| 18. Dwi Fatkhurohman | Site Engineering Manager PT. Hutama Karya (Penero) |
| 19. Yayat Hidayat | Team Leader PT. LAPI Ganeshatama Consulting |
| 20. Indra Cahya Purnama | Site Engineer PT. LAPI Ganeshatama Consulting |

Berdasarkan :

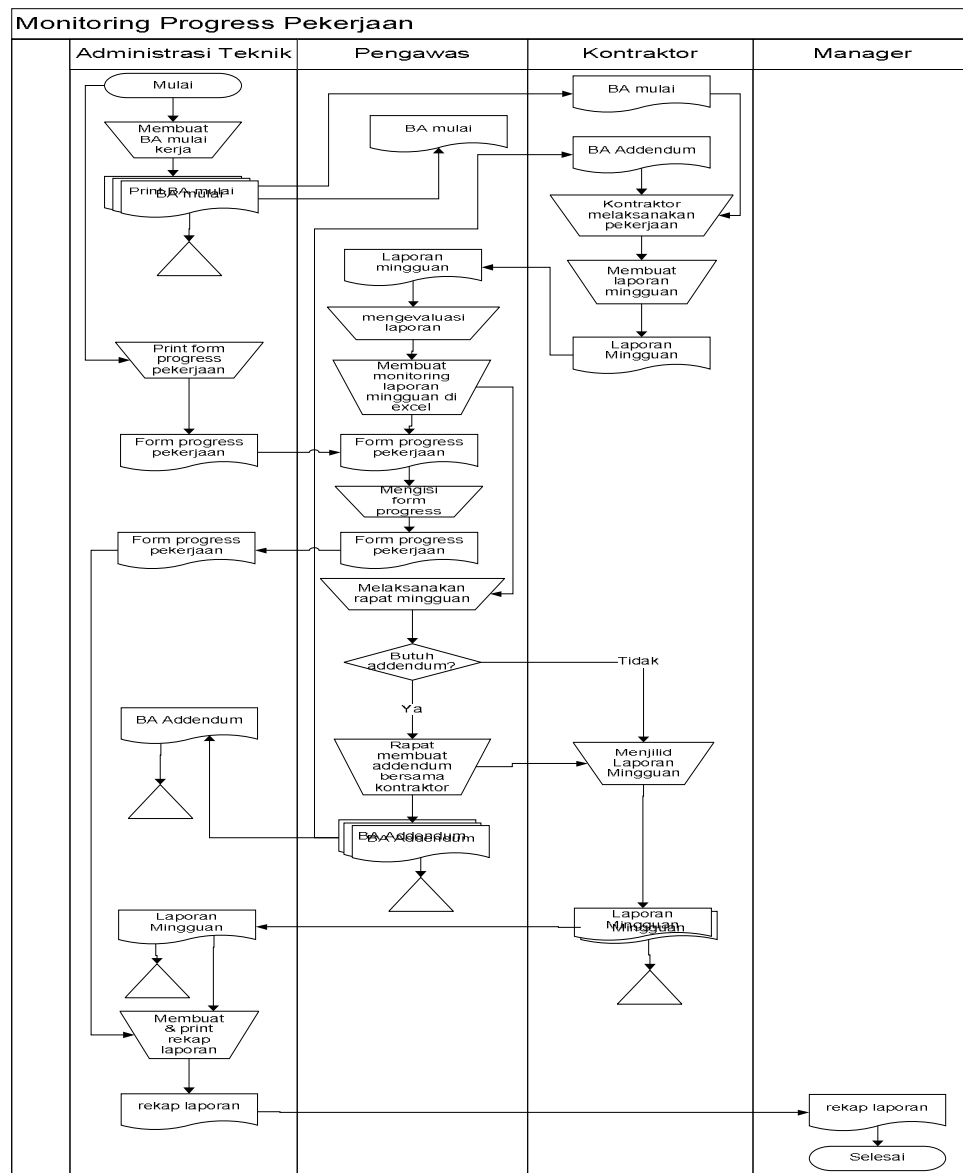
- a. Surat Perjanjian Nomor : HK.566 / 27 / 8 / 1 / PTP - 15 tanggal 27 Agustus 2015, tentang Pembangunan Lapangan Penampukan Ex. Gudang 005 Pelabuhan Tanjung Priok, Peningkatan Lapangan Penampukan Ketel Uap di Pelabuhan Tanjung Priok, Perbaikan Lapangan Penampukan di Ex. Inggom di Pelabuhan Tanjung Priok, Pembangunan Lapangan Sisi Barat Pos 1 di Pelabuhan Tanjung Priok, dan Pembangunan Jalan Tembus dari Jalan Nusantara ke Jalan Industri 1 Tahap 2 (Lapangan Sisi Utara Jalan Tembus) di Pelabuhan Tanjung Priok.
- b. Berita Acara Nomor : UM.339 / 28 / 8 / 6 / PTP - 15 tanggal 28 Agustus 2015, tentang Mulai Pelaksanaan Pekerjaan Pembangunan Lapangan Penampukan Ex. Gudang 005 Pelabuhan Tanjung Priok, Peningkatan Lapangan Penampukan Ketel Uap di Pelabuhan Tanjung Priok, Perbaikan Lapangan Penampukan di Ex. Inggom di Pelabuhan Tanjung Priok, Pembangunan

Gambar IV.8 Berita Acara Addendum Pekerjaan
Sumber : Administrasi Teknik 2016

4.5 Sistem yang Berjalan

Untuk mengetahui sistem yang berjalan dilakukan pengamatan bagaimana cara kerja sistem yang digunakan di PT Pelabuhan Tanjung Priok khususnya mengenai sistem laporan progress pekerjaan di divisi teknik sipil. Tujuan melakukan pengamatan sistem tersebut adalah agar dapat mengidentifikasi masalah-masalah yang ada pada sistem laporan progress pekerjaan sehingga dapat melakukan saran-saran perbaikan. Berikut beberapa flowmap sistem yang berjalan pada divisi teknik sipil

1. Sistem Monitoring *Progress* Pekerjaan



Gambar IV.9 Flowmap Sistem Monitoring *Progress* Pekerjaan

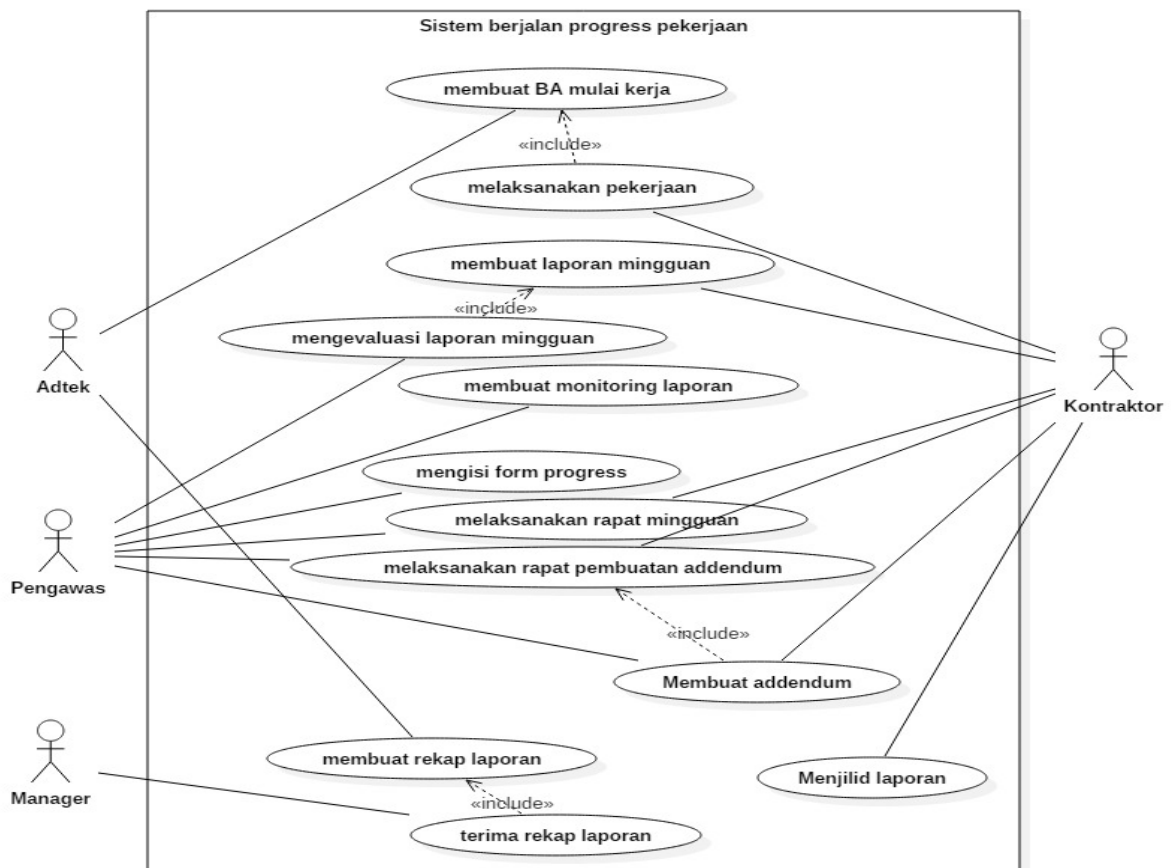
Sumber : Analisis 2016

Penjelasan sistem monitoring *progress* pekerjaan sebagai berikut:

- a. Pertama di bagian administrasi teknik membuat berita acara mulai pekerjaan, Berita Acara (BA) mulai pekerjaan sebagai dasar awal kontraktor memulai pekerjaannya. Berita acara mulai pekerjaan disampaikan ke kontraktor, dan kontraktor bisa mulai pekerjaannya sesuai tanggal yang sudah ditetapkan di berita acara tersebut.
- b. Ketika kontraktor pekerjaannya sudah berjalan, pekerjaannya akan diawasi oleh pengawas yang sudah ditunjuk dari divisi teknik sipil. Kontraktor juga harus menyampaikan laporan mingguan mengenai kegiatan mereka selama pekerjaan berlangsung.
- c. Kontraktor menyampaikan laporan mingguan pengawas, kemudian dicek kelengkapannya dan kemudian dievaluasi. Dari evaluasi tersebut kemudian dilaporkan hasilnya ke atasan langsung untuk diminta keputusan selanjutnya.
- d. Setelah laporan mingguan selesai di cek, laporan mingguan dikembalikan kekontraktor untuk dijilid.
- e. Selanjutnya laporan mingguan yang sudah dijilid diberikan kembali ke administrasi teknik untuk diarsipkan.
- f. Kemudian bagian administrasi teknik membuat rekap laporan seluruh pekerjaan lalu rekap tersebut di print dan diberikan kepada atasan sebagai laporan.

4.6 Use Case Diagram Sistem Saat Ini

Use case diagram menggambarkan siapa saja aktor yang melakukan prosedur dalam sistem serta fungsi-fungsi (proses) yang terlibat dalam transformasi pada sistem tersebut. *Use case diagram* sistem monitoring *progress* pekerjaan saat ini pada divisi teknik sipil dapat dilihat pada Gambar 4.7.



Gambar IV.10 Use Case Monitoring Sistem Berjalan
Sumber : Analisis 2016

Tabel IV.1 Use Case Description Membuat BA Mulai Kerja

| | |
|---------------------|--|
| Nama Use Case | Membuat BA mulai kerja |
| Deskripsi Use Case | Use case ini menggambarkan pembuatan surat Berita Acara (BA) mulai pekerjaan, guna dari BA ini adalah sebagai dasar kontraktor melaksanakan pekerjaan. |
| Aktor | Adtek |
| Relationship | |
| Normal Flow Events: | <ol style="list-style-type: none"> 1. Adtek membuat berita acara mulai pekerjaan. 2. Adtek memberikan BA mulai pekerjaan ke kontraktor. 3. Pengawas menerima tembusan BA mulai pekerjaan. |

Tabel IV.2 Use Case Description Melaksanakan Pekerjaan

| | |
|--------------------|--|
| Nama Use Case | Melaksanakan Pekerjaan |
| Deskripsi Use Case | Use case ini menggambarkan kontraktor melaksanakan |

Tabel IV.2 *Use Case Description* Melaksanakan Pekerjaan (Lanjutan)

| | |
|----------------------------|--|
| <i>Case</i> | pekerjaan. |
| Aktor | Kontraktor. |
| <i>Relationship</i> | |
| <i>Normal Flow Events:</i> | 1. Kontraktor melaksanakan pekerjaannya. |

Tabel IV.3 *Use Case Description* Membuat Laporan Mingguan

| | |
|----------------------------|---|
| Nama <i>Use Case</i> | Membuat Laporan Mingguan. |
| Deskripsi <i>Use Case</i> | <i>Use case</i> ini menggambarkan pembuatan laporan mingguan oleh kontraktor. |
| Aktor | Kontraktor. |
| <i>Relationship</i> | |
| <i>Normal Flow Events:</i> | 1. Kontraktor membuat laporan mingguan pekerjaan. 2. Kontraktor menyerahkan laporan mingguan ke adtek. |

Tabel IV.4 *Use Case Description* Mengevaluasi Laporan

| | |
|----------------------------|--|
| Nama <i>Use Case</i> | Mengevaluasi laporan. |
| Deskripsi <i>Use Case</i> | <i>Use case</i> ini menggambarkan pengawas mengevaluasi laporan mingguan. |
| Aktor | Pengawas. |
| <i>Relationship</i> | |
| <i>Normal Flow Events:</i> | 1. Pengawas mengevaluasi laporan mingguan. 2. Pengawas diskusi dengan kontraktor mengenai laporan mingguan. |

Tabel IV.5 *Use Case Description* Membuat Monitoring Laporan

| | |
|----------------------------|---|
| Nama <i>Use Case</i> | Membuat Monitoring Laporan. |
| Deskripsi <i>Use Case</i> | <i>Use case</i> ini menggambarkan pengawas membuat monitoring laporan mingguan. |
| Aktor | Pengawas. |
| <i>Relationship</i> | |
| <i>Normal Flow Events:</i> | 1. Pengawas membuat monitoring laporan mingguan di excel. |

Tabel IV.6 *Use Case Description Mengisi Form Progress*

| | |
|----------------------------|--|
| Nama <i>Use Case</i> | Mengisi Form Progress. |
| Deskripsi <i>Use Case</i> | <i>Use case</i> ini menggambarkan kontraktor mengisi <i>form progress</i> . |
| Aktor | Pengawas |
| <i>Relationship</i> | |
| <i>Normal Flow Events:</i> | 1. Pengawas mengisi <i>form progress</i> pekerjaan yang diberikan oleh bagian administrasi teknik. |

Tabel IV.7 *Use Case Description Melaksanakan Rapat Mingguan*

| | |
|----------------------------|--|
| Nama <i>Use Case</i> | Melaksanakan Rapat Mingguan. |
| Deskripsi <i>Use Case</i> | <i>Use case</i> ini menggambarkan pengawas dan kontraktor melaksanakan rapat mingguan. |
| Aktor | Pengawas dan Kontraktor. |
| <i>Relationship</i> | |
| <i>Normal Flow Events:</i> | 1. Pengawas dan kontraktor melaksanakan rapat mingguan . |

Tabel IV.8 *Use Case Description Melaksanakan Rapat Pembuatan Addendum*

| | |
|----------------------------|---|
| Nama <i>Use Case</i> | Melaksanakan Rapat Pembuatan Addendum. |
| Deskripsi <i>Use Case</i> | <i>Use case</i> ini pengawas dan kontraktor rapat pembuatan addendum. |
| Aktor | Pengawas dan Kontraktor |
| <i>Relationship</i> | |
| <i>Normal Flow Events:</i> | 2. Pengawas dan Kontraktor rapat pembahasan pembuatan addendum. |

Tabel IV.9 *Use Case Description Membuat Addendum*

| | |
|---------------------------|---|
| Nama <i>Use Case</i> | Membuat Addendum. |
| Deskripsi <i>Use Case</i> | <i>Use case</i> ini menggambarkan pengawas dan kontraktor membuat addendum. |

Tabel IV.9 *Use Case Description* Membuat Addendum (Lanjutan)

| | |
|----------------------------|---|
| Aktor | Pengawas, Kontraktor |
| <i>Relationship</i> | |
| <i>Normal Flow Events:</i> | 2. Pengawas dan kontraktor rapat pembahasan pembuatan addendum. |

Tabel IV.10 *Use Case Description* Menjilid Laporan

| | |
|----------------------------|---|
| Nama <i>Use Case</i> | Menjilid Laporan. |
| Deskripsi <i>Use Case</i> | <i>Use case</i> ini menggambarkan kontraktor menjilid laporan mingguan. |
| Aktor | Kontraktor. |
| <i>Relationship</i> | |
| <i>Normal Flow Events:</i> | 1. Pengawas membuat monitoring laporan mingguan di excel. |

Tabel IV.11 *Use Case Description* Membuat Rekap Laporan

| | |
|----------------------------|---|
| Nama <i>Use Case</i> | Membuat Rekap Laporan. |
| Deskripsi <i>Use Case</i> | <i>Use case</i> ini menggambarkan adtek membuat rekap laporan pekerjaan. |
| Aktor | Adtek. |
| <i>Relationship</i> | |
| <i>Normal Flow Events:</i> | 1. Adtek membuat rekap laporan mingguan dan memberikannya ke Manager sebagai laporan. |

Tabel IV.12 *Use Case Description* Terima Rekap Laporan

| | |
|----------------------------|--|
| Nama <i>Use Case</i> | Terima Rekap Laporan. |
| Deskripsi <i>Use Case</i> | <i>Use case</i> ini menggambarkan manager menerima rekap laporan dari adtek. |
| Aktor | Manager. |
| <i>Relationship</i> | |
| <i>Normal Flow Events:</i> | 1. Manager menerima rekap laporan yang dibuat oleh adtek. |

BAB V

ANALISIS DAN PEMBAHASAN

5.1 Analisis Kebutuhan Sistem

Analisis kebutuhan sistem dilakukan untuk mendapatkan informasi kebutuhan sistem, dengan mengetahui kebutuhan sistem diharapkan dapat diusulkan perbaikan terhadap sistem yang diamati. Berdasarkan hasil pengolahan data, maka diidentifikasi dibutuhkannya suatu aplikasi untuk menunjang proses pengawasan pekerjaan di PT Pelabuhan Tanjung Priok. Dalam merancang aplikasi ini, tahap pengidentifikasian kebutuhan sistem merupakan tahapan yang perlu dilakukan agar aplikasi yang dibuat dapat menjawab permasalahan yang ada.

Berikut adalah kebutuhan sistem yang diidentifikasi untuk sistem informasi *tracking project*:

Tabel V.1 Kebutuhan Sistem

| Kebutuhan Sistem | |
|-------------------------------------|---|
| <i>Project Name</i> | Sistem Informasi Project Tracking |
| <i>Business Need</i> | Terwujudnya sebuah sistem yang dapat membantu mengawasi seluruh jalannya pekerjaan proyek. |
| <i>Business Requirement</i> | Memberikan sistem informasi pengawasan proyek yang dapat membantu dalam mengawasi jalannya proyek. |
| <i>Business Value</i> | <ol style="list-style-type: none"> 1. Membantu pengawas dalam mengevaluasi laporan mingguan 2. Membantu pengawas dalam menginformasikan nilai kontrak suatu pekerjaan 3. Membantu pengawas dalam mengumpulkan bahan – bahan yang digunakan untuk membuat addendum pekerjaan. |
| <i>Special Issues or Constrains</i> | Pembuatan sistem dilakukan selama enam bulan. |

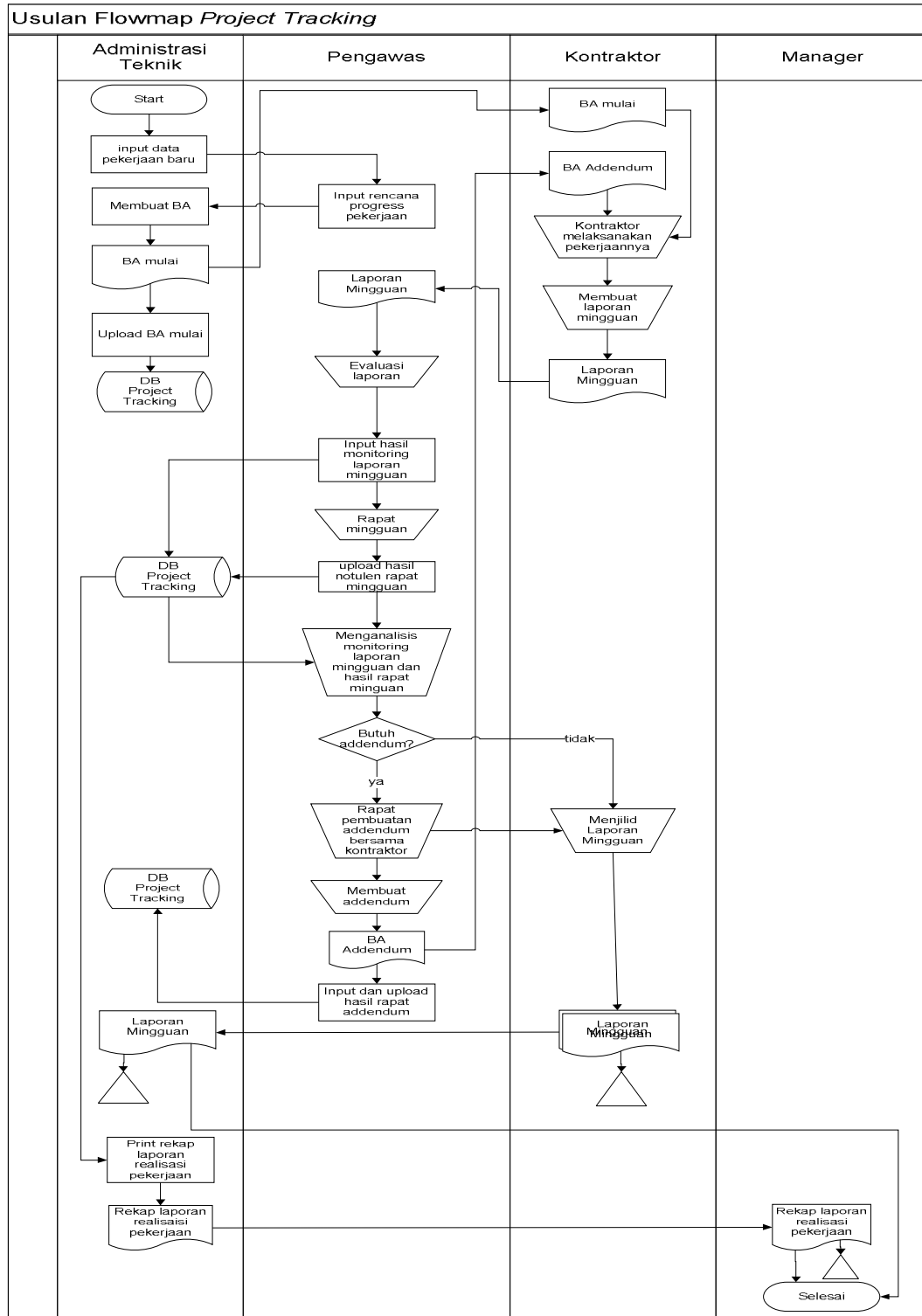
(Sumber: Analisis Data, 2016)

5.2 Prosedur *Project Tracking* Usulan

Prosedur *Tracking Project* yang diusulkan melalui tahapan sebagai berikut:

1. Administrasi Teknik membuat Berita Acara (BA) mulai kerja dan melakukan login kedalam sistem lalu membuat pekerjaan baru di aplikasi dengan menu tambah pekerjaan, kemudian mengisi field-field yang sudah disediakan sesuai kontrak dari pekerjaan tersebut.
2. Pengawas mengisi rencana progress pekerjaan di pekerjaan yang sudah dibuat oleh bagian administrasi teknik.
3. Kontraktor melaksanakan pekerjaan dan membuat laporan mingguan yang berdasarkan *progress* pekerjaan yang dilaksanakan.
4. Laporan mingguan diberikan kepada pengawas untuk di evaluasi. Setelah di evaluasi hasilnya di input kedalam sistem.
5. Setiap minggu ada jadwal rapat mingguan antara kontraktor dengan pengawas yang membahas tentang pekerjaan yang sedang dilaksanakan. Hasil rapat tersebut kemudian diinput kedalam sistem.
6. Jika didalam hasil rapat mingguan tersebut ada pekerjaan yang membutuhkan addendum, akan dilakukan rapat kembali untuk membahas pembuatan addendum.
7. Hasil addendum tersebut kemudian diinput kedalam sistem, dan bagian pengawas membuat berita acara addendum.
8. Setelah addendum dibuat, kontraktor melaksanakan pekerjaan dan membuat laporan mingguan kembali.
9. Pengawas mengevaluasi laporan mingguan yang diberikan kontraktor dan dievaluasi.
10. Ketika sudah tidak ada pekerjaan yang membutuhkan addendum, laporan dikembalikan kepada pengawas untuk dijilid.
11. Setelah dijilid laporan mingguan diberikan kepada administrasi teknik untuk kemudian di arsipkan.
12. Bagian administrasi teknik print rekap laporan bulanan, dan memberikannya kepada manager sebagai laporan.

Seperti yang dapat dilihat pada Gambar V.1 adalah *Flowmap* Sistem *Project Tracking* Usulan sebagai berikut:



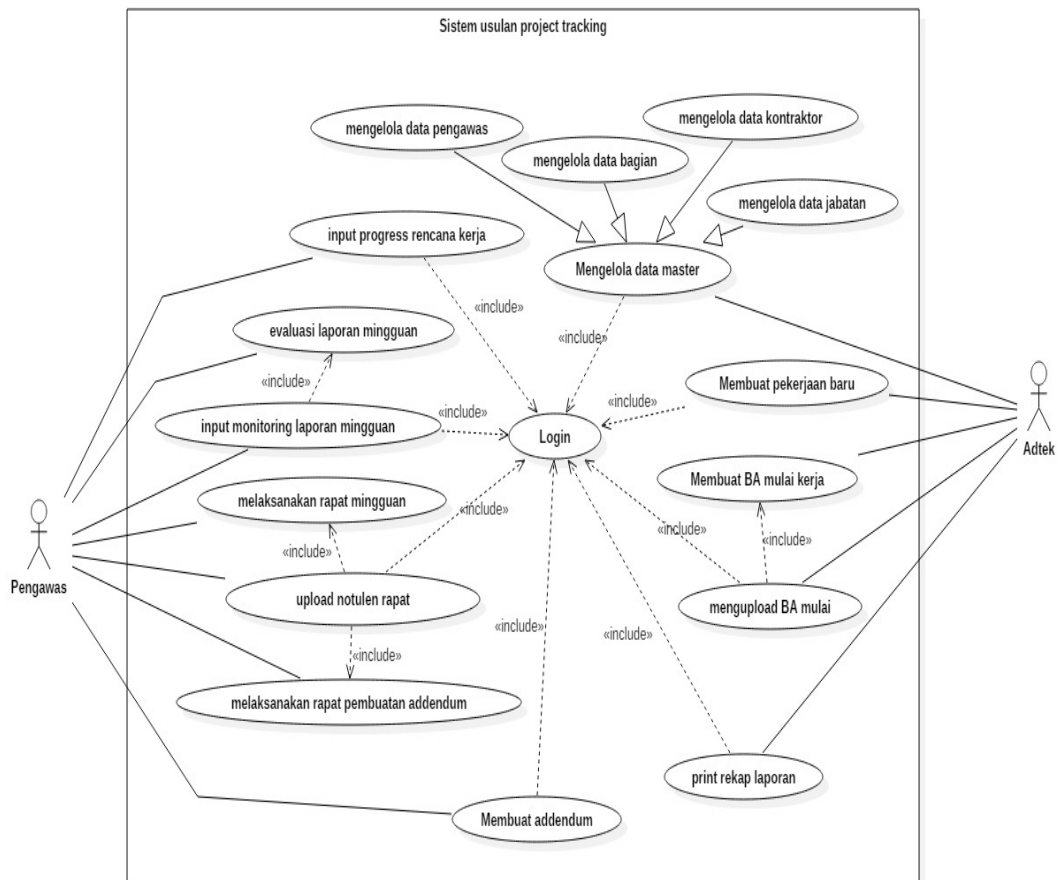
Gambar V.1 *Flowmap* Sistem *Project Tracking* Usulan
Sumber: Hasil Analisis Data (2016)

5.3 Analisis dan Perancangan Sistem Usulan

Tahap ini adalah tahap kedua pada metodologi *prototipe evolusioner* yaitu membuat *prototipe* untuk model sistem setelah sebelumnya melakukan tahap mengidentifikasi kebutuhan. Analisis proses sistem informasi menggunakan *tools* pemodelan sistem UML (*unified modeling language*) yang meliputi pembuatan *use case diagram*, *activity diagram*, *sequence diagram*, *deployment diagram*, *class diagram* dan juga pembuatan kamus data. Tahapan analisis ini akan memberikan gambaran mengenai aliran informasi dan data pada sistem informasi yang akan dibangun. Selanjutnya akan dilakukan perancangan sistem informasi *project tracking* berdasarkan hasil analisis.

5.3.1 Use Case Diagram

Use case diagram berfungsi untuk mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem yang akan dibuat. Dengan kata lain *use case diagram* juga digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem dan siapa saja yang berhak menggunakan fungsi-fungsi itu. Berikut adalah *use case diagram* sistem informasi *project tracking* yang diusulkan dapat dilihat pada Gambar V.2.



Gambar V.2. Use Case Diagram Sistem Informasi Project Tracking Usulan

Sumber: Hasil Analisis (2016)

Di dalam *Use Case Diagram*, sebuah aktor tidak hanya untuk mewakili orang tetapi juga untuk mewakili proses atau sistem lain yang berinteraksi dengan sistem. Oleh karena itu, tahapan selanjutnya yang perlu dilakukan yaitu mendefinisikan aktor-aktor apa saja yang terkait dalam sistem dan menjelaskan peran atau siapa aktor tersebut. Pendefinisian aktor pada *Use Case Diagram* Gambar V.2 dapat dilihat pada Tabel V.2.

Tabel V.2 Definisi Aktor *Use Case Diagram* yang Diusulkan

| No | Actor | Description |
|----|---------------------|--|
| 1. | Administrasi Teknik | User yang membuat pekerjaan baru dan mengarsipkan surat-surat yang terkait dengan pekerjaan. |
| 2. | Pengawas | User yang mengawasi jalannya proyek-proyek |

Tabel V.2 Definisi Aktor *Use Case Diagram* yang Diusulkan (Lanjutan)

| No | Actor | Description |
|----|-------|---|
| | | pekerjaan baik itu laporan mingguan dari kontraktor maupun inspeksi kelapangan. |

Sumber: Hasil Analisis (2016)

Setelah mendefinisikan aktor yang terkait dalam *use case Diagram*, tahap selanjutnya yaitu mendefinisikan *use case* itu sendiri. *Use case* ini merupakan suatu fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor yang dinyatakan dengan menggunakan kata kerja di awal frase nama *use case*. Pendefinisian *use case* pada *use case diagram* Gambar V.2. dapat dilihat pada Tabel V.3.

Tabel V.3 *Use Case Description Login*

| Nama Use Case | Login |
|---------------------|---|
| Deskripsi Use Case | <i>Use case</i> ini menggambarkan proses awal untuk masuk ke dalam aplikasi dengan mengisi <i>username</i> dan <i>password</i> . |
| Aktor | Administrasi Teknik dan Pengawas. |
| Normal Flow Events: | <ol style="list-style-type: none"> 1. <i>User</i> mengisi <i>username</i> dan <i>password</i>. 2. Jika <i>username</i> dan <i>password</i> salah maka akan muncul pesan kesalahan. 3. Apabila <i>username</i> dan <i>password</i> benar maka <i>user</i> akan masuk ke dalam Aplikasi <i>Project Tracking</i>. |

Sumber: Hasil Analisis Data (2016)

Tabel V.4. *Use Case Description* Mengelola Data Master

| Nama Use Case | Mengelola Data Master |
|---------------------|--|
| Deskripsi Use Case | <i>Use case</i> ini menggambarkan mengelola data master. |
| Aktor | Administrasi Teknik |
| Normal Flow Events: | <ol style="list-style-type: none"> 1. <i>User login</i> ke dalam aplikasi sesuai dengan hak akses yang telah diberikan. 2. <i>User</i> mengelola data master yang ada seperti: data pengawas, data kontraktor, data bagian dan data jabatan. |

Sumber: Hasil Analisis Data (2016)

Tabel V.5. *Use Case Description* Membuat Pekerjaan Baru

| Nama Use Case | Membuat Pekerjaan Baru |
|----------------------|--|
| Deskripsi Use Case | Use case ini menggambarkan pembuatan pekerjaan baru. |
| Aktor | Administrasi Teknik |
| Normal Flow Events: | <ol style="list-style-type: none"> 1. User login ke dalam aplikasi sesuai dengan hak akses yang telah diberikan. 2. User membuat pekerjaan baru dan menginput field – field yang sudah disediakan. |

Sumber: Hasil Analisis Data (2016)

Tabel V.6. *Use Case Description* Membuat BA Mulai Kerja

| Nama Use Case | Membuat BA Mulai Kerja |
|----------------------|--|
| Deskripsi Use Case | Use case ini menggambarkan pembuatan berita acara (BA) mulai kerja sebagai dasar kontraktor melaksanakan pekerjaannya. |
| Aktor | Administrasi Teknik. |
| Normal Flow Events: | <ol style="list-style-type: none"> 1. Administrasi Teknik membuat BA mulai kerja. 2. BA mulai kerja di tanda tangani oleh pengawas dan kontraktor. 3. BA mulai kerja diberikan kepada kontraktor. |

Sumber: Hasil Analisis Data (2016)

Tabel V.7. *Use Case Description* Input Progress Rencana Kerja

| Nama Use Case | Input Progress Rencana Kerja |
|----------------------|---|
| Deskripsi Use Case | Use case ini menggambarkan pengawas menginput progress rencana kerja. |
| Aktor | Pengawas. |
| Normal Flow Events: | <ol style="list-style-type: none"> 1. Pengawas login ke dalam aplikasi. 2. Pengawas menginput progress rencana pekerjaan. |

Sumber: Hasil Analisis Data (2016)

Tabel V.8. *Use Case Description* Evaluasi Laporan Mingguan

| Nama Use Case | Evaluasi Laporan Mingguan |
|----------------------------|--|
| Deskripsi Use Case | <i>Use case</i> ini menggambarkan pengawas mengevaluasi laporan pekerjaan dari kontraktor. |
| Aktor | Pengawas. |
| <i>Normal Flow Events:</i> | 1. Pengawas melakukan evaluasi terhadap laporan pekerjaan yang diberikan oleh kontraktor. |

Sumber: Hasil Analisis Data (2016)

Tabel V.9. *Use Case Description* Input Monitoring Laporan Mingguan

| Nama Use Case | Input Monitoring Laporan Mingguan |
|----------------------------|--|
| Deskripsi Use Case | <i>Use case</i> ini menggambarkan pengawas menginput monitoring laporan. |
| Aktor | Pengawas. |
| <i>Normal Flow Events:</i> | 1. Pengawas <i>login</i> ke dalam aplikasi. 2. Pengawas melakukan input monitoring laporan dari hasil evaluasi laporan pekerjaan yang diberikan kontraktor. |

Sumber: Hasil Analisis Data (2016)

Tabel V.10. *Use Case Description* Melaksanakan Rapat Mingguan

| Nama Use Case | Melaksanakan Rapat Mingguan. |
|----------------------------|--|
| Deskripsi Use Case | <i>Use case</i> ini menggambarkan pelaksanaan rapat mingguan. |
| Aktor | Pengawas. |
| <i>Normal Flow Events:</i> | 1. Pengawas rapat mingguan membahas progress pekerjaan bersama kontraktor. |

Sumber: Hasil Analisis Data (2016)

Tabel V.11. *Use Case Description* Melaksanakan Rapat Pembuatan Addendum

| Nama Use Case | Melaksanakan Rapat Pembuatan Addendum. |
|----------------------|---|
| Deskripsi Use Case | <i>Use case</i> ini menggambarkan pelaksanaan rapat pembuatan addendum. |
| Aktor | Pengawas. |

Tabel V.11. *Use Case Description* Melaksanakan Rapat Pembuatan Addendum (Lanjutan)

| | |
|----------------------------|---|
| <i>Normal Flow Events:</i> | 1. Pengawas rapat pembahasan pembuatan addendum pekerjaan terhadap pekerjaan yang membutuhkan addendum. |
|----------------------------|---|

Sumber: Hasil Analisis Data (2016)

Tabel V.12. *Use Case Description Upload* Notulen Rapat

| | |
|----------------------------|---|
| Nama Use Case | Upload Notulen Rapat. |
| <i>Deskripsi Use Case</i> | <i>Use case ini menggambarkan pengawas menginput data kegiatan dan mengupload notulen rapat.</i> |
| Aktor | Pengawas. |
| <i>Normal Flow Events:</i> | <ol style="list-style-type: none"> 1. Pengawas <i>login</i> ke dalam aplikasi. 2. Pengawas masuk ke menu pekerjaan, lalu klik menu kegiatan. 3. Mengisi <i>field</i> kegiatan dan mengupload file. |

Sumber: Hasil Analisis Data (2016)

Tabel V.13. *Use Case Description* Membuat Addendum

| | |
|----------------------------|--|
| Nama Use Case | Membuat Addendum |
| <i>Deskripsi Use Case</i> | <i>Use case ini menggambarkan pengawas membuat addendum</i> |
| Aktor | Pengawas |
| <i>Normal Flow Events:</i> | <ol style="list-style-type: none"> 1. Pengawas <i>login</i> ke dalam aplikasi. 2. Pengawas masuk ke menu pekerjaan, lalu klik menu addendum. 3. Mengisi <i>field</i> addendum dan simpan. |

Sumber: Hasil Analisis Data (2016)

Tabel V.14. *Use Case Description* Print Rekap Laporan

| | |
|----------------------------|--|
| Nama Use Case | Print Rekap Laporan |
| <i>Deskripsi Use Case</i> | <i>Use case ini menggambarkan adtek mencetak rekap laporan.</i> |
| Aktor | Adtek. |
| <i>Normal Flow Events:</i> | <ol style="list-style-type: none"> 1. Adtek <i>login</i> ke dalam aplikasi. 2. Adtek masuk ke menu pekerjaan, lalu klik print. |

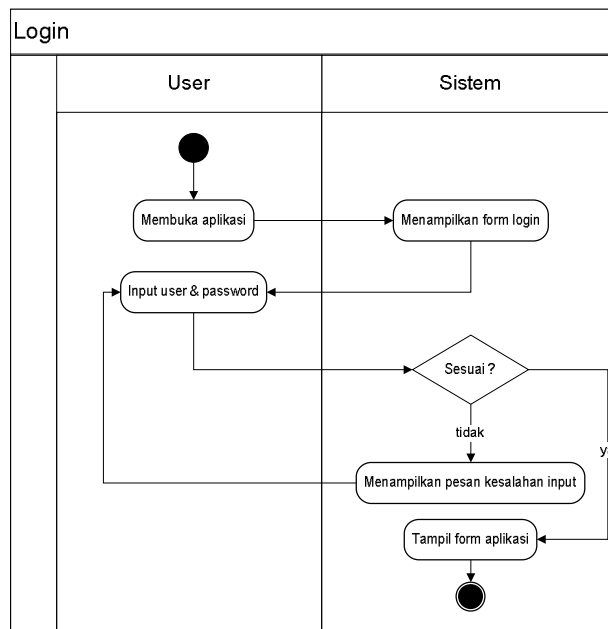
Sumber: Hasil Analisis Data (2016)

5.3.2 Activity Diagram

Penggunaan *activity diagram* bertujuan untuk menggambarkan berbagai aliran aktivitas dalam sistem yang sedang dirancang, mulai dari titik awal, melalui kondisi sampai titik akhir. Berikut adalah *activity diagram* sistem informasi *project tracking* yang diusulkan:

1. Activity Diagram Login

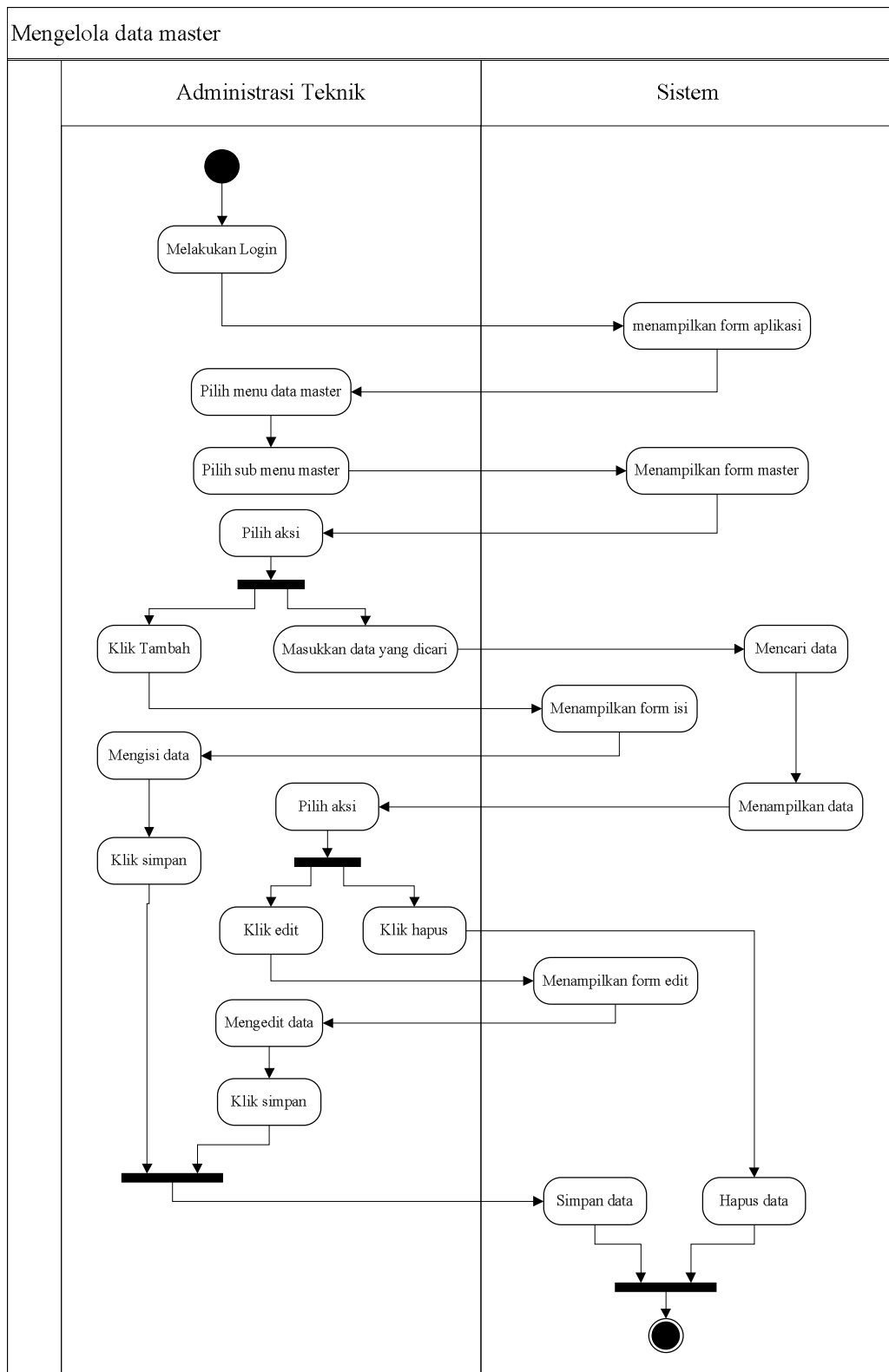
Activity Diagram pada Gambar V.3. di bawah ini merupakan *activity diagram login* yang dilakukan oleh user. Pada diagram tersebut dijelaskan bagaimana alur aktivitas login yang harus dilakukan user.



Gambar V.3. Activity Diagram Login
Sumber: Hasil Analisis (2016)

2. Activity Diagram Mengelola Data Master

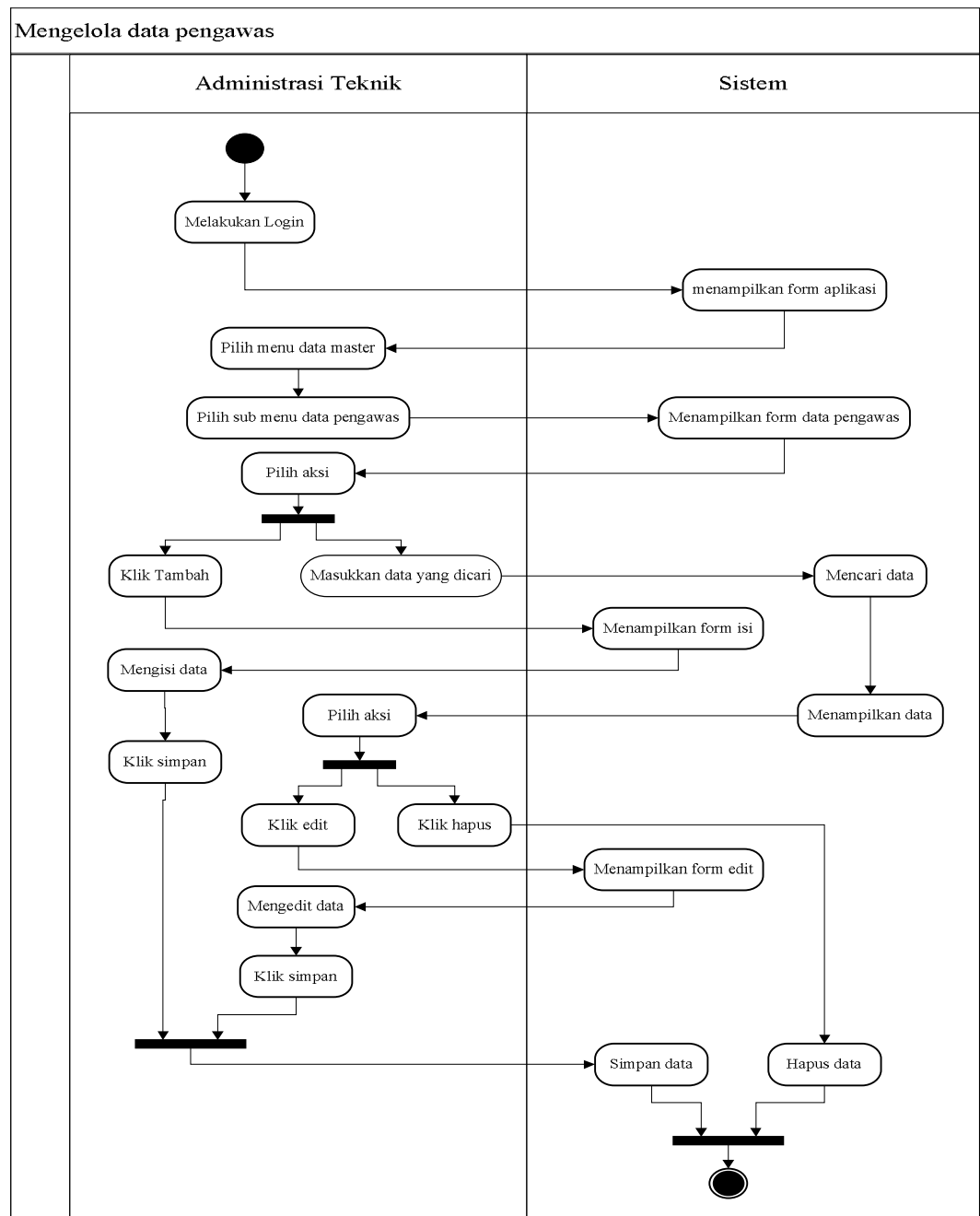
Activity Diagram pada Gambar V.4. di bawah ini merupakan *activity diagram* mengelola data master yang dilakukan oleh bagian administrasi teknik. Pada diagram tersebut dijelaskan bagaimana alur yang harus dilakukan dalam mengelola data master.



Gambar V.4. Activity Diagram Mengelola Data Master
Sumber: Hasil Analisis (2016)

3. Activity Diagram Mengelola Data Pengawas

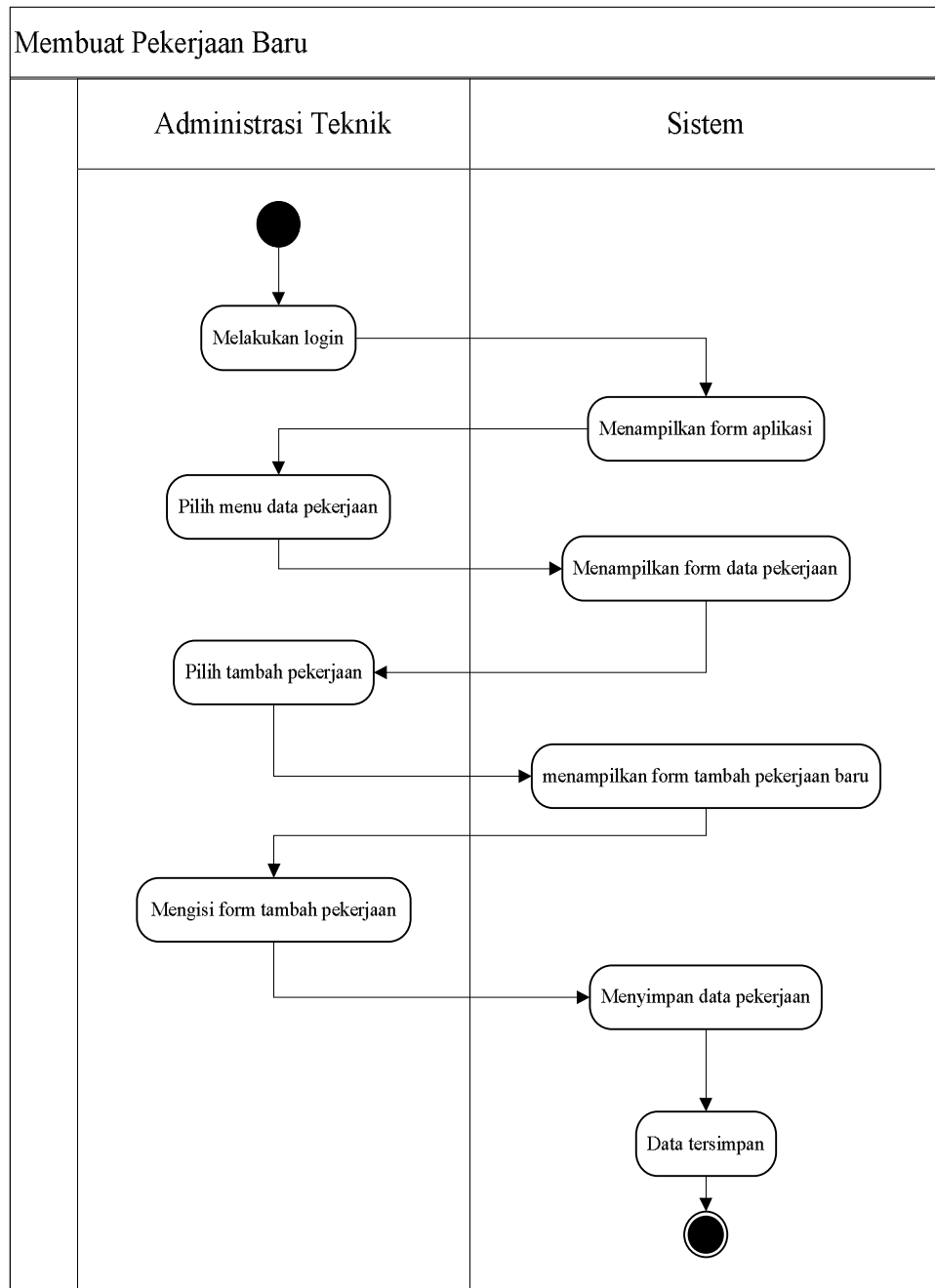
Activity Diagram pada Gambar V.5. di bawah ini merupakan *activity diagram* mengelola data pengawas yang dilakukan oleh bagian administrasi teknik. Pada diagram tersebut dijelaskan bagaimana alur yang harus dilakukan dalam mengelola data pengawas.



Gambar V.5. Activity Diagram Mengelola Data Pengawas
Sumber: Hasil Analisis (2016)

4. Activity Diagram Membuat Pekerjaan Baru

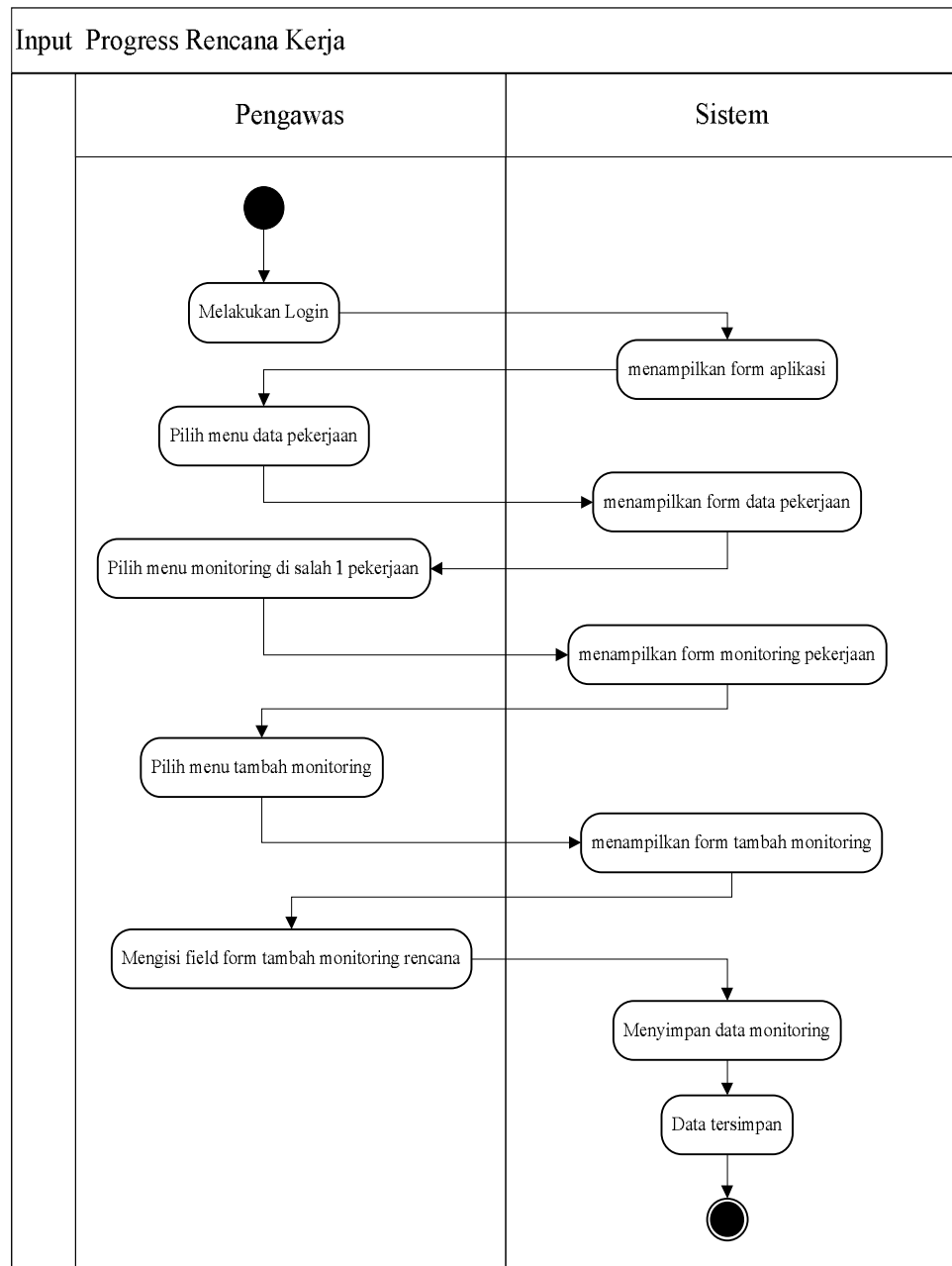
Activity Diagram pada Gambar V.5. di bawah ini merupakan activity diagram membuat pekerjaan baru yang dilakukan oleh bagian administrasi teknik. Pada diagram tersebut dijelaskan bagaimana alur yang harus dilakukan dalam pembuatan pekerjaan baru.



Gambar V.6. Activity Diagram Membuat Pekerjaan Baru
Sumber: Hasil Analisis (2016)

5. Activity Diagram Input Progress Rencana Kerja

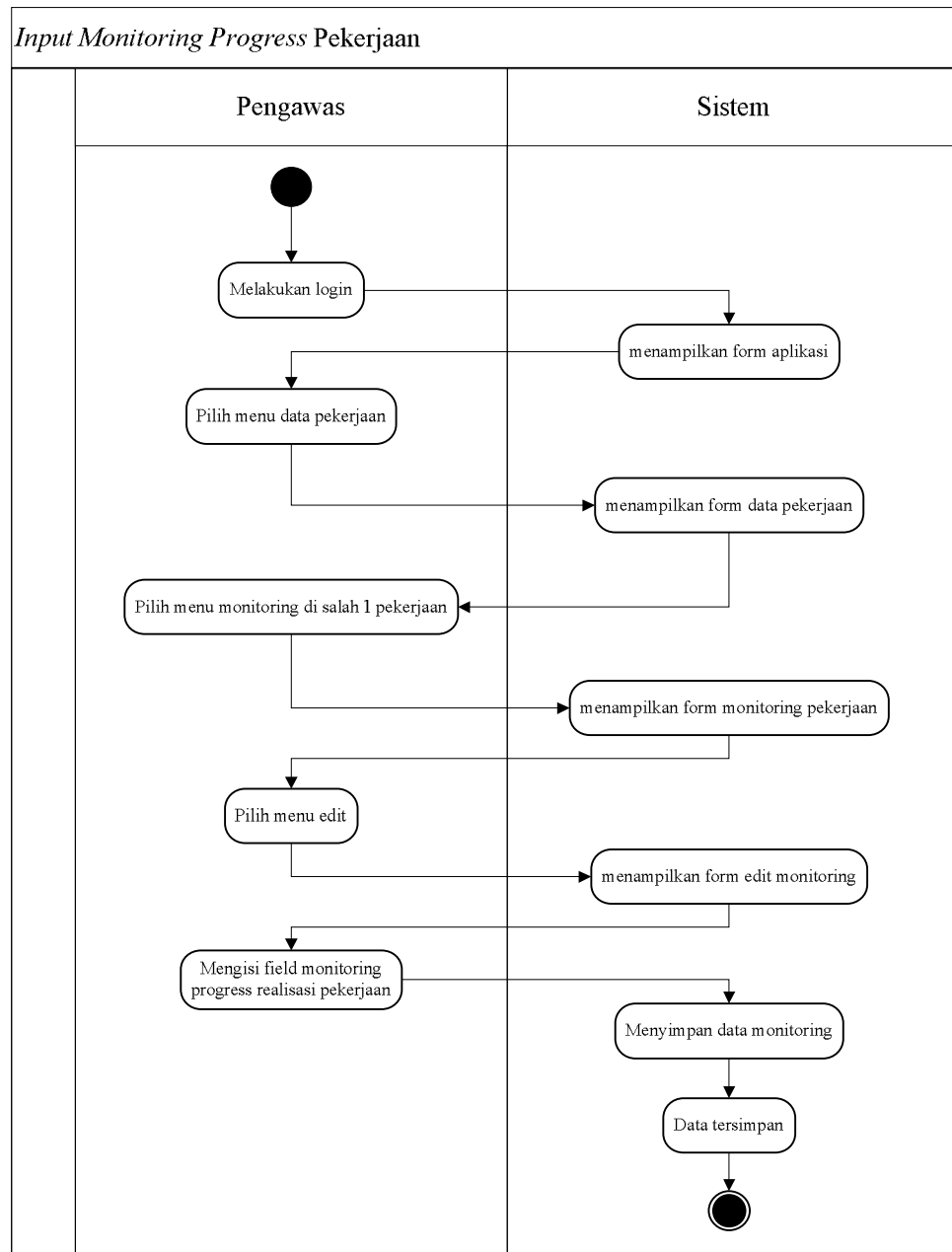
Diagram pada Gambar V.6. di bawah ini merupakan *activity diagram input progress* rencana kerja yang bisa dilakukan oleh pengawas. Pada diagram tersebut dijelaskan bagaimana alur yang harus dilakukan untuk menginput *rencana progress* pekerjaan.



Gambar V.7. Activity Diagram Input Progress Rencana Kerja
Sumber: Hasil Analisis (2016)

6. Activity Diagram Input Monitoring Progress Pekerjaan

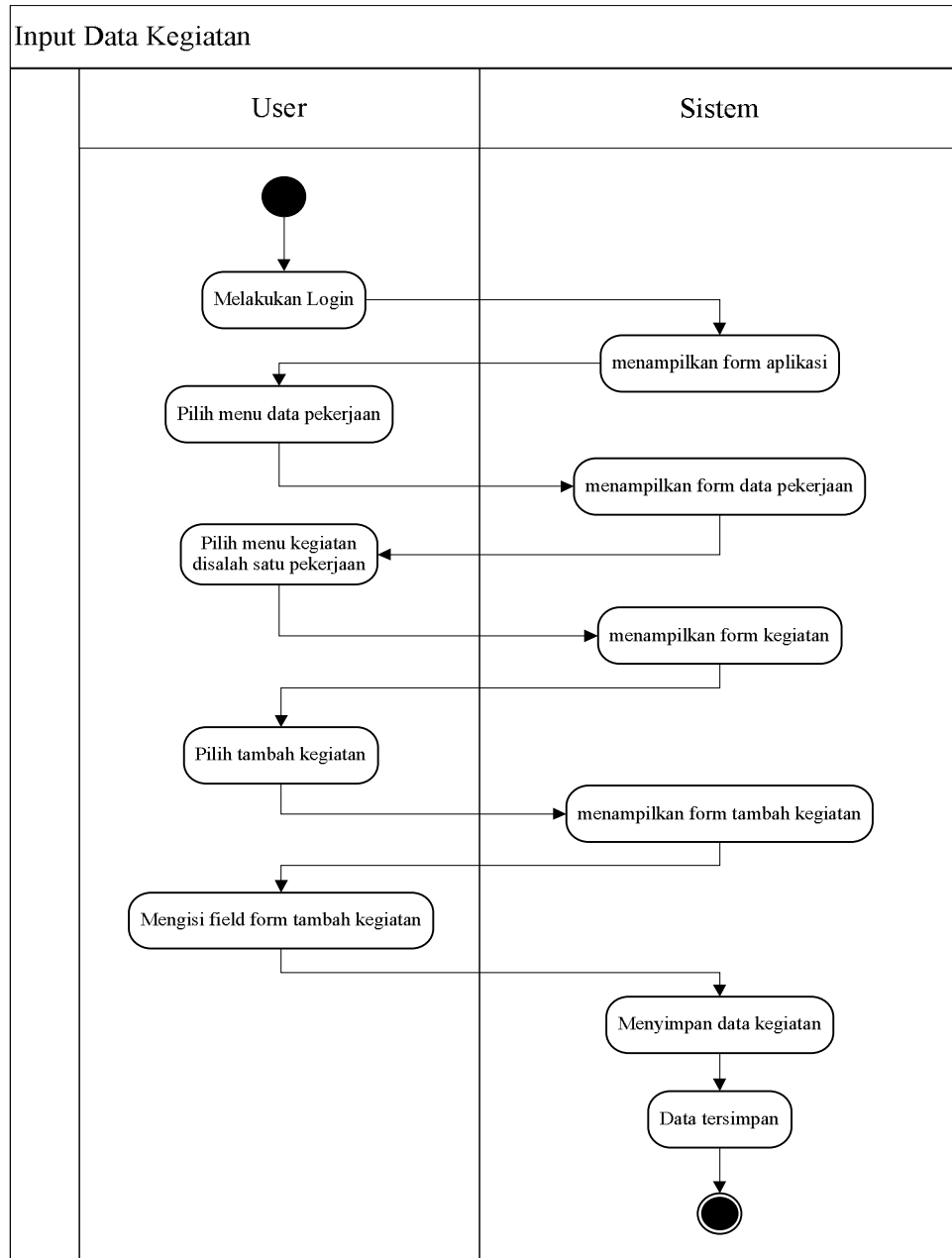
Diagram pada Gambar V.7. di bawah ini merupakan *activity diagram input monitoring progress* pekerjaan. Dalam hal ini pengawas menginput nilai *progress* pekerjaan yang sudah dievaluasi oleh pengawas pekerjaan tersebut.



Gambar V.8. Activity Diagram Input Monitoring Progress Pekerjaan
Sumber: Hasil Analisis (2016)

7. Activity Diagram Input Data Kegiatan

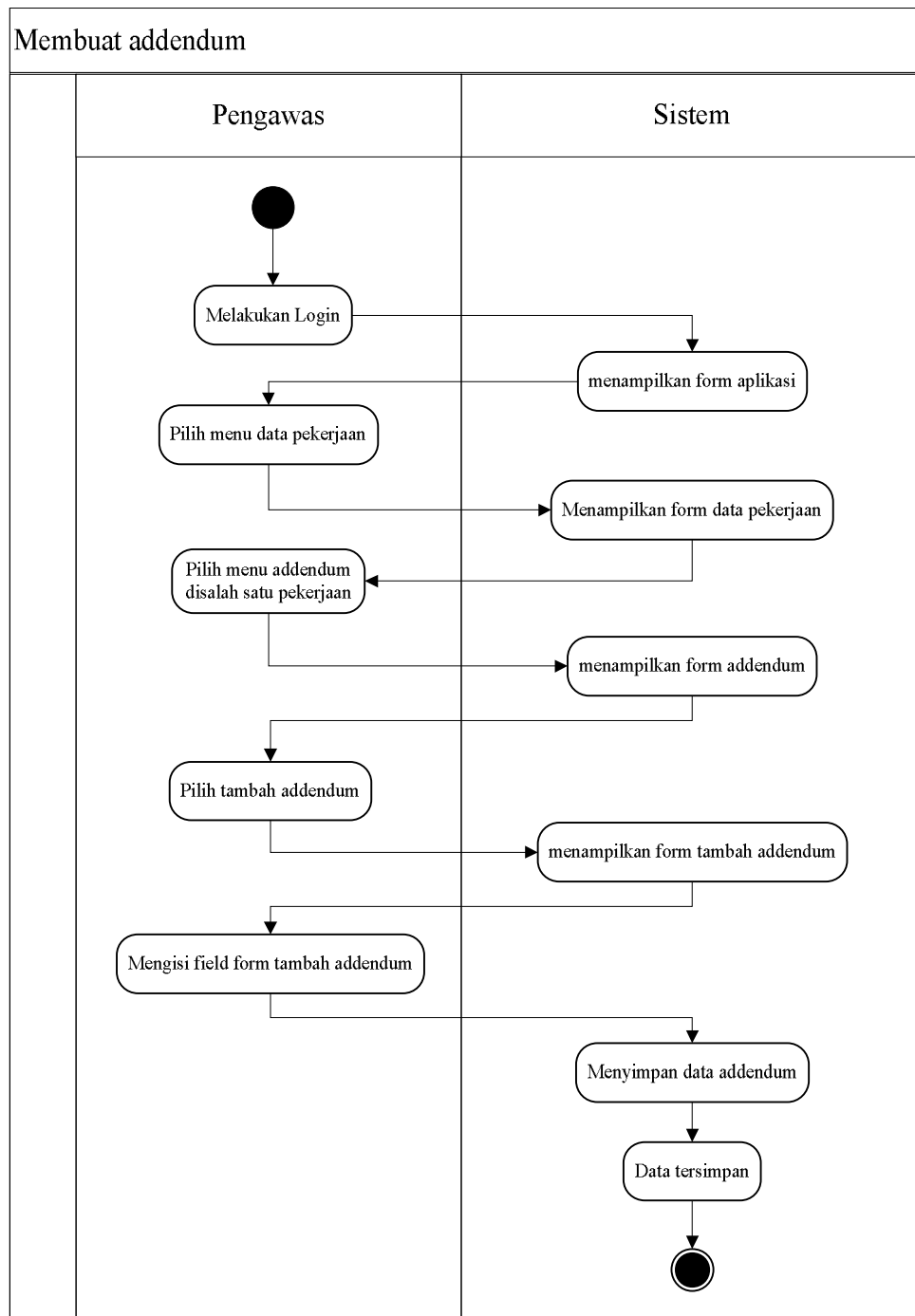
Diagram pada Gambar V.8. di bawah ini menjelaskan tentang *activity diagram* menginput data kegiatan yang salah satunya mengupload notulen rapat mingguan, dalam aktivitas ini tidak hanya notulen rapat bisa diupload, file – file yang terkait dengan pekerjaan bisa juga diupload.



Gambar V.9. Activity Diagram Input Data Kegiatan
Sumber: Hasil Analisis (2016)

8. Activity Diagram Membuat Addendum

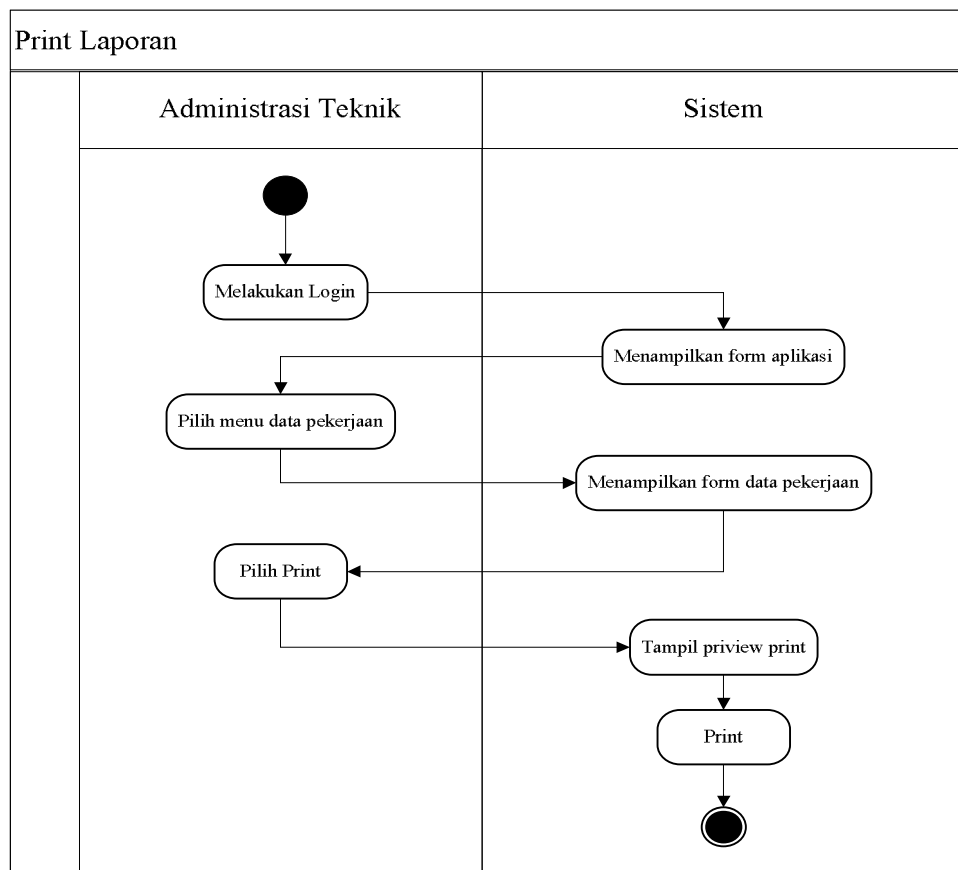
Diagram pada Gambar V.9. di bawah ini merupakan *activity diagram* membuat addendum. Aktivitas ini dilakukan untuk membuat addendum terhadap pekerjaan yang membutuhkan addendum.



Gambar V.10. Activity Diagram Membuat Addendum
Sumber: Hasil Analisis (2016)

9. Activity Diagram Print Laporan

Diagram pada Gambar V.10. di bawah ini merupakan *activity diagram* print laporan. Aktivitas ini dilakukan untuk mencetak laporan pekerjaan. Berikut *activity diagram* print laporan:



Gambar V.11. Activity Diagram Print Laporan
Sumber: Hasil Analisis (2016)

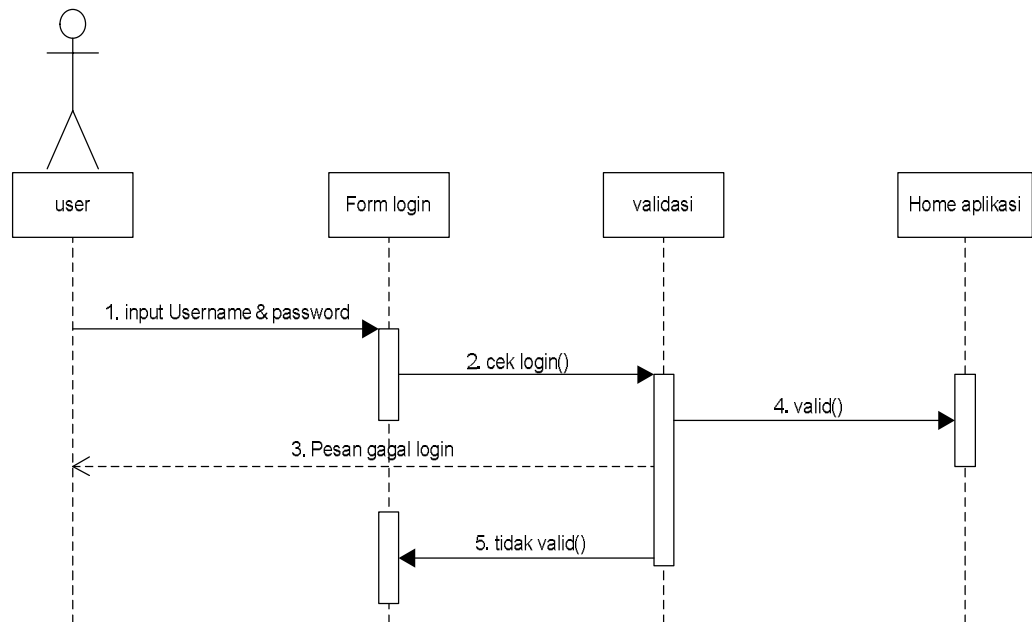
5.3.3 Sequence Diagram

Sequence diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Diagram ini menjelaskan urutan proses yang dilakukan aktor untuk melakukan suatu tugas.

1. Sequence Diagram Login

Pada Gambar V.12 menjelaskan tentang *sequence diagram login user*. Dalam proses *login* ini terdapat beberapa macam hak akses yang telah diberikan sesuai dengan fungsi masing-masing, antara lain:

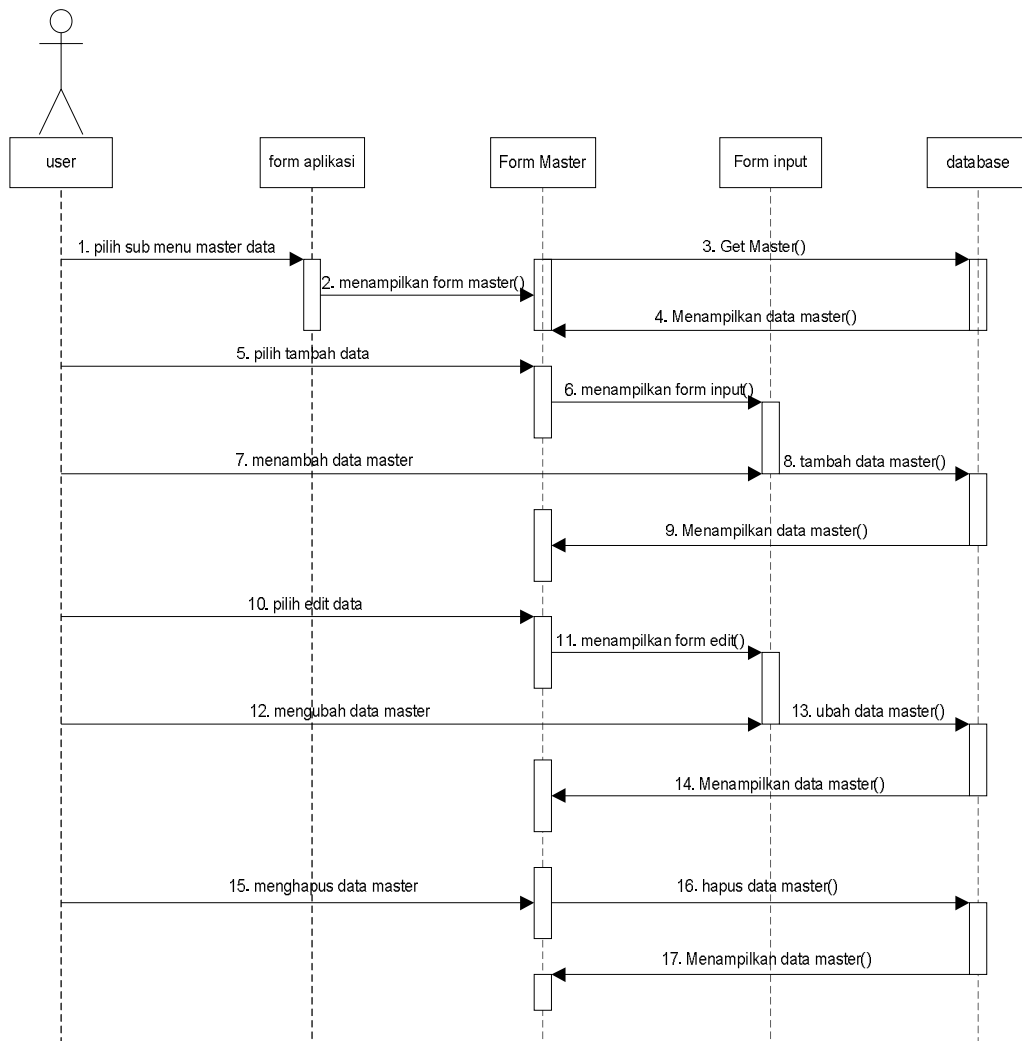
- a. Administrasi Teknik : *User* tersebut adalah yang yang memiliki hak akses untuk mengelola data master.
- b. Pengawas : *User* tersebut adalah yang melakukan monitoring didalam pekerjaan.



Gambar V.12 *Sequence Diagram* Proses Login
Sumber: Analisis Data (2016)

2. *Sequence Diagram* Mengelola Data Master

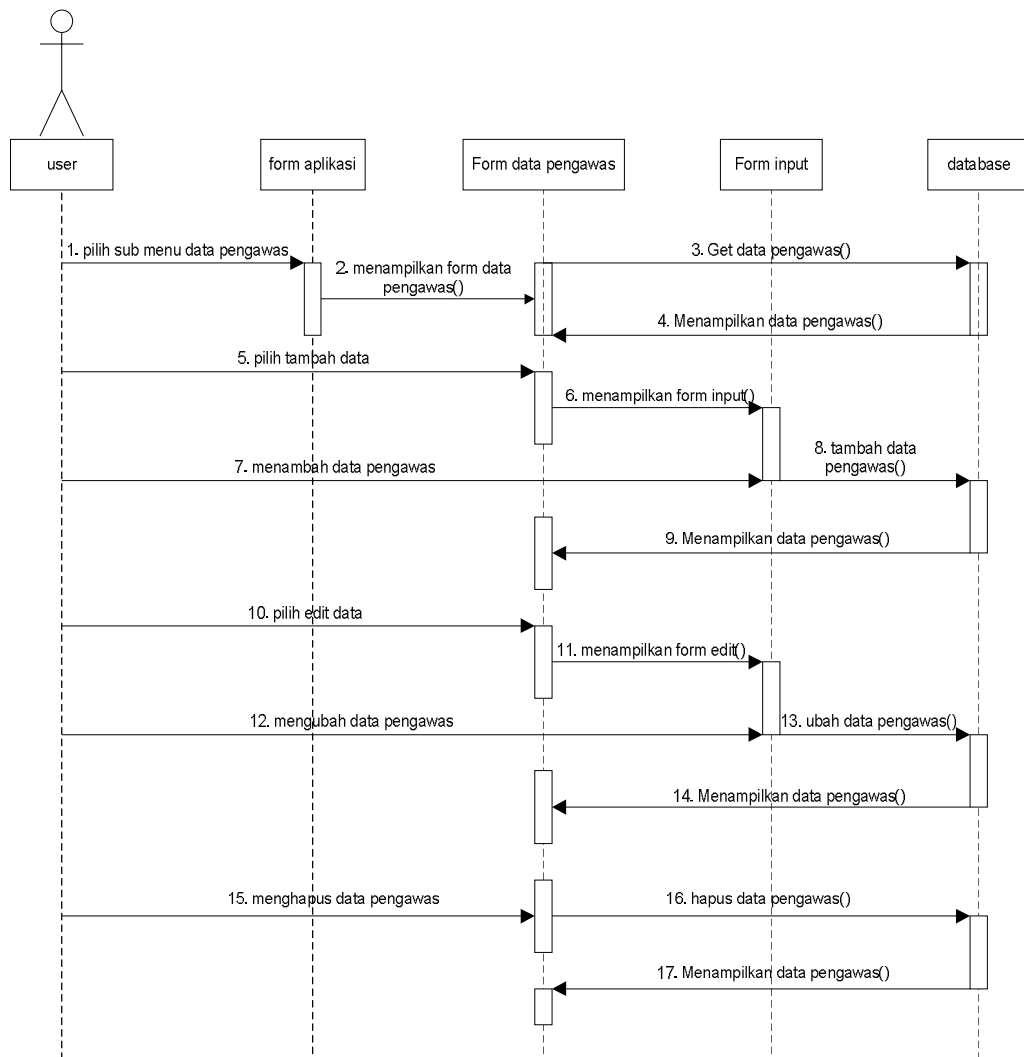
Sequence diagram mengelola data master menjelaskan sebuah *sequence diagram* untuk mengelola data master. Proses ini dilakukan oleh bagian administrasi teknik yang sudah melakukan proses *login* sebelumnya. Adapun *sequence diagram* mengelola data master dapat dilihat pada Gambar V.13 di bawah ini.



Gambar V.13 *Sequence Diagram* Mengelola Data Master
Sumber: Analisis Data (2016)

3. *Sequence Diagram* Mengelola Data Pengawas

Sequence diagram mengelola data pengawas menjelaskan sebuah *sequence diagram* untuk mengelola data pengawas. Proses ini dilakukan oleh bagian administrasi teknik yang sudah melakukan proses *login* sebelumnya. Adapun *sequence diagram* pembuatan pekerjaan baru dapat dilihat pada Gambar V.14 di bawah ini.

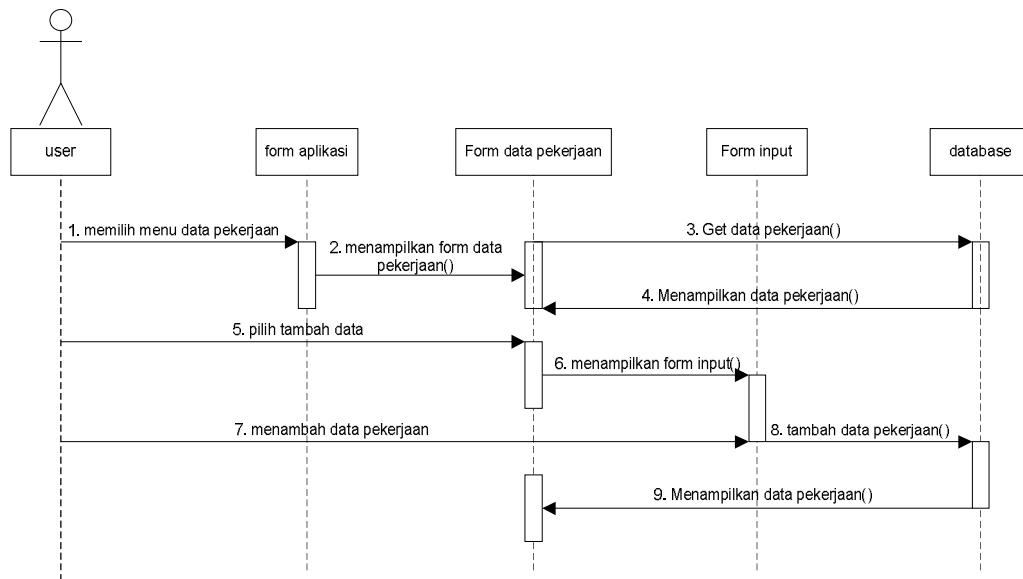


Gambar V.14 *Sequence Diagram* Mengelola Data Pengawas

Sumber: Analisis Data (2016)

4. *Sequence Diagram* Membuat Pekerjaan Baru

Sequence diagram pembuatan pekerjaan baru menjelaskan sebuah *sequence diagram* untuk membuat pekerjaan baru yang akan dilaksanakan baik itu tentang informasi pekerjaan maupun progress pekerjaan. Proses ini dilakukan oleh bagian administrasi teknik yang sudah melakukan proses *login* sebelumnya. Adapun *sequence diagram* pembuatan pekerjaan baru dapat dilihat pada Gambar V.15 di bawah ini.

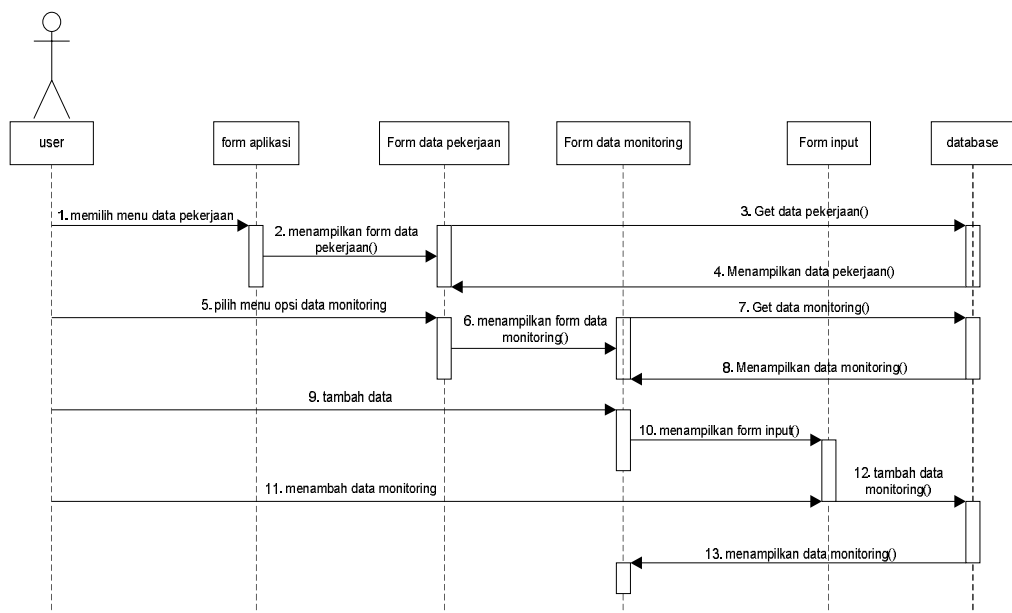


Gambar V.15 *Sequence Diagram* Membuat Pekerjaan Baru

Sumber: Analisis Data (2016)

5. *Sequence Diagram* Input Rencana *Progress* Pekerjaan

Sequence diagram input rencana *progress* pekerjaan merupakan sebuah *sequence diagram* untuk menginput rencana *progress* pekerjaan yang menjadi acuan *progress* kontraktor melaksanakan pekerjaannya. Adapun *sequence diagram* menginput rencana *progress* pekerjaan dapat dilihat pada Gambar V.16 di bawah ini.

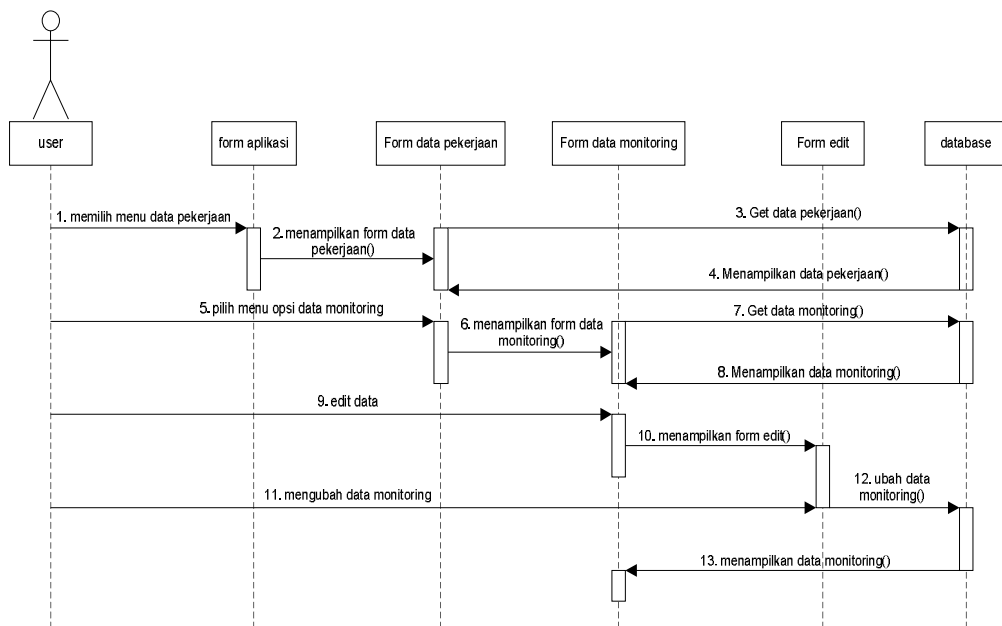


Gambar V.16 *Sequence Diagram* Input Rencana *Progress* Pekerjaan

Sumber: Analisis Data (2016)

6. *Sequence Diagram* Monitoring Progress Pekerjaan

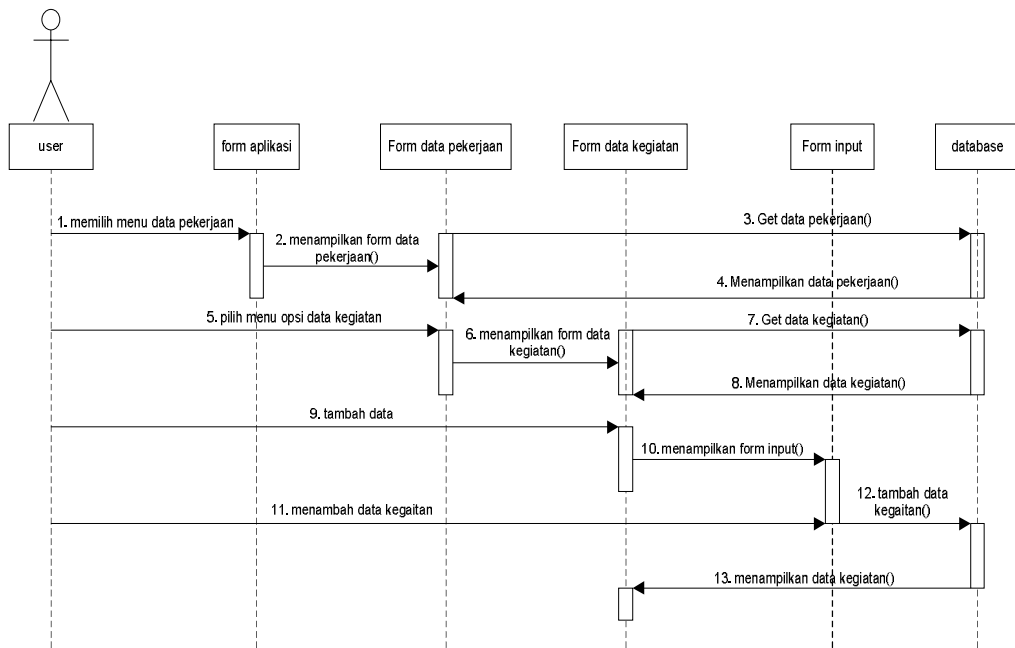
Sequence diagram monitoring pekerjaan merupakan *sequence diagram* yang dilakukan oleh pengawas untuk menginput progress pekerjaan realisasi yang dilakukan oleh kontraktor yang berdasarkan dari laporan mingguan. Adapun *sequence diagram* monitoring pekerjaan dapat dilihat pada gambar V.17 dibawah ini.



Gambar V.17 *Sequence Diagram* Monitoring Progress Pekerjaan
Sumber: Analisis Data (2016)

7. *Sequence Diagram* Menginput Data Kegiatan

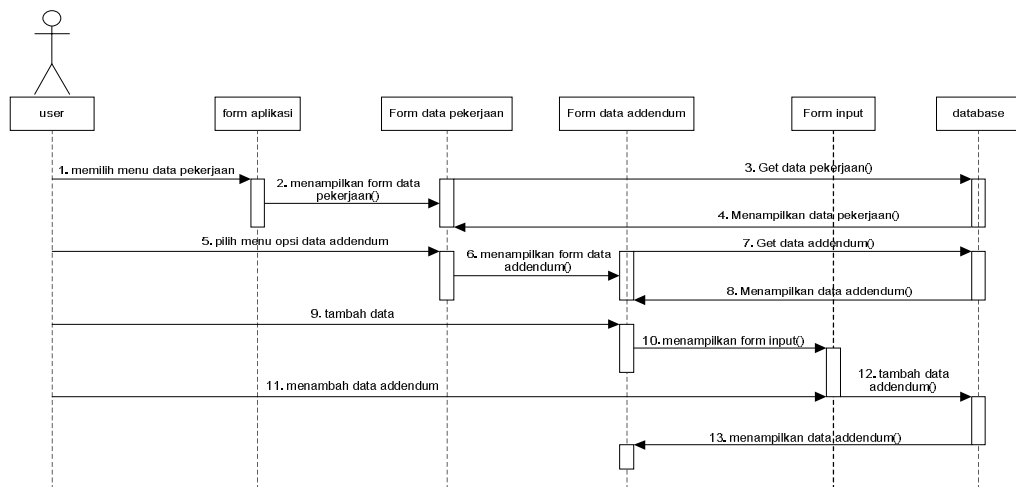
Sequence diagram menginput data kegiatan merupakan *sequence diagram* untuk menginput kegiatan – kegiatan yang ada didalam proyek seperti rapat mingguan ataupun pengawas yang melakukan inspeksi kelapangan. Adapun *sequence diagram* menginput data kegiatan dapat dilihat pada Gambar V.18 dibawah ini.



Gambar V.18 *Sequence Diagram* Menginput Data Kegiatan
 Sumber: Analisis Data (2016)

8. *Sequence Diagram* Membuat Addendum

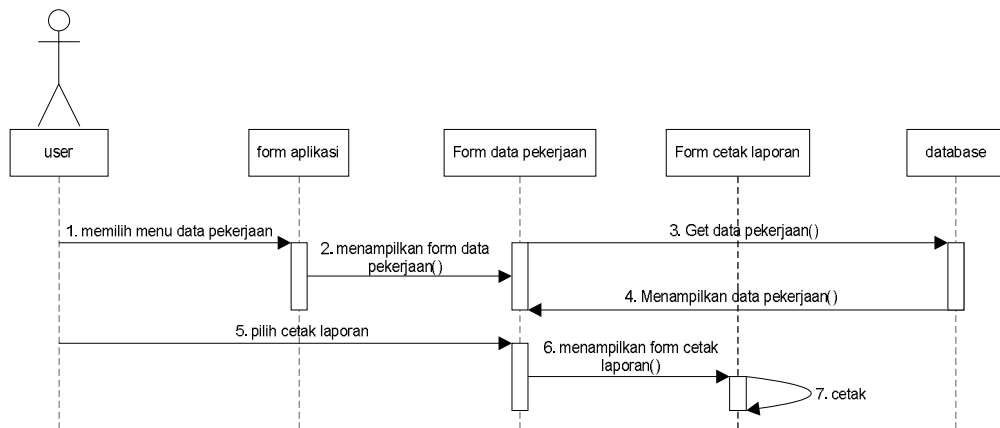
Sequence diagram membuat addendum merupakan *sequence diagram* proses pembuatan addendum, addendum dibuat untuk pekerjaan-pekerjaan yang berjalan tidak sesuai rencana, maka dari itu dibuat addendum. Adapun *sequence diagram* membuat addendum dapat dilihat pada Gambar V.19 dibawah ini.



Gambar V.19 *Sequence Diagram* Membuat Addendum
 Sumber: Analisis Data (2016)

9. *Sequence Diagram* Print Laporan

Sequence diagram print laporan merupakan *sequence diagram* proses mencetak laporan pekerjaan. Adapun *sequence diagram* print laporan dapat dilihat pada Gambar V.20 dibawah ini.

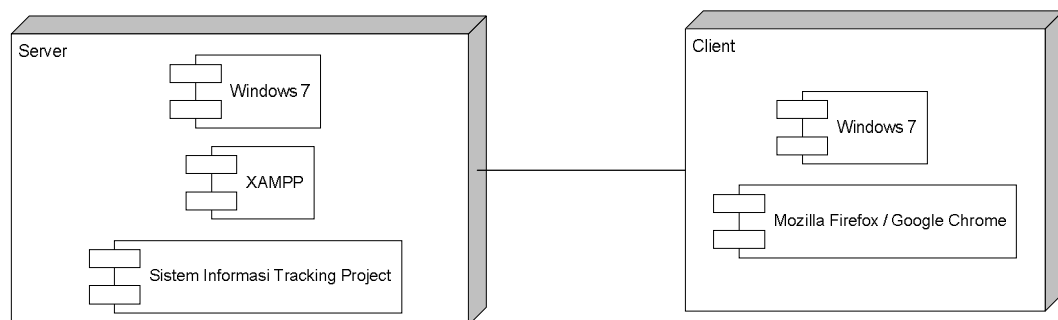


Gambar V.20 *Sequence Diagram* Print Laporan

Sumber: Analisis Data (2016)

5.3.4 *Deployment Diagram*

Deployment diagram digunakan pada bagian-bagian awal proses perancangan system untuk mendokumentasikan arsitektur fisik sebuah system. *Deployment diagram* Sistem Informasi *Project Tracking* yang diusulkan dapat dilihat pada Gambar V.21 dibawah ini.



Gambar V.21 *Deployment Diagram* Sistem Informasi *Project Tracking*

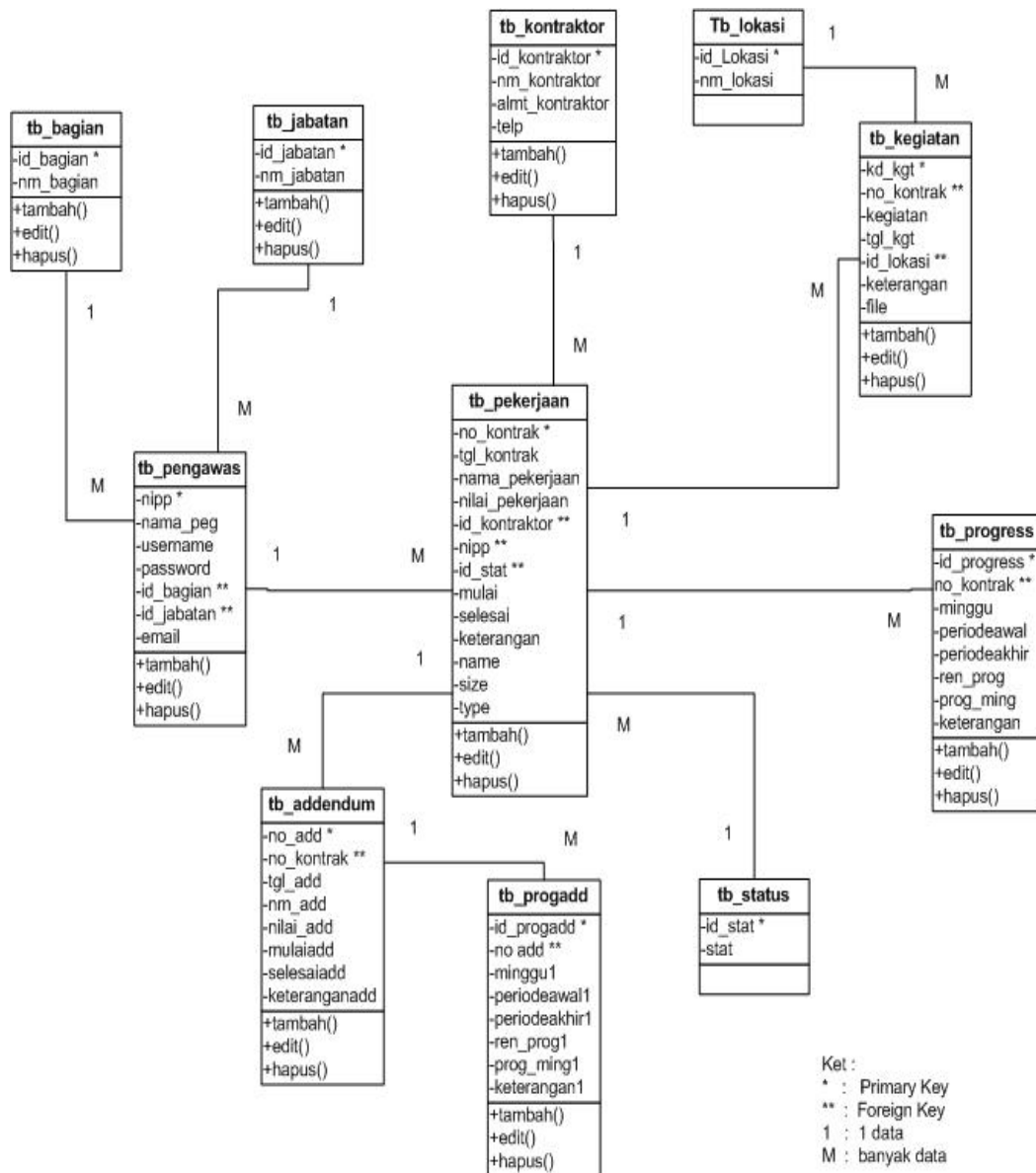
Sumber: Analisis Data (2016)

5.3.5 *Class Diagram*

Class diagram membantu dalam visualisasi struktur kelas-kelas dari suatu sistem dan merupakan tipe diagram yang paling banyak dipakai. *Class diagram*

memperlihatkan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem.

Selama proses analisis, *class diagram* memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem. Selama tahap desain, *class diagram* berperan dalam merangkap struktur dari semua kelas yang membentuk arsitektur yang dibuat. *Class diagram* sistem *project tracking* yang diusulkan dapat dilihat pada Gambar V.22 dibawah ini.



Gambar V.22 Class Diagram Sistem Informasi Project Tracking

Sumber: Analisis Data (2016)

5.4 Perancangan Basis Data

Dalam perancangan basis data pendefinisian struktur data dari file-file *database* akan didokumentasikan dalam kamus data. Kamus data ini diperlukan untuk membuat file secara fisik. Kamus data dapat berfungsi membantu pelaku sistem untuk mengartikan aplikasi secara detail dan mengorganisasi semua elemen data yang digunakan di dalam system secara persis sehingga pemakai dan penganalisis system mempunyai dasar pengertian yang sama tentang masukan, keluaran, penyimpanan dan proses.

1. Spesifikasi Tabel

Nama tabel : tb_pekerjaan
 Fungsi : Untuk menyimpan data pekerjaan
 Tipe : File transaksi

Tabel V.15 Tabel tb_pekerjaan

| No | Nama Elemen | Akronim | Tipe Data | Panjang | Keterangan |
|----|-----------------|-----------------|-----------|---------|--------------------|
| 1 | Nomor kontrak | no_kontrak | Varchar | 30 | <i>Primary Key</i> |
| 2 | Tanggal kontrak | tgl_kontrak | Date | | |
| 3 | Nama pekerjaan | Nama_pekerjaan | Varchar | 100 | |
| 4 | Nilai pekerjaan | Nilai_pekerjaan | Bigint | 20 | |
| 5 | Id kontraktor | Id_kontraktor | Char | 5 | |
| 6 | Nipp | Nipp | Int | 9 | |
| 7 | Mulai | Mulai | Date | | |
| 8 | Selesai | Selesai | Date | | |
| 9 | Keterangan | Keterangan | Varchar | 50 | |
| 9 | Name | Name | Varchar | 100 | |
| 10 | Size | Size | Int | 11 | |
| 11 | Type | Type | Varchar | 100 | |
| 12 | Id status | Id_stat | Int | 1 | |

Sumber: Analisis Data (2016)

2. Spesifikasi Tabel

Nama tabel : tb_kegiatan

Fungsi : Untuk menyimpan data kegiatan pekerjaan

Tipe : File transaksi

Tabel V.16 Tabel tb_kegiatan

| No | Nama Elemen | Akronim | Tipe Data | Panjang | Keterangan |
|----|------------------|----------------|-----------|---------|--------------------|
| 1 | Kode kegiatan | Kd_kgt | Int | 1 | <i>Primary Key</i> |
| 2 | Nomor kontrak | No_kontrak | Varchar | 30 | <i>Foreign Key</i> |
| 3 | Kegiatan | Kegiatan | Varchar | 100 | |
| 4 | Deskripsi | deskripsi | Varchar | 150 | |
| 5 | Tanggal kegiatan | Tgl_kgt | Date | | |
| 6 | Id lokasi | Id_lokasi | Char | 5 | <i>Foreign Key</i> |
| 7 | Keterangan | Keterangan_kgt | Varchar | 75 | |
| 8 | File | File | Text | | |

Sumber: Analisis Data (2016)

3. Spesifikasi Tabel

Nama tabel : tb_bagian

Fungsi : Untuk menyimpan bagian pengawas

Tipe : File master

Tabel V.17 Tabel tb_bagian

| No | Nama Elemen | Akronim | Tipe Data | Panjang | Keterangan |
|----|-------------|-----------|-----------|---------|--------------------|
| 1 | Id bagian | Id_bagian | Char | 5 | <i>Primary Key</i> |
| 2 | Nama bagian | Nm_bagian | Varchar | 20 | |

Sumber: Analisis Data (2016)

4. Spesifikasi Tabel

Nama tabel : tb_jabatan

Fungsi : Untuk menyimpan jabatan pengawas

Tipe : File master

Tabel V.18 Tabel tb_jabatan

| No | Nama Elemen | Akronim | Tipe Data | Panjang | Keterangan |
|----|--------------|------------|-----------|---------|--------------------|
| 1 | Id jabatan | Id_jabatan | Char | 5 | <i>Primary Key</i> |
| 2 | Nama jabatan | Nm_jabatan | Varchar | 30 | |

Sumber: Analisis Data (2016)

5. Spesifikasi Tabel

Nama tabel : tb_pengawas

Fungsi : Untuk menyimpan data pengawas

Tipe : File master

Tabel V.19 Tabel tb_pengawas

| No | Nama Elemen | Akronim | Tipe Data | Panjang | Keterangan |
|----|--------------|------------|-----------|---------|--------------------|
| 1 | Nipp | Nipp | Int | 9 | <i>Primary Key</i> |
| 2 | Nama pegawai | Nama_peg | Varchar | 30 | |
| 3 | Username | Username | Char | 15 | |
| 4 | Password | Password | Char | 15 | |
| 5 | Id bagian | Id_bagian | Char | 5 | <i>Foreign Key</i> |
| 6 | Id jabatan | Id_jabatan | Char | 5 | <i>Foreign Key</i> |
| 7 | email | email | Varchar | 50 | |

Sumber: Analisis Data (2016)

6. Spesifikasi Tabel tb_progress

Nama tabel : tb_progress

Fungsi : Untuk menyimpan data progress pekerjaan

Tipe : File transaksi

Tabel V.20 Tabel tb_progress

| No | Nama Elemen | Akronim | Tipe Data | Panjang | Keterangan |
|----|---------------|--------------|-----------|---------|--------------------|
| 1 | Id progress | Id_prog | Int | 11 | <i>Primary Key</i> |
| 2 | Nomor kontrak | No_kontrak | Varchar | 30 | <i>Foreign Key</i> |
| 3 | Minggu | Minggu | Int | 3 | |
| 4 | Periode | Periodeawal | Date | | |
| 5 | Periode | Periodeakhir | Date | | |

Tabel V.20 Tabel tb_progress (Lanjutan)

| No | Nama Elemen | Akronim | Tipe Data | Panjang | Keterangan |
|----|-------------------|---------------------|-----------|---------|------------|
| 6 | Rencana Progress | Ren_prog | Numeric | 6,2 | |
| 7 | Progress mingguan | Prog_ming | Numeric | 6,2 | |
| 8 | Keterangan | Keterangan_ ming | Varchar | 255 | |

Sumber: Analisis Data (2016)

7. Spesifikasi Tabel tb_addendum

Nama tabel : tb_addendum

Fungsi : Untuk menyimpan data addendum pekerjaan

Tipe : File transaksi

Tabel V.21 Tabel tb_addendum

| No | Nama Elemen | Akronim | Tipe Data | Panjang | Keterangan |
|----|------------------|----------------|-----------|---------|--------------------|
| 1 | Nomor addendum | No_add | Varchar | 30 | <i>Primary Key</i> |
| 2 | Nomor kontrak | No_kontrak | Varchar | 30 | <i>Foreign Key</i> |
| 3 | Tanggal addendum | Tgl_add | Date | | |
| 4 | Nama addendum | Nm_add | Varchar | 15 | |
| 5 | Nilai addendum | Nilai_add | Bigint | 20 | |
| 6 | Mulai | Mulai_add | Date | | |
| 7 | Selesai | Selesai_add | Date | | |
| 8 | Keterangan | Keterangan_add | Varchar | 30 | |

Sumber: Analisis Data (2016)

8. Spesifikasi Tabel tb_progadd

Nama tabel : tb_progadd

Fungsi : Untuk menyimpan progress dari pekerjaan yang memiliki addendum

Tipe : File transaksi

Tabel V.22 Tabel tb_progadd

| No | Nama Elemen | Akronim | Tipe Data | Panjang | Keterangan |
|----|----------------|---------|-----------|---------|--------------------|
| 1 | Id progadd | Id_add | Int | 11 | <i>Primary Key</i> |
| 2 | Nomor addendum | No_add | Varchar | 30 | <i>Foreign Key</i> |

Tabel V.22 Tabel tb_progadd (Lanjutan)

| No | Nama Elemen | Akronim | Tipe Data | Panjang | Keterangan |
|----|-------------------|---------------|-----------|---------|------------|
| 3 | minggu | Minggu1 | Int | 3 | |
| 4 | Periode addendum | Periodeawal1 | Date | | |
| 5 | Periode addendum | Periodeakhir1 | Date | | |
| 6 | Rencana progress | Ren_prog1 | Numeric | 6,2 | |
| 7 | Progress mingguan | Prog_ming1 | Numeric | 6,2 | |
| 8 | Keterangan | Keterangan1 | Varchar | 40 | |

Sumber: Analisis Data (2016)

9. Spesifikasi Tabel tb_kontraktor

Nama tabel : tb_kontraktor

Fungsi : Untuk menyimpan data kontraktor

Tipe : File master

Tabel V.23 Tabel tb_kontraktor

| No | Nama Elemen | Akronim | Tipe Data | Panjang | Keterangan |
|----|-------------------|-----------------|-----------|---------|--------------------|
| 1 | Id kontraktor | Id_kontraktor | Char | 5 | <i>Primary Key</i> |
| 2 | Nama kontraktor | Nm_kontraktor | Varchar | 30 | |
| 3 | Alamat kontraktor | Almt_kontraktor | Varchar | 50 | |
| 4 | Telpon kontraktor | telp | int | 15 | |

Sumber: Analisis Data (2016)

10. Spesifikasi Tabel tb_status

Nama tabel : tb_status

Fungsi : Untuk menyimpan status pekerjaan yang masih berjalan dan yang sudah selesai

Tipe : File Master

Tabel V.24 Tabel tb_status

| No | Nama Elemen | Akronim | Tipe Data | Panjang | Keterangan |
|----|-------------|---------|-----------|---------|--------------------|
| 1 | Id status | Id_stat | int | 1 | <i>Primary Key</i> |
| 2 | Nama status | stat | Varchar | 10 | |

Sumber: Analisis Data (2016)

11. Spesifikasi Tabel tb_lokasi

Nama tabel : tb_lokasi
 Fungsi : Untuk menyimpan lokasi kegiatan yang dilakukan selama pekerjaan pengawasan berlangsung
 Tipe : File Master

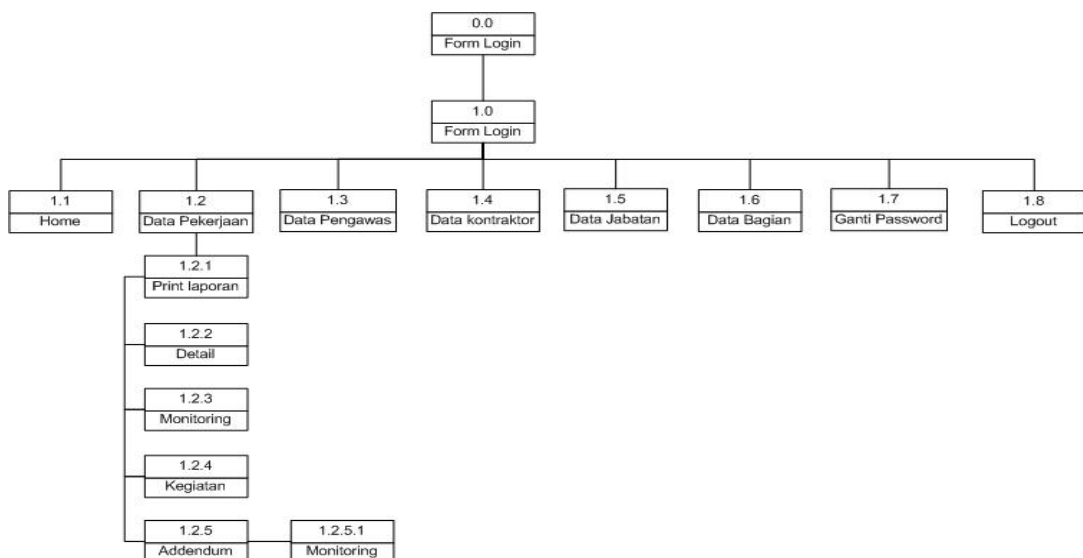
Tabel V.25 Tabel tb_lokasi

| No | Nama Elemen | Akronim | Tipe Data | Panjang | Keterangan |
|----|-------------|-----------|-----------|---------|--------------------|
| 1 | Id lokasi | Id_lok | char | 5 | <i>Primary Key</i> |
| 2 | Nama lokasi | Nm_lokasi | Varchar | 50 | |

Sumber: Analisis Data (2016)

5.5 *Hierarchy plus Input-Process-Output (HIPO) Sistem Informasi Project Tracking*

Hierarchy plus Input-Process-Output (HIPO) digunakan untuk mendokumentasikan sebuah struktur yang menggambarkan hubungan antar fungsi dalam program secara hierarkis. HIPO yang digunakan untuk mendokumentasikan struktur dari program absensi ini yaitu menggunakan diagram HIPO tipe daftar isi visual (DIV). Diagram ini memuat semua modul yang ada dalam sistem beserta nama dan nomornya.

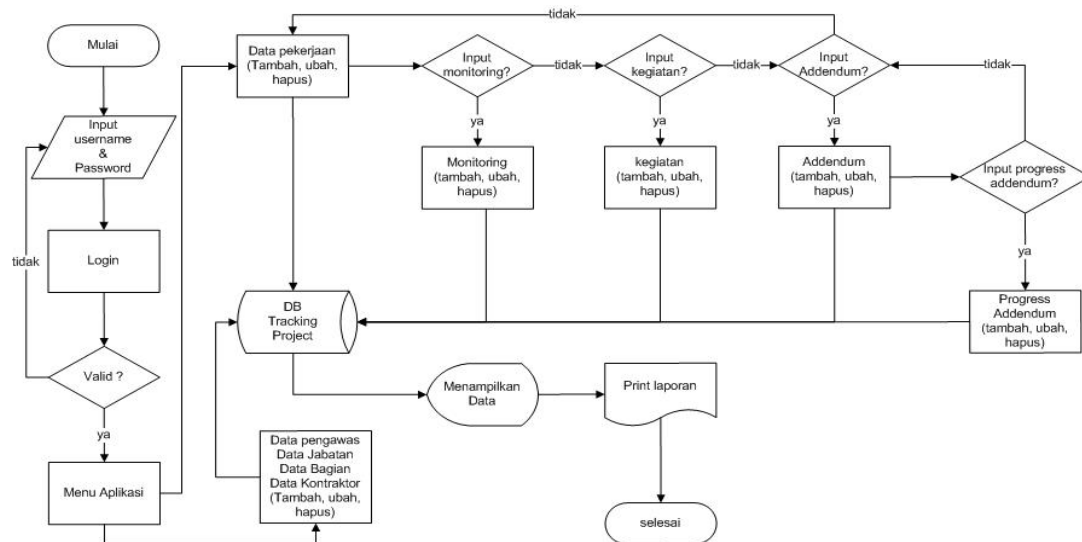


Gambar V.23 HIPO Project Tracking

(Sumber: Analisis Data, 2016)

5.6 Flowchart Aplikasi Sistem Tracking Project

Flowchart yang digunakan untuk mendokumentasikan aplikasi sistem informasi project tracking ini menggunakan bagan alir logika program (*program logic flowchart*). Bagan alir ini digunakan untuk menggambarkan tiap-tiap langkah di dalam program komputer secara logika, bukan untuk menggambarkan instruksi-instruksi program komputer secara terinci. Berikut adalah *program logic flowchart* aplikasi sistem informasi *project tracking*:



Gambar V.24 Flowchart Sistem Project Tracking
(Sumber: Analisis Data, 2016)

Pada Gambar V. 24 terdapat gambaran proses yang terdapat pada sistem informasi tracking project secara keseluruhan. Pada dasarnya terdapat dua hak akses yang dimiliki. Adapun penjelasan terhadap hak akses tersebut yaitu:

Tabel V.26 Pembagian Hak Akses

| No. | Nama User | Hak Akses |
|-----|---------------------|--|
| 1. | Administrasi Teknik | - Dapat memelihara sistem file master yang ada didalam sistem <i>project tracking</i> . - Dapat mengatur hak akses. |
| 2. | Pengawas | - Mengatur dan menjalankan informasi yang ada di dalam menu data pekerjaan. |

(Sumber: Analisis Data, 2016)

5.7 Perancangan Interface Sistem Informasi *Project Tracking*

Rancangan *interface* dari program Sistem Informasi Tracking Project ini adalah sebagai berikut:

1. Rancangan *Form Login*

Rancangan *form login* adalah *form* yang digunakan untuk masuk ke dalam aplikasi. Untuk masuk ke dalam aplikasi, pengguna harus memasukkan username dan password yang benar. Rancangan *form login* dapat dilihat pada Gambar V.25.

Silahkan Login

username

password

Gambar V.25 Rancangan *Form Login*

(Sumber: Analisis Data, 2016)

2. Rancangan Daftar Menu

Rancangan Daftar menu adalah tampilan menu yang masing-masing memiliki fungsi berbeda. Pada daftar menu berisi delapan menu. Delapan menu tersebut adalah Profil Pengguna, Konfigurasi Aplikasi, Master, Transaksi, Stok dan Keluar. Rancangan *form* menu dapat dilihat pada Gambar V.26.

| PROJECT TRACKING |
|------------------|
| HOME |
| Data Pekerjaan |
| Data Pengawas |
| Data Kontraktor |
| Data Jabatan |
| Data Bagian |
| Ganti Password |
| Logout |

Gambar V.26 Rancangan Daftar Menu

(Sumber: Analisis Data, 2016)

3. Rancangan Tampilan Untuk Menampilkan Data Pekerjaan

Rancangan tampilan untuk menampilkan data pekerjaan pada Gambar V.27 adalah tampilan yang digunakan untuk menampilkan pekerjaan – pekerjaan yang berada didalam database

| PROJECT TRACKING | | | | |
|------------------|------------------|----------------|-----------------|----------------------|
| HOME | Data Pekerjaan | | | |
| Data Pekerjaan | tambah pekerjaan | | | |
| Data Pengawas | Search | | | |
| Data Kontraktor | nomor kontrak | nama pekerjaan | nilai pekerjaan | nama kontraktor opsi |
| Data Jabatan | | | | |
| Data Bagian | | | | |
| Ganti Password | | | | |
| Logout | | | | |

Gambar V.27 Rancangan Tampilan Untuk Menampilkan Data Pekerjaan

(Sumber: Analisis Data, 2016)

4. Rancangan *Form* Tambah Data Pekerjaan

Rancangan *form* tambah data pekerjaan pada Gambar V.28 adalah *form* yang digunakan untuk menambah data pekerjaan kedalam database. Berikut penjelasan mengenai penggunaan pada *form* ini:

- a. Tombol Simpan : Untuk menyimpan data pekerjaan.
- b. Tombol Batal : Untuk membatalkan proses penambahan data pekerjaan

Tambah Pekerjaan Baru

Nomor Kontrak

Tanggal Kontrak

Nama Pekerjaan

Nilai Pekerjaan

Pengawas

Nama Kontraktor

Batal Simpan

Gambar V.28 Rancangan *Form* Tambah Data Pekerjaan

(Sumber: Analisis Data, 2016)

5. Rancangan *Form* Edit Data Pekerjaan

Rancangan *form* edit data pekerjaan pada Gambar V.29 adalah *form* yang digunakan untuk mengedit data pekerjaan yang sudah disimpan sebelumnya. Berikut adalah penjelasan mengenai penggunaan *form* edit data pekerjaan adalah sebagai berikut:

- a. Tombol Kembali : Untuk kembali ke list data pekerjaan.
- b. Tombol Simpan : Untuk menyimpan data yang sudah diedit.

Edit Pekerjaan
<-- kembali

| | |
|-------------------|--|
| Nomor Kontrak | |
| Tanggal Kontrak | |
| Nilai Pekerjaan | |
| Nama Pekerjaan | |
| Nama Pengawas | |
| Nama Kontraktor | |
| Mulai Pekerjaan | |
| Selesai Pekerjaan | |
| Foto | |

Gambar V.29 Rancangan *Form* Edit Data Pekerjaan
(Sumber: Analisis Data, 2016)

6. Rancangan Tampilan Monitoring Pekerjaan

Rancangan *form* monitoring pekerjaan pada Gambar V.30 adalah *form* yang digunakan untuk membuat rencana progress pekerjaan dan hasil evaluasi dari laporan progress mingguan yang diserahkan oleh kontraktor.

PROJECT TRACKING

HOME

Data Pekerjaan

Data Pengawas

Data Kontraktor

Data Jabatan

Data Bagian

Ganti Password

Logout

Data Monitoring

| minggu ke- | periode | rencana progress | progress minggu ini | progress kumulatif | deviasi | keterangan | opsi |
|------------|---------|------------------|---------------------|--------------------|---------|------------|------|
| | | | | | | | |

Gambar V.30 Rancangan Tampilan Monitoring Pekerjaan
(Sumber: Analisis Data, 2016)

7. Rancangan Tampilan Data Pengawas

Rancangan tampilan data pengawas pada Gambar V.31 adalah tampilan yang digunakan untuk menampilkan data pengawas yang berada di divisi teknik sipil tanjung priok.

The image shows a web application interface for 'PROJECT TRACKING'. On the left is a sidebar menu with the following items: HOME, Data Pekerjaan, Data Pengawas, Data Kontraktor, Data Jabatan, Data Bagian, Ganti Password, and Logout. The main content area is titled 'Data Pengawas' and contains a 'tambah pengawas' button, a search box labeled 'search', and a table with the following columns: nipp, nama, bagian, jabatan, email, and opsi.

Gambar V.31 Rancangan Tampilan Data Pengawas

(Sumber: Analisis Data, 2016)

8. Rancangan Tampilan Rekap Laporan

Rancangan tampilan rekap laporan pada Gambar V.32 adalah tampilan yang digunakan untuk menampilkan rekap laporan *progress* seluruh pekerjaan yang berada di divisi teknik sipil tanjung priok.

DATA PEKERJAAN

| No Kontrak | Tgl Kontrak | Nama Pekerjaan | Nilai Pekerjaan | Nama Kontraktor | No Addendum | Nama Addendum | Tgl Addendum | Nilai Addendum | Progress s.d minggu lalu | Progress s.d minggu ini | Keterangan |
|------------|-------------|----------------|-----------------|-----------------|-------------|---------------|--------------|----------------|--------------------------|-------------------------|------------|
| | | | | | | | | | | | |

Gambar V.32 Rancangan Rekap Laporan

(Sumber: Analisis Data, 2016)

5.8 Implementasi Sistem

Untuk dapat menggunakan aplikasi, tentunya diperlukan suatu spesifikasi perangkat lunak ataupun perangkat keras yang mendukung agar aplikasi dapat berjalan dengan baik.

5.8.1 Kebutuhan Perangkat Lunak

Untuk menggunakan aplikasi sistem informasi *Project Tracking* ini diperlukan suatu minimum kebutuhan perangkat lunak sebagai berikut:

- a. XAMPP 3.2.1 untuk pengelolaan *database*.
- b. *Browser* Mozilla Firefox atau Google Chrome untuk menjalankan fitur phpMyadmin XAMPP.
- c. Sistem Operasi minimal Windows XP.

5.8.2 Kebutuhan Perangkat Keras

Untuk mendukung kebutuhan implementasi sistem informasi Tracking Project pada PT Pelabuhan Tanjung Priok diperlukan dukungan sistem komputer yang memadai dalam segi *hardware*, diantaranya sebagai berikut:

- a. *Processor* : minimal Intel Atom (2,6 GHz).
- b. RAM : minimal 1 GB.
- c. *Harddisk* : minimal 250 GB.

BAB VI

KESIMPULAN DAN SARAN

6.1 Kesimpulan

Berdasarkan pembuatan aplikasi yang dilakukan mengenai sistem informasi *project tracking* di PT Pelabuhan Tanjung Priok, dapat disimpulkan beberapa hal yaitu:

1. Dengan adanya aplikasi ini dapat membantu pengawas dalam mengontrol laporan mingguan dan mempermudah pengawas dalam membuat addendum pekerjaan.
2. Memudahkan pengawas dalam memantau seluruh *progress* pekerjaan untuk laporan kepada atasan langsung (*manager*).
3. Dengan aplikasi ini dapat membantu pengawas dalam menginformasikan nilai kontrak suatu pekerjaan bila ada kontraktor yang ingin melakukan penagihan.

6.2 Saran

Adapun saran yang diharapkan agar aplikasi manajemen aset ini dapat berjalan dengan baik yaitu:

1. Untuk penerapan sistem baru terhadap sistem lama, sebaiknya dilakukan secara bertahap dan diperlukan sosialisasi kepada bagian-bagian yang terkait yang akan menggunakan sistem baru ini.
2. Sebaiknya dilakukan pemeliharaan aplikasi secara berkala, sehingga aplikasi dapat berjalan dengan baik.
3. Tersedianya sumber daya manusia yang berkompeten dalam bidang IT untuk dapat memperbaiki jika terjadi kesalahan atau *error* pada aplikasi.
4. Untuk penelitian selanjutnya, penulis menyarankan agar aplikasi ini dapat menyediakan informasi terhadap harga – harga satuan tiap material pekerjaan dan bobot pekerjaannya.

DAFTAR PUSTAKA

A.S Rosa dan Salahuddin M, 2011. *Modul Pembelajaran Rekayasa Perangkat Lunak* (Terstruktur dan Berorientasi Objek), Modula, Bandung.

Aryanto, 2001. *Pengolahan Database dengan Microsoft Visual Foxpro 6.0*, Penerbit PT. elex Media Komputindo, Jakarta.

Basuki, 2015. *Kolaborasi Codeigniter & Bootstrap Membangun Aplikasi PSB Sekolah*. Jakarta: CV. Lokomedia.

Bodnar, George H., William S. Hopwood, 2000. *Sistem Informasi Akuntansi*, Edisi keenam, Terjemahan Amir Abadi Jusuf, Rudi M. Tambunan, Salemba Empat, Buku Satu, Jakarta.

Davis, Gordon B. 1991. *Sistem Informasi Manajemen*. Jakarta: PT. Pustaka Binaman Pressindo.

Fathansyah.1999. *Basis Data*. Bandung: Informatika Bandung.

F Nash, John, diterjemahkan oleh La Midjan. 2003. "*Sistem Informasi Akuntansi I Pendekatan Manual Pratika Penyusunan Metode dan Prosedur*". Bandung: Lembaga Informatika Akuntansi.

Gelinas, Oram, dan Wiggins 1990. *Analisa dan Perancangan Sistem Pengisian Formulir Rencana Studi Online*. Jakarta: STMIK Perbanas.

Gelinas, Ulric, Allan E. Oram, William P. Wiggins. 1990. *Accounting Information System*. Boston: Kent.

Handoko. 1995. *Dasar-Dasar Manajemen Produksi dan Operasi*. BPFE-UGM

Indrajit. 2001. *Analisis dan Perancangan Sistem Berorientasi Objek*. Bandung: Informatika.

Jogiyanto, HM. 2005. *Analisis dan Desain Sistem Informasi: Pendekatan Terstruktur Teori dan Praktek Aplikasi Bisnis*. Edisi Ketiga. Yogyakarta: Andi Offset.

- Kadir Abdul. 2002. *Pengenalan Sistem Informasi*, Andi Yogyakarta.
- McFadden, dkk. 1999. *Konsep dan Tuntunan Praktis Basis Data*, Andi Yogyakarta, Yogyakarta.
- McLeod. 2004. *Sistem Informasi Manajemen*. PT. Indeks. Jakarta.
- Munawar. 2005. *Pemodelan Visual dengan UML*, Graha Ilmu, Yogyakarta.
- Murdick, dkk. 1991. *Sistem Informasi Untuk Manajemen Modern*. Jakarta: Erlangga.
- Raharjo, 2015. *Belajar Otodidak Framework Codeigniter*. Jakarta: Informatika.
- Ramakrishnan, R. dan Gehrke, J. 2003. *Sistem Manajemen Database*. Edisi ke-3. Terjemahan Tim Penerjemah ANDI. Yogyakarta: Andi dan McGraw-Hill Education.
- Sarwoto, 1983, *Dasar-dasar Organisasi dan Manajemen*, Jakarta: Cetakan keempat, Ghalia.
- Septian, Gungun. 2011. *Trik Pintar Menguasai Codeigniter*. Jakarta: PT Elex Media Komputindo.
- Siagian, 1970. *Filsafat Administrasi*. Jakarta: Gunung Agung.
- Silberschatz, A. dkk. 2002. *Database System Concepts, 4th Edition*, New York: McGraw-Hill.
- Soekarno K, 1968. *Dasar-Dasar Manajemen*. Jakarta
- Stephens and Ronald Plew. 2000 *Database Design*. 1st ed. Sams Publishing.
- Sutabri Tata, 2005. *Sistem Informasi Manajemen*, ANDI Yogyakarta, Yogyakarta.
- Sidharta, Lani. 1995. *Pengantar Sistem Informasi Bisnis*. Jakarta: Elex Media Komputindo.

Wilkinson. 1992. *Konsep dan Tuntunan Praktis Basis Data*, Andi Yoyakarta, Yogyakarta.

_____, 2012, *Tipe Data Mysql* <http://www.nulis-ilmu.com/2014/11/mengenal-tipe-data-pada-database-mysql.html> , (Tanggal Akses: 2 Maret 2016)

_____, 2012, *Manajemen Proyek* <https://asaladablog.wordpress.com/>, (Tanggal Akses: 2 Maret 2016)

_____, 2013, *Monitoring*,
<http://nuwrieardkhiyari.blogdetik.com/2013/12/01/monitoring>, (Tanggal Akses: 2 Maret 2016)

_____, 2015, *Basis Data Menurut Para Ahli*
<http://ariplie.blogspot.co.id/2015/04/pengertian-basis-data-menurut-para-ahli.html>, (Tanggal Akses: 28 Maret 2016)

_____, 2013, *Pengertian PHP dan MySQL*
<http://jordansyahreza.blogspot.com/p/pengertian-php-dan-my-sql.html>
(Tanggal Akses: 28 Maret 2016)

_____, 2013, *Kelebihan dan Kekurangan PHP*
<http://www.nurulimam.com/2013/09/kelebihan-kekurangan-php.html>
(Tanggal Akses: 28 Maret 2016)

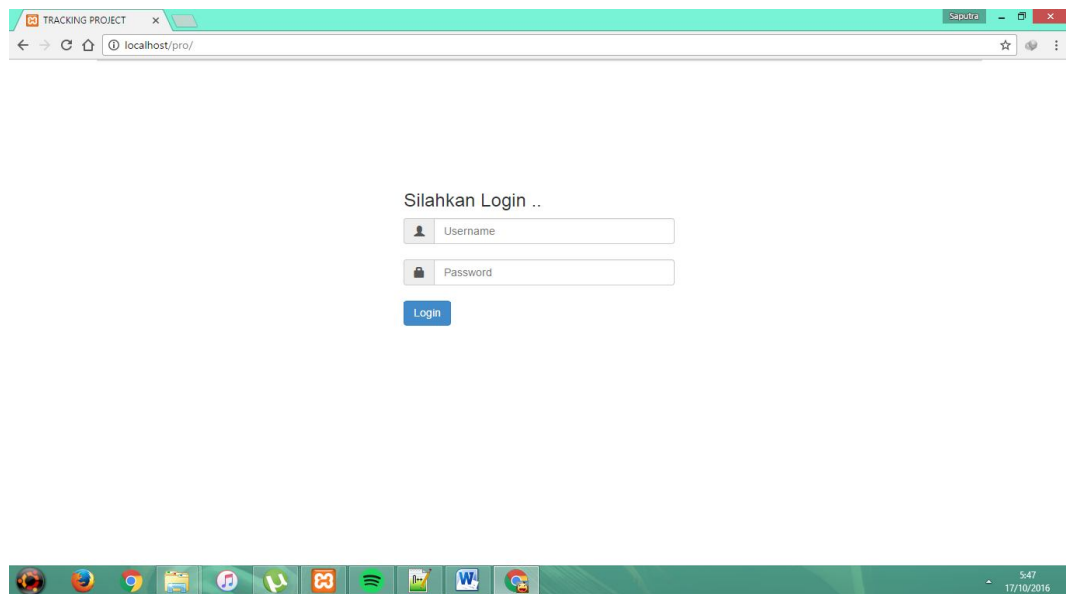
_____, 2013, *Kelebihan dan Kekurangan PHP*
<http://www.jejaring.web.id/kekurangan-dan-kelebihan-php/>
(Tanggal Akses: 28 Maret 2016)

_____, 2016, *UML Diagram* <http://www.uml-diagrams.org/>
(Tanggal Akses: 28 Maret 2016)

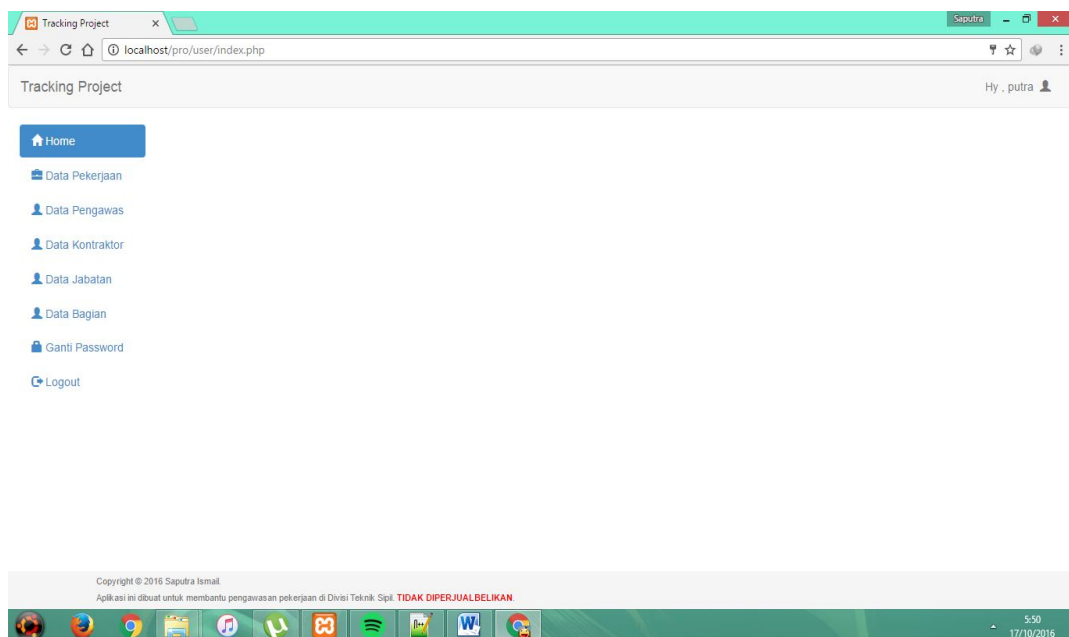
LAMPIRAN A

TAMPILAN PROGRAM

1. Login



2. Menu Utama



3. Tampilan Data Pekerjaan

Tracking Project

Hy, putra

Home | Data Pekerjaan | + Tambah Pekerjaan

Cari Pekerjaan ...

| No | No Kontrak | Nama Pekerjaan | Nilai Pekerjaan | Nama Kontraktor | Status | Opsi |
|----|-----------------------|--|----------------------|---------------------------|------------|------------------------|
| 1 | HK.556/31/7/2/PTP-15 | Peningkatan dan Pendalaman Dermaga 212 dan 213 | Rp.100.621.000.000,- | PT Nindya Karya (Persero) | On Progres | Menu Opsi + Edit Hapus |
| 2 | HK.566/3/4/1/C.TPK-14 | Perkuatan dan Pendalaman dermaga 209L, 210-211 berikut pemasangan rel | Rp.122.151.599.000,- | PT Wijaya Karya (Persero) | On Progres | Menu Opsi + Edit Hapus |
| 3 | HK.566/29/1/1/PTP-16 | Pengadaan dan Pemasangan Panel Distribusi dan Kabel Feeder untuk alat B/M Dermaga dan Lapangan 303-305 | Rp.1.023.617.000,- | PT Todo Mitra Utama | Selesai | Menu Opsi + Edit Hapus |

1 2

Copyright © 2016 Saputra Ismail
Aplikasi ini dibuat untuk membantu pengawasan pekerjaan di Divisi Teknik Sipil. **TIDAK DIPERJUALBELIKAN**

5:52 17/10/2016

4. Data Pengawas

Tracking Project

Hy, putra

Home | Data Pengawas | + Tambah Pengawas

Cari Pengawas ...

| No | NIPP | Nama Pengawas | Bagian | Jabatan | E-mail | Opsi |
|----|-----------|-----------------------|-------------------|---------------------|---|------------|
| 1 | 268056615 | Meinawati Kusumaria | Estimasi & Desain | Administrasi Teknik | meinawati.kusumaria@indonesiaport.co.id | Edit Hapus |
| 2 | 284107350 | Eko Sugito | Penilikan | Manager | eko.sugito@indonesiaport.co.id | Edit Hapus |
| 3 | 285067285 | M. Yudhatin Satriandi | Penilikan | Pengawas | muhammad.yudhatin@indonesiaport.co.id | Edit Hapus |
| 4 | 286097457 | Agung Yudhantara | Penilikan | Pengawas | agung.yudhantara@indonesiaport.co.id | Edit Hapus |
| 5 | 287057688 | Andrianto Rahmadha | Penilikan | Pengawas | andrianto.rahmadha@indonesiaport.co.id | Edit Hapus |

1 2 3

Copyright © 2016 Saputra Ismail
Aplikasi ini dibuat untuk membantu pengawasan pekerjaan di Divisi Teknik Sipil. **TIDAK DIPERJUALBELIKAN**

22:54 17/11/2016

5. Data Kontraktor

Tracking Project

Hy, putra

Home

Data Kontraktor

Data Pekerjaan + Tambah Kontraktor

| No | ID | Nama | Alamat | Telepon | Opsi |
|----|-------|---------------------------|---------------------------|----------|------------|
| 1 | KT001 | PT Wijaya Karya (Persero) | Jl. Pattimura No 15 | 4302354 | Edit Hapus |
| 2 | KT002 | PT Hutama Karya (Persero) | Jl. Gajah Mada Utara No.3 | 4451276 | Edit Hapus |
| 3 | KT003 | PT Nindya Karya (Persero) | jalan raya pennas | 44410829 | Edit Hapus |
| 4 | KT004 | PT Leo Tunggal Mandiri | | 0 | Edit Hapus |
| 5 | KT005 | CV Tri Indah Mulya | | 0 | Edit Hapus |

1 2

Copyright © 2016 Saputra Ismail.
Aplikasi ini dibuat untuk membantu pengawas an pekerjaan di Divisi Teknik Sipil. **TIDAK DIPERJUALBELIKAN**

22:55
17/11/2016

6. Data Jabatan

Tracking Project

Hy, putra

Home

Data Jabatan

Data Pekerjaan + Tambah Jabatan

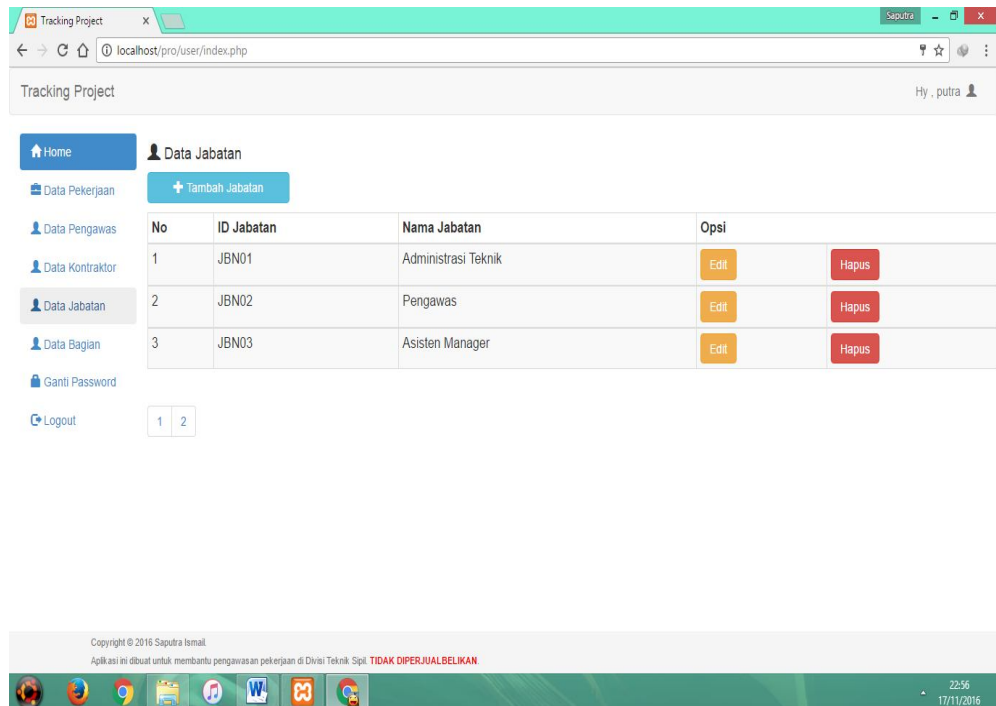
| No | ID Jabatan | Nama Jabatan | Opsi |
|----|------------|---------------------|------------|
| 1 | JBN01 | Administrasi Teknik | Edit Hapus |
| 2 | JBN02 | Pengawas | Edit Hapus |
| 3 | JBN03 | Asisten Manager | Edit Hapus |

1 2

Copyright © 2016 Saputra Ismail.
Aplikasi ini dibuat untuk membantu pengawas an pekerjaan di Divisi Teknik Sipil. **TIDAK DIPERJUALBELIKAN**

22:56
17/11/2016

7. Data Bagian



Tracking Project

Hy, putra

Home

Data Jabatan

Data Pekerjaan + Tambah Jabatan

| No | ID Jabatan | Nama Jabatan | Opsi |
|----|------------|---------------------|------------|
| 1 | JBN01 | Administrasi Teknik | Edit Hapus |
| 2 | JBN02 | Pengawas | Edit Hapus |
| 3 | JBN03 | Asisten Manager | Edit Hapus |

Data Pengawas

Data Kontraktor

Data Jabatan

Data Bagian

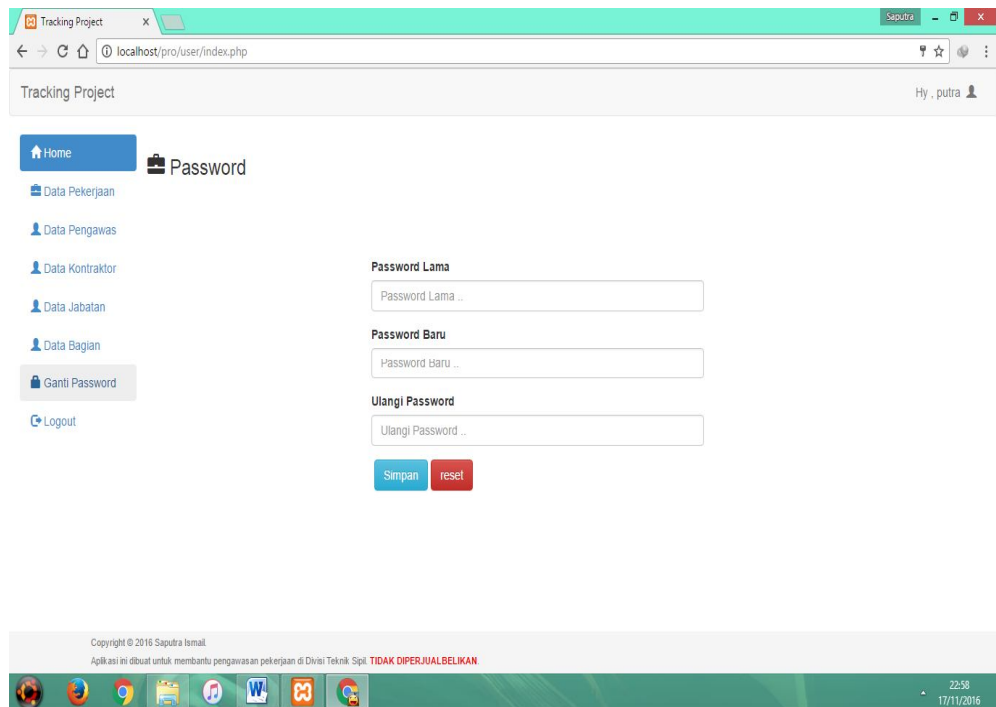
Ganti Password

Logout 1 2

Copyright © 2016 Saputra Ismail
Aplikasi ini dibuat untuk membantu pengawasan pekerjaan di Divisi Teknik Sipil. **TIDAK DIPERJUALBELIKAN**

22:56
17/11/2016

8. Ganti Password



Tracking Project

Hy, putra

Home

Password

Data Pekerjaan

Data Pengawas

Data Kontraktor

Data Jabatan

Data Bagian

Ganti Password

Logout

Password Lama

Password Lama

Password Baru

Password Baru

Ulangi Password

Ulangi Password

Simpan reset

Copyright © 2016 Saputra Ismail
Aplikasi ini dibuat untuk membantu pengawasan pekerjaan di Divisi Teknik Sipil. **TIDAK DIPERJUALBELIKAN**

22:58
17/11/2016

LAMPIRAN B

CODING PROGRAM

1. Login

```
<!DOCTYPE html>
<html>
<head>

    <title>PROJECT TRACKING</title>
    <link rel="stylesheet" type="text/css" href="assets/css/bootstrap.css">
    <link rel="stylesheet" type="text/css" href="assets/js/jquery-ui/jquery-
ui.css">
    <script type="text/javascript" src="assets/js/jquery.js"></script>
    <script type="text/javascript" src="assets/js/bootstrap.js"></script>
    <script type="text/javascript" src="assets/js/jquery-ui/jquery-
ui.js"></script>
    <?php include 'user/config.php'; ?>
    <style type="text/css">
        .kotak{
            margin-top: 150px;
        }

        .kotak .input-group{
            margin-bottom: 20px;
        }
    </style>
</head>
<body>
    <div class="container">
```

```

<?php
if(isset($_GET['pesan'])){
    if($_GET['pesan'] == "gagal"){
        echo "<div style='margin-bottom:-55px' class='alert
alert-danger' role='alert'><span class='glyphicon glyphicon-warning-
sign'></span> Login Gagal !! Username dan Password Salah !!</div>";
    }
}
?>
<div class="panel panel-default">
    <form action="login_act.php" method="post">
        <div class="col-md-4 col-md-offset-4 kotak">
            <h3>Silahkan Login ..</h3>
            <div class="input-group">
                <span class="input-group-
addon"><span class="glyphicon glyphicon-user"></span></span>
                <input type="text" class="form-
control" placeholder="Username" name="username">
            </div>
            <div class="input-group">
                <span class="input-group-
addon"><span class="glyphicon glyphicon-lock"></span></span>
                <input type="password"
class="form-control" placeholder="Password" name="password">
            </div>
            <div class="input-group">
                <input type="submit" class="btn btn-
primary" value="Login">
            </div>
        </div>
    </form>

```

```
        </div>
    </div>
</body>
</html>
```

2. Data Pekerjaan

```
<?php session_start(); ?>
<HTML>
<HEAD>
<?php

    include 'cek.php';
    include 'config.php';
    $use=$_SESSION['username'];
    //echo $use
    $sql="select * from tb_pengawas where username='$use'";
    $edit=mysql_query($sql);
    //echo $sql;
    $d=mysql_fetch_array($edit);
    //echo $d[5];
    $jabatan=$d[5];
    ?>
<title>Tracking</title>
<link rel="stylesheet" type="text/css" href="../assets/css/bootstrap.css">
<link rel="stylesheet" type="text/css" href="../assets/js/jquery-ui/jquery-
ui.css">
<script type="text/javascript" src="../assets/js/jquery.js"></script>
<script type="text/javascript" src="../assets/js/bootstrap.js"></script>
<script type="text/javascript" src="../assets/js/jquery-ui/jquery-
ui.js"></script>
<script type="text/javascript">
```

```

$(document).ready(function () {
    $('#tanggal').datepicker({
        format: "dd-mm-yyyy",
        autoclose:true
    });
});
</script>

```

</HEAD>

<h4> Data
Pekerjaan</h4>

<?php

```

if(isset($_GET['pesan'])){

```

```

    $pesan=mysql_real_escape_string($_GET['pesan']);

```

```

    if($pesan=="gagal"){

```

```

        echo "<div class='alert alert-danger'>Data Pekerjaan Tidak  
Berhasil Disimpan !!</div>";

```

```

    }else if($pesan=="tdksama"){

```

```

        echo "<div class='alert alert-warning'>Password yang anda  
masukkan tidak sesuai !! silahkan ulangi !! </div>";

```

```

    }else if($pesan=="oke"){

```

```

        echo "<div class='alert alert-success'>Data Pekerjaan Berhasil  
Disimpan </div>";

```

```

    }

```

```

}

```

?>

```

<button style="margin-bottom:10px" data-toggle="modal" data-  
target="#myModal" class="btn btn-info col-md-2"><span class="glyphicon  
glyphicon-plus"></span> Tambah Pekerjaan</button>

```

```

<form action="cari_act.php" method="get">

```

```

<div class="input-group col-md-5 col-md-offset-7">
    <span class="input-group-addon" id="basic-addon1"><span
class="glyphicon glyphicon-search"></span></span>
    <input type="text" class="form-control" placeholder="Cari
Pekerjaan .." aria-describedby="basic-addon1" name="cari">
</div>
</form>
<form action="act_laporan.php" method="post">
    <div class="col-md-12">
        <!-- <a style="margin-bottom:10px" href="act_laporan.php"
target="_blank" class="btn btn-default pull-right"><span class='glyphicon
glyphicon-print'></span>Cetak</a> -->
        <button name="btnCetak" type="submit" formtarget="_blank"
class="btn btn-default pull-left"><span class='glyphicon glyphicon-
print'></span> Cetak</button>
    </div>
    <div class="col-md-12">
        <input type="radio" name="radFilter" value="0">Semua
        <input type="radio" name="radFilter" value="1">On Progress
        <input type="radio" name="radFilter" value="2">Selesai
    </div>
</form>

<?php

$per_hal=10;
$jumlah_record=mysql_query("SELECT COUNT(*) from tb_pekerjaan");
$jum=mysql_result($jumlah_record, 0);
$halaman=ceil($jum / $per_hal);
$page = (isset($_GET['page'])) ? (int)$_GET['page'] : 1;
$start = ($page - 1) * $per_hal;

```

```

?>
<td colspan=2>

</td>
<br/>
<table width=2000 border=0>
<div class="container">
  <table class="table table-striped table-bordered table-condensed">
    <thead>
      <tr>
        <th>No</th>
        <th>No Kontrak</th>
        <th>Nama Pekerjaan</th>
        <th>Nilai Pekerjaan</th>
        <th>Nama Kontraktor</th>
        <th>Keterangan</th>
        <th>Status</th>
        <!-- <td class="col-md-1">Sisa</td>      -->
        <th width=150 colspan=3>Opsi</th>
      </tr>
    </thead>
    <?php
    if(isset($_GET['cari'])){
      $cari=mysql_real_escape_string($_GET['cari']);
      $search=$_GET['cari'];
      $pkry=mysql_query("select * from tb_pekerjaan, tb_kontraktor,
tb_pengawas, tb_status where tb_pengawas.nipp = tb_pekerjaan.nipp and
tb_kontraktor.id_kontraktor=tb_pekerjaan.id_kontraktor and tb_status.id_stat
= tb_pekerjaan.id_stat and tb_pekerjaan.nama_pekerjaan like '%$search%'");
    }else{

```

```

        $pkrij=mysql_query("select * from tb_pekerjaan, tb_kontraktor,
tb_pengawas, tb_status where tb_pengawas.nipp = tb_pekerjaan.nipp and
tb_kontraktor.id_kontraktor=tb_pekerjaan.id_kontraktor and tb_status.id_stat
= tb_pekerjaan.id_stat limit $start, $per_hal");
    }
    $no=1;
    while($b=mysql_fetch_array($pkrij)){

?>
<tbody>
    <tr>
        <td><?php echo $no++ ?></td>
        <td><?php echo $b['no_kontrak'] ?></td>
        <td><?php echo $b['nama_pekerjaan'] ?></td>
        <td>Rp.<?php echo number_format($b['nilai_pekerjaan'])
?>,-</td>
        <td><?php echo $b['nm_kontraktor'] ?></td>
        <td><?php echo $b['keterangan'] ?></td>
        <td><?php echo $b['stat'] ?></td>
<!--        <td>

        <?php
//            $det=mysql_query("select sum (prog_ming) from
tb_progress where no_kontrak=$b['no_kontrak'] group by no_kontrak")or
die(mysql_error());
//            $d=mysql_fetch_array($det);
//            echo $d[0]
?>
        </td>
-->
        <td class="dropdown">

```

```

        <button class="btn btn-primary dropdown-toggle"
type="button" data-toggle="dropdown">Menu Opsi
        <span class="caret"></span></button>
        <ul class="dropdown-menu">
            <li><a href="detail.php?id=<?php echo
$b[no_kontrak]; ?>">Detail</a></li>
            <li><a href="Kegiatan.php?id=<?php echo
$b[no_kontrak]; ?>">Kegiatan</a></li>
            <?php
            //echo $jabatan;
            if($jabatan != "JBN01")
            {
            ?>
            <li><a href="mingguan.php?id=<?php echo
$b[no_kontrak]; ?>">Monitoring</a></li>
            <?php
            }
            ?>
            <li><a href="addendum.php?id=<?php echo
$b[no_kontrak]; ?>">Addendum</a></li>
        </ul>
    </td>
    <td><a href="pekerjaan_edit.php?id=<?php echo
$b[no_kontrak]; ?>" class="btn btn-warning">Edit</a></td>
    <td><a href='pekerjaan_hapus.php?id=<?php echo
$b[no_kontrak]; ?>' onclick="return confirm('Apakah anda yakin ingin
menghapus data ini ??');" class="btn btn-danger">Hapus</a></td>
</tr>

```

```

        <?php
    }
    ?>
</tbody>
</table>
</div>
<ul class="pagination">
    <?php
        for($x=1;$x<=$halaman;$x++){
            ?>
            <li><a href="?page=<?php echo $x ?>"><?php
echo $x ?></a></li>
            <?php
                }
            ?>
        </ul>
<!-- modal input -->
<div id="myModal" class="modal fade">
    <div class="modal-dialog">
        <div class="modal-content">
            <div class="modal-header">
                <button type="button" class="close" data-
dismiss="modal" aria-hidden="true">&times;</button>
                <h4 class="modal-title">Tambah Pekerjaan
Baru</h4>
            </div>
            <div class="modal-body">
                <form name="reg" action="pekerjaan_tmbh.php"
onSubmit="return validasi()" method="get">
                    <div class="form-group">
                        <label>Nomor Kontrak</label>

```

```
        <input          name="no_kontrak"
type="text" class="form-control" placeholder="Nomor Kontrak ..">
```

```
</div>
```

```
<div class="form-group">
```

```
        <label          for="tanggal">Tanggal
Kontrak</label>
```

```
        <input          name="tgl_kontrak"
type="text"  class="form-control"  id="tanggal"  placeholder="Tanggal
Kontrak .."/>
```

```
</div>
```

```
<div class="form-group">
```

```
        <label>Nama Pekerjaan</label>
        <input          name="nama_pekerjaan"
type="text" class="form-control" placeholder="Nama Pekerjaan ..">
```

```
</div>
```

```
<div class="form-group">
```

```
        <label>Nilai Pekerjaan</label>
        <input          name="nilai_pekerjaan"
type="text" class="form-control" placeholder="Nilai Pekerjaan">
```

```
</div>
```

```
<div class="form-group">
```

```
        <table>
            <tr>
                <td>
```

```
<label>Pengawas</label>
```

```
</td>
```

```
</tr>
```

```
<tr>
```

```
<td>
```



```

        <input type="submit" class="btn btn-
primary" value="Simpan">
    </div>
</form>
</div>
</div>
</div>

```

3. Ganti Password

```

<?php
session_start();
include 'config.php';
?>
    <link rel="stylesheet" href="../assets/bootstrap-
3.3.6/css/bootstrap.min.css" type="text/css"/>
    <link href="../assets/bootstrap-3.3.6/css/bootstrap-theme.min.css"
rel="stylesheet">
    <link href="../assets/css/style.css" rel="stylesheet">
    <link rel="stylesheet" type="text/css" href="../assets/js/jquery-ui/jquery-
ui.css">
    <script type="text/javascript" src="../assets/js/jquery.js"></script>
    <script type="text/javascript" src="../assets/js/jquery-ui/jquery-
ui.js"></script>

<h3><span class="glyphicon glyphicon-briefcase"></span> Password</h3>
<br/><br/>
<?php
if(isset($_GET['pesan'])){
    $pesan=mysql_real_escape_string($_GET['pesan']);
    if($pesan=="gagal"){

```

```

        echo "<div class='alert alert-danger'>Password gagal di ganti !!
Periksa Kembali Password yang anda masukkan !!</div>";
    }else if($pesan=="tdksama"){
        echo "<div class='alert alert-warning'>Password yang anda
masukkan tidak sesuai !! silahkan ulangi !! </div>";
    }else if($pesan=="oke"){
        echo "<div class='alert alert-success'>Password berhasil di ubah
</div>";
    }
}
?>

```



```
<div class="col-md-5 col-md-offset-3">
```

```

    <form action="ganti_pass_act.php" method="post">
        <div class="form-group">
            <input name="user" type="hidden" value="<?php echo
$_SESSION['username']; ?>">
        </div>
        <div class="form-group">
            <label>Password Lama</label>
            <input name="lama" type="password" class="form-
control" placeholder="Password Lama ..">
        </div>
        <div class="form-group">
            <label>Password Baru</label>
            <input name="baru" type="password" class="form-control"
placeholder="Password Baru ..">
        </div>
        <div class="form-group">
            <label>Ulangi Password</label>

```

```
        <input name="ulang" type="password" class="form-
control" placeholder="Ulangi Password ..">
    </div>
    <div class="form-group">
        <label></label>
        <input type="submit" class="btn btn-info"
value="Simpan">
        <input type="reset" class="btn btn-danger" value="reset">
    </div>

</form>
</div>
```