

**RANCANG BANGUN APLIKASI UNTUK PERENCANAAN
BIAYA DAN WAKTU PROYEK SISTEM INFORMASI**

TUGAS AKHIR



Nama : Alex Slamet Suhamto

NIM : 07.41010.0004

Program : S1 (Strata Satu)

Jurusan : Sistem Informasi

SEKOLAH TINGGI

MANAJEMEN INFORMATIKA & TEKNIK KOMPUTER

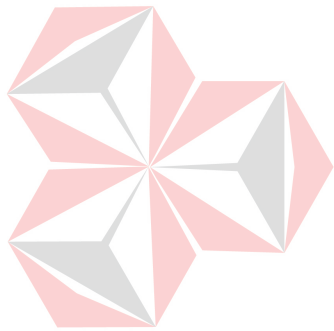
SURABAYA

2011

**RANCANG BANGUN APLIKASI UNTUK PERENCANAAN
BIAYA DAN WAKTU PROYEK SISTEM INFORMASI**

TUGAS AKHIR

**Diajukan sebagai salah satu syarat untuk menyelesaikan
Program Sarjana Komputer**



UNIVERSITAS
Dinamika

Oleh:

Nama : Alex Slamet Suhamto

NIM : 07.41010.0004

Program : S1 (Strata Satu)

Jurusan : Sistem Informasi

SEKOLAH TINGGI

MANAJEMEN INFORMATIKA & TEKNIK KOMPUTER

SURABAYA

2011



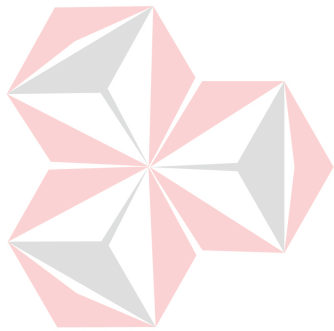
"Yesterday is History.."

"Tomorrow a Mystery.."

"Today is a Gift.. Thats why it's called the Present"

UNIVERSITAS

Dinamika



Kupersembahkan kepada

Ayah, Ibu, dan keluargaku

Beserta semua orang-orang terdekat

UNIVERSITAS
Dinamika

Tugas Akhir

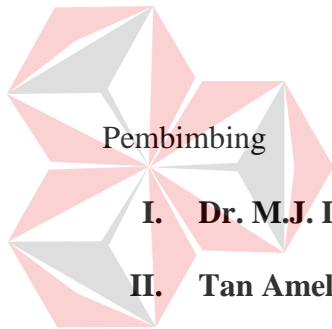
**RANCANG BANGUN APLIKASI UNTUK PERENCANAAN BIAYA DAN
WAKTU PROYEK SISTEM INFORMASI**

dipersiapkan dan disusun oleh:

Alex Slamet Suhamto

NIM : 07.41010.0004

Telah diperiksa, diuji, dan disetujui oleh Dewan Penguji
pada : Juni 2011



Susunan Dewan Penguji

Pembimbing

I. Dr. M.J. Dewiyani Sunarto _____

II. Tan Amelia, S.Kom.,M.MT. _____

Penguji

I. Haryanto Tanuwijaya, S.Kom.,M.MT. _____

II. Tutut Wurijanto, M.Kom. _____

Tugas Akhir ini telah diterima sebagai salah satu persyaratan
untuk memperoleh gelar Sarjana

Pantjawati Sudarmaningtyas, S.Kom.
Pembantu Ketua Bidang Akademik

PERNYATAAN

Dengan ini saya menyatakan dengan benar, bahwa Tugas Akhir ini adalah asli karya saya, bukan plagiat baik sebagian maupun apalagi keseluruhan. Karya atau pendapat orang lain yang ada dalam Tugas Akhir ini adalah semata hanya rujukan yang dicantumkan dalam Daftar Pustaka saya. Apabila dikemudian hari ditemukan adanya tindakan plagiat pada karya Tugas Akhir ini, maka saya bersedia untuk dilakukan pencabutan terhadap gelar kesarjanaan yang telah diberikan kepada saya.



UNIVERSITAS
Dinamika
Surabaya, 8 Juni 2011

Alex Slamet Suhamto

NIM : 07.41010.0004

ABSTRAK

Dalam proses pembuatan sistem informasi diperlukan perencanaan biaya dan waktu agar dapat menghasilkan sistem yang baik. Salah satu jenis dari perencanaan biaya dan waktu adalah menghasilkan perhitungan estimasi biaya dan pengembangan jadwal. Hasil yang baik akan membuat pihak konsumen merasa puas dan perusahaan yang mengerjakan sistem informasi mendapatkan kepercayaan dari konsumen.

Sistem ini dibangun untuk menghasilkan perhitungan estimasi biaya yang efektif dengan metode *Cost of Quality* dan pengelolaan waktu yang baik dengan metode *Critical Path*. Manajer maupun kepala proyek dapat melihat dan mengontrol kondisi dari sebuah proyek yang sedang berjalan agar tidak melewati batas yang telah direncanakan.

Dari implementasi dan evaluasi yang telah dilakukan, sistem yang telah dapat menampilkan informasi estimasi biaya proyek namun masih dibawah biaya aktual dan pengelolaan waktu dengan metode *Cost of Quality* dan *Critical Path* sehingga pengerjaan proyek dapat berjalan dengan lancar.

Kata-kata kunci : sistem informasi, perencanaan proyek, *cost of quality*, *critical path*.

KATA PENGANTAR

Pertama-tama penulis panjatkan puji syukur ke hadirat Tuhan Yang Maha Esa karena atas berkat dan rahmat-Nya akhirnya penulis dapat menyelesaikan laporan Tugas Akhir ini dengan sebaik-baiknya. Penulis membuat laporan Tugas Akhir yang berjudul “Rancang Bangun Aplikasi Untuk Perencanaan Biaya Dan Waktu Proyek Sistem Informasi (Studi Kasus STIKOM Surabaya)” ini sebagai salah satu syarat dalam menyelesaikan Program Studi Sarjana Komputer di Sekolah Tinggi Manajemen Informatika & Teknik Komputer Surabaya.

Dalam pelaksanaan Tugas Akhir maupun pembuatan laporan Tugas Akhir ini, penulis mendapatkan bantuan dari berbagai pihak. Oleh karena itu, penulis ingin mengucapkan terima kasih kepada:

1. Kedua orang tua dan keluarga yang telah memberikan motivasi dan dorongan moril sehingga penulis dapat menyelesaikan laporan Tugas Akhir ini.
2. Saudara kembar penulis, yang menjadi sumber motivasi terbesar penulis dan saudari penulis yang telah membantu penulis dalam mengerjakan laporan Tugas Akhir ini.
3. Semua pengajar penulis mulai SD hingga kuliah yang memberikan ilmu kepada penulis selama ini.
4. Ibu Tan Amelia, S.Kom.,M.MT selaku dosen pembimbing II yang senantiasa membimbing penulis dalam mengerjakan laporan maupun aplikasi Tugas Akhir ini.

5. Ibu Dr. M.J. Dewiyani S. selaku dosen pembimbing I yang senantiasa memberikan dorongan moril selama penulis mengerjakan laporan Tugas Akhir ini.
6. Valent Roby, Rima Izzul Millah, Frans, dan Eric yang telah membantu dan memberikan motivasi kepada penulis dalam penyelesaian laporan Tugas Akhir ini.
7. Pihak SSI sebagai tempat studi kasus penulis baik yang sekarang ada maupun yang sudah tidak ada lagi di SSI.
8. Teman-teman terdekat penulis yang senantiasa bersama selama empat tahun di STIKOM, Heru, Yohan, Subiyanto, Hendry, dan yang lainnya.
9. Semua pihak yang telah membantu dalam pembuatan laporan ini, baik secara langsung maupun secara tidak langsung.

Penulis menyadari bahwa dalam laporan Tugas Akhir ini masih banyak terdapat kekurangan. Oleh karena itu, penulis memohon saran dan kritik yang bersifat konstruktif dari semua pihak untuk perbaikan penulis di masa mendatang.

Penulis juga memohon maaf yang sebesar-besarnya jika ada kata-kata yang menyinggung atau menyakiti hati para pembaca. Akhir kata, penulis mengucapkan terima kasih atas perhatiannya. Semoga laporan Tugas Akhir ini dapat bermanfaat bagi para pembaca.

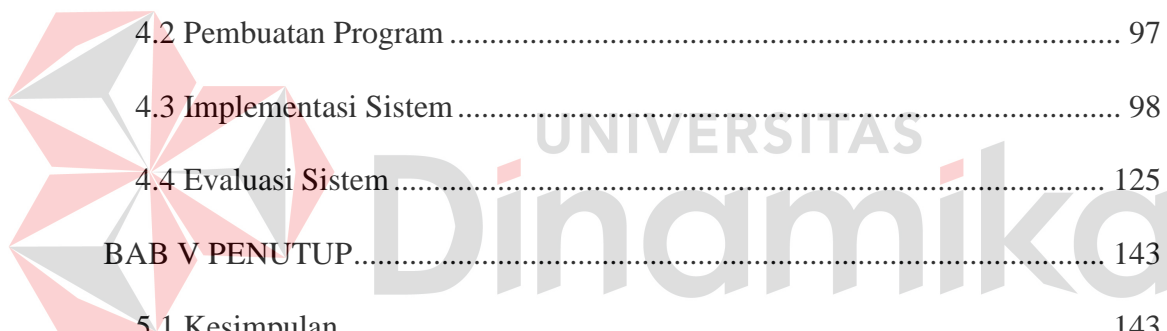
Surabaya, Juni 2011

Penulis

DAFTAR ISI

	Halaman
ABSTRAK	vi
KATA PENGANTAR	vii
DAFTAR ISI.....	ix
DAFTAR TABEL.....	xi
DAFTAR GAMBAR	xvi
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang Masalah.....	1
1.2 Perumusan Masalah	2
1.3 Pembatasan Masalah	2
1.4 Tujuan	3
1.5 Sistematika Penulisan	3
BAB II LANDASAN TEORI.....	5
2.1 Konsep Dasar Sistem Informasi.....	5
2.2 Manajemen Proyek Sistem Informasi	6
2.3 Pemegang Kepentingan Proyek	8
2.4 Perencanaan Biaya	9
2.5 Metode <i>Cost of Quality</i>	10
2.6 Perencanaan Waktu.....	14
2.7 Metode <i>Critical Path</i>	15
2.8 <i>Database</i>	20
2.9 <i>Testing</i>	20
BAB III METODE PENELITIAN DAN PERANCANGAN SISTEM	22

3.1 Metode Penelitian.....	22
3.2 Identifikasi Masalah	24
3.3 Analisa Kebutuhan	25
3.3 Analisa dan Perancangan Sistem	28
3.4 <i>Data Flow Diagram</i> (DFD)	40
3.5 Pemodelan Database	47
3.6 Desain <i>Input/Output</i>	58
3.7 Desain Uji Coba	86
BAB IV IMPLEMENTASI DAN EVALUASI.....	96
4.1 Kebutuhan Sistem	96
4.2 Pembuatan Program	97
4.3 Implementasi Sistem	98
4.4 Evaluasi Sistem	125
BAB V PENUTUP.....	143
5.1 Kesimpulan	143
5.2 Saran.....	143
DAFTAR PUSTAKA	145



DAFTAR TABEL

	Halaman
Tabel 2.1 Tahapan Dalam Manajemen Proyek	7
Tabel 2.2 Parameter Pada Metode <i>Cost of Quality</i>	13
Tabel 2.3 Tabel Perhitungan Estimasi Biaya Dengan Cara Manual	13
Tabel 2.4 Tabel Perhitungan Estimasi Biaya Dengan Metode <i>Cost of Quality</i>	14
Tabel 2.5 Tabel Kegiatan Dalam Proyek Sistem	17
Tabel 2.6 Tabel Hasil Proses <i>Forward Pass</i>	18
Tabel 2.7 Tabel Hasil Proses <i>Backward Pass</i>	19
Tabel 2.8 Tabel <i>Slack</i> Dari Proses <i>Forward Pass</i> dan <i>Backward Pass</i>	19
Tabel 3.1 Tabel Parameter <i>Cost of Quality</i> yang digunakan SSI.....	27
Tabel 3.2 Tabel UserPPIC	48
Tabel 3.3 Tabel Kota.....	49
Tabel 3.4 Tabel Customer.....	49
Tabel 3.5 Tabel Project	50
Tabel 3.6 Tabel Surat_Kontrak	50
Tabel 3.7 Tabel Parameter_Biaya	51
Tabel 3.8 Tabel Level	51
Tabel 3.9 Tabel Kompetensi	51
Tabel 3.10 Tabel Task.....	52
Tabel 3.11 Tabel Sub_Task.....	52
Tabel 3.12 Tabel Resource.....	53
Tabel 3.13 Tabel Resource_Level.....	53
Tabel 3.14 Tabel Project_Cost.....	54

Tabel 3.15 Tabel Hasil_Project.....	54
Tabel 3.16 Tabel App_Project	54
Tabel 3.17 Tabel Sub_Task_Predecessor	55
Tabel 3.18 Tabel History_Coding.....	55
Tabel 3.19 Tabel Testing.....	56
Tabel 3.20 Tabel Training.....	56
Tabel 3.21 Tabel Status_Sub_Task.....	57
Tabel 3.22 Tabel Sub2_Task.....	57
Tabel 3.23 Tabel Sub3_Task.....	58
Tabel 3.24 Fungsi Objek <i>Form Login</i>	58
Tabel 3.25 Fungsi Objek <i>Form Master User</i>	59
Tabel 3.26 Fungsi Objek <i>Form Menu Utama Market</i>	60
Tabel 3.27 Fungsi Objek <i>Form Master Kota</i>	61
Tabel 3.28 Fungsi Objek <i>Form Master Customer</i>	62
Tabel 3.29 Fungsi Objek <i>Form Master Project</i>	63
Tabel 3.30 Fungsi Objek <i>Form Pencarian Customer</i>	64
Tabel 3.31 Fungsi Objek <i>Form Surat Kontrak</i>	65
Tabel 3.32 Fungsi Objek <i>Form Menu Utama Manajer</i>	65
Tabel 3.33 Fungsi Objek <i>Form Master Kompetensi</i>	66
Tabel 3.34 Fungsi Objek <i>Form Master Parameter</i>	67
Tabel 3.35 Fungsi Objek <i>Form Master Level</i>	68
Tabel 3.36 Fungsi Objek <i>Form Master Resource</i>	69
Tabel 3.37 Fungsi Objek <i>Form Pencarian User</i>	70
Tabel 3.38 Fungsi Objek <i>Form Master Cost</i>	71

Tabel 3.39 Fungsi Objek <i>Form Resource Level</i>	71
Tabel 3.40 Fungsi Objek <i>Form View and App Task</i>	72
Tabel 3.41 Fungsi Objek <i>Form Summary</i>	73
Tabel 3.42 Fungsi Objek <i>Form Review Project</i>	74
Tabel 3.43 Fungsi Objek <i>Form Project Leader Taskboard</i>	74
Tabel 3.44 Fungsi Objek <i>Form Master Sub Task</i>	75
Tabel 3.45 Fungsi Objek <i>Form Pencarian Predesesor Sub Task</i>	77
Tabel 3.46 Fungsi Objek <i>Form Down Level Sub Task</i>	77
Tabel 3.47 Fungsi Objek <i>Form Project Team Taskboard</i>	78
Tabel 3.48 Fungsi Objek <i>Form History Work</i>	79
Tabel 3.49 Fungsi Objek <i>Form Project Test Taskboard</i>	80
Tabel 3.50 Fungsi Objek <i>Form History Test</i>	81
Tabel 3.51 Fungsi Objek <i>Form Project Train Taskboard</i>	81
Tabel 3.52 Fungsi Objek <i>Form History Train</i>	82
Tabel 3.53 Fungsi Objek <i>Form Approve Sub Task</i>	83
Tabel 3.54 Fungsi Objek <i>Form View History</i>	84
Tabel 3.55 Fungsi Objek <i>Form Hasil Project</i>	85
Tabel 3.56 Fungsi Objek <i>Form Laporan</i>	85
Tabel 3.57 Data User.....	87
Tabel 3.58 Desain Uji Coba <i>Form Login</i>	87
Tabel 3.59 Data Proyek.....	88
Tabel 3.60 Desain Uji Coba <i>Form Master Project</i>	88
Tabel 3.61 Data Biaya.....	89
Tabel 3.62 Desain Uji Coba <i>Form Master Cost</i>	89

Tabel 3.63 Desain Uji Coba Perbandingan Dengan Dan Tanpa Metode.....	89
Tabel 3.64 Data <i>Resource Level</i>	90
Tabel 3.65 Desain Uji Coba <i>Form Resource Level</i>	90
Tabel 3.66 Data <i>Resource Level</i>	91
Tabel 3.67 Desain Uji Coba <i>Form Resource Level</i>	91
Tabel 3.68 Data Jalur	91
Tabel 3.69 Desain Uji Coba <i>Form Master Sub Task</i>	92
Tabel 3.70 Data Jalur Normal	92
Tabel 3.71 Desain Uji Coba <i>Form Summary</i>	93
Tabel 3.72 Data <i>History</i>	93
Tabel 3.73 Desain Uji Coba <i>Form History Work</i>	94
Tabel 3.74 Data <i>History</i>	94
Tabel 3.75 Desain Uji Coba <i>Form History</i>	94
Tabel 3.76 Data Hasil Proyek	95
Tabel 3.77 Desain Uji Coba <i>Form Hasil Proyek</i>	95
Tabel 4.1 <i>Task AAK</i>	126
Tabel 4.2 <i>Task Keuangan</i>	126
Tabel 4.3 <i>Task HRD</i>	126
Tabel 4.4 <i>Task KRS Online</i>	126
Tabel 4.5 <i>Task Sicyca</i>	127
Tabel 4.6 Data User.....	128
Tabel 4.7 Hasil Uji Coba <i>Form Login</i>	128
Tabel 4.8 Data Proyek.....	129
Tabel 4.9 Hasil Uji Coba <i>Form Master Project</i>	129

Tabel 4.10 Data Biaya.....	130
Tabel 4.11 Hasil Uji Coba <i>Form Master Cost</i>	131
Tabel 4.12 Hasil Uji Coba Perbandingan Dengan Dan Tanpa Metode	132
Tabel 4.13 Data <i>Resource Level</i>	133
Tabel 4.14 Hasil Uji Coba <i>Form Resource Level</i>	133
Tabel 4.15 Data <i>Resource Level</i>	134
Tabel 4.16 Hasil Uji Coba <i>Form Resource Level</i>	134
Tabel 4.17 Data Jalur	135
Tabel 4.18 Hasil Uji Coba <i>Form Master Sub Task</i>	135
Tabel 4.19 Kegiatan Pada <i>Task AAK</i> dan Keuangan	137
Tabel 4.20 Hasil Uji Coba <i>Form Summary</i>	137
Tabel 4.21 Data <i>History</i>	139
Tabel 4.22 Hasil Uji Coba <i>Form History Work</i>	139
Tabel 4.23 Data <i>History</i>	140
Tabel 4.24 Hasil Uji Coba <i>Form History</i>	140
Tabel 4.25 Data Hasil Proyek	142
Tabel 4.26 Hasil Uji Coba <i>Form Hasil Proyek</i>	142

DAFTAR GAMBAR

	Halaman
Gambar 2.1 Pemegang Kepentingan Proyek	8
Gambar 3.1 Diagram Blok Proses Perencanaan Biaya dan Waktu.....	22
Gambar 3.2 Flowchart metode Cost of Quality	30
Gambar 3.3 Flowchart metode Critical Path.....	32
Gambar 3.4 Bagan Alir Program untuk Membuat Proyek.....	34
Gambar 3.5 Bagan Alir Program untuk Merancang Jadwal Kerja Proyek	36
Gambar 3.6 Bagan Alir Program untuk Menjalankan Jadwal Kerja Proyek	38
Gambar 3.7 Bagan Alir Program untuk Penutupan Proyek	39
Gambar 3.8 Diagram Berjenjang	40
Gambar 3.9 <i>Context Diagram</i>	42
Gambar 3.10 DFD Level 0 Aplikasi Untuk Perencanaan Biaya dan Waktu	43
Gambar 3.11 DFD Level 1 Subsistem Membuat Proyek.....	44
Gambar 3.12 DFD Level 1 Subsistem Merancang Proyek	45
Gambar 3.13 DFD Level 1 Subsistem Mengimplementasikan Proyek	46
Gambar 3.14 <i>Conceptual Data Model</i>	47
Gambar 3.15 <i>Physical Data Model</i>	48
Gambar 3.16 Desain <i>Form Login</i>	58
Gambar 3.17 Desain <i>Form Master User</i>	59
Gambar 3.18 Desain <i>Form Menu Utama Market</i>	60
Gambar 3.19 Desain <i>Form Master Kota</i>	61
Gambar 3.20 Desain <i>Form Master Customer</i>	62
Gambar 3.21 Desain <i>Form Master Project</i>	63

Gambar 3.22 Desain <i>Form Pencarian Customer</i>	64
Gambar 3.23 Desain <i>Form Surat Kontrak</i>	64
Gambar 3.24 Desain <i>Form Menu Utama Manajer</i>	65
Gambar 3.25 Desain <i>Form Master Kompetensi</i>	66
Gambar 3.26 Desain <i>Form Master Parameter</i>	67
Gambar 3.27 Desain <i>Form Master Level</i>	68
Gambar 3.28 Desain <i>Form Master Resource</i>	69
Gambar 3.29 Desain <i>Form Pencarian User</i>	70
Gambar 3.30 Desain <i>Form Master Cost</i>	70
Gambar 3.31 Desain <i>Form Resource Level</i>	71
Gambar 3.32 Desain <i>Form View and App Task</i>	72
Gambar 3.33 Desain <i>Form Summary</i>	73
Gambar 3.34 Desain <i>Form Review Project</i>	73
Gambar 3.35 Desain <i>Form Project Leader Taskboard</i>	74
Gambar 3.36 Desain <i>Form Master Sub Task</i>	75
Gambar 3.37 Desain <i>Form Pencarian Predesesor Sub Task</i>	76
Gambar 3.38 Desain <i>Form Down Level Sub Task</i>	77
Gambar 3.39 Desain <i>Form Project Team Taskboard</i>	78
Gambar 3.40 Desain <i>Form History Work</i>	79
Gambar 3.41 Desain <i>Form Project Test Taskboard</i>	80
Gambar 3.42 Desain <i>Form History Test</i>	80
Gambar 3.43 Desain <i>Form Project Train Taskboard</i>	81
Gambar 3.44 Desain <i>Form History Train</i>	82
Gambar 3.45 Desain <i>Form Approve Sub Task</i>	83

Gambar 3.46 Desain <i>Form View History</i>	83
Gambar 3.47 Desain <i>Form Hasil Project</i>	84
Gambar 3.48 Desain <i>Form Laporan</i>	85
Gambar 4.1 <i>Form Menu Utama</i>	98
Gambar 4.2 <i>Form Login</i>	99
Gambar 4.3 <i>Form Master User</i>	100
Gambar 4.4 <i>Form Menu Utama Market</i>	101
Gambar 4.5 <i>Form Master Kota</i>	101
Gambar 4.6 <i>Form Master Customer</i>	102
Gambar 4.7 <i>Form Master Project</i>	103
Gambar 4.8 <i>Form Pencarian Customer</i>	103
Gambar 4.9 <i>Form Surat Kontrak</i>	104
Gambar 4.10 <i>Form Menu Utama Manajer</i>	105
Gambar 4.11 <i>Form Master Kompetensi</i>	105
Gambar 4.12 <i>Form Master Parameter</i>	106
Gambar 4.13 <i>Form Master Level</i>	107
Gambar 4.14 <i>Form Master Resource</i>	107
Gambar 4.15 <i>Form Pencarian User</i>	108
Gambar 4.16 <i>Form Master Cost</i>	109
Gambar 4.17 <i>Form Resource Level</i>	110
Gambar 4.18 <i>Form View and App Task</i>	110
Gambar 4.19 <i>Form Summary</i>	111
Gambar 4.20 <i>Form Review Project</i>	112
Gambar 4.21 <i>Form Project Leader Taskboard</i>	113

Gambar 4.22 <i>Form Master Sub Task</i>	114
Gambar 4.23 <i>Form Pencarian Predecessor Sub Task</i>	114
Gambar 4.24 <i>Form Detail Sub Task</i>	115
Gambar 4.25 <i>Form Project Team Taskboard</i>	115
Gambar 4.26 <i>Form History Work</i>	116
Gambar 4.27 <i>Form Project Test Taskboard</i>	117
Gambar 4.28 <i>Form History Test</i>	117
Gambar 4.29 <i>Form Project Train Taskboard</i>	118
Gambar 4.30 <i>Form History Train</i>	119
Gambar 4.31 <i>Form Project Approve Sub Task</i>	119
Gambar 4.32 <i>Form View History</i>	120
Gambar 4.33 <i>Form Hasil Project</i>	121
Gambar 4.34 <i>Laporan Penutupan Project</i>	121
Gambar 4.35 <i>Laporan Project Cost</i>	122
Gambar 4.36 <i>Laporan Project Overview</i>	123
Gambar 4.37 <i>Laporan Project</i>	123
Gambar 4.38 <i>Laporan Project</i>	124
Gambar 4.39 <i>Laporan Project</i>	125
Gambar 4.40 <i>Hasil Test Case 1</i>	128
Gambar 4.41 <i>Hasil Test Case 2</i>	128
Gambar 4.42 <i>Hasil Test Case 4</i>	130
Gambar 4.43 <i>Hasil Test Case 7</i>	131
Gambar 4.44 <i>Estimasi Biaya Dengan Sistem Lama SSI</i>	132
Gambar 4.45 <i>Hasil Test Case 8</i>	133

Gambar 4.46 Hasil <i>Test Case</i> 9.....	134
Gambar 4.47 Hasil <i>Test Case</i> 2.....	136
Gambar 4.48 Hasil <i>Test Case</i> 3.....	136
Gambar 4.49 Hasil <i>Test Case</i> 4.....	136
Gambar 4.50 Diagram Dari Kegiatan Pada <i>Task</i> AAK dan Keuangan	137
Gambar 4.51 Gantt Chart <i>Critical Path</i>	138
Gambar 4.52 <i>Slack</i> Pada <i>Task</i> AAK Dan Keuangan	138
Gambar 4.53 Hasil <i>Test Case</i> 1.....	139
Gambar 4.54 Hasil <i>Test Case</i> 2.....	141
Gambar 4.55 Hasil <i>Test Case</i> 3.....	141
Gambar 4.56 Hasil <i>Test Case</i> 4.....	142



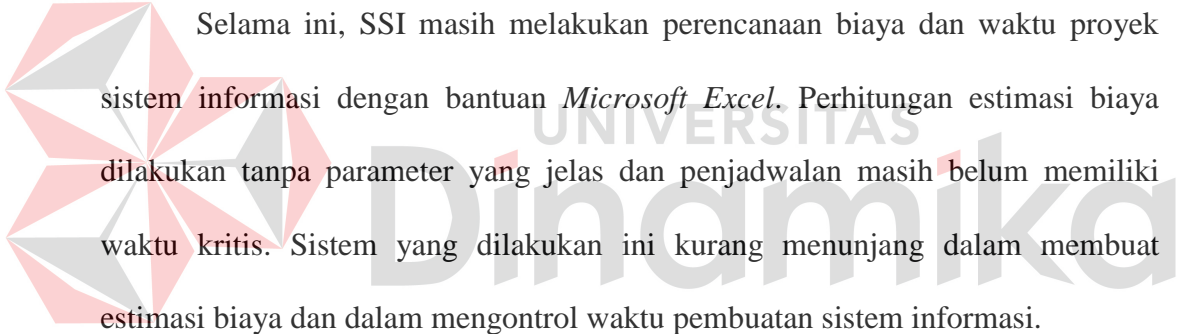
UNIVERSITAS
Dinamika

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Dalam proses pembuatan sistem informasi diperlukan perencanaan biaya dan waktu agar dapat menghasilkan sistem yang baik. Hasil yang baik akan membuat pihak konsumen merasa puas dan perusahaan yang mengerjakan sistem informasi mendapatkan kepercayaan dari konsumen. Di STIKOM Surabaya, bagian yang menangani pembuatan sistem informasi adalah Solusi Sistem Informasi (SSI). SSI mengerjakan proyek sistem informasi diluar STIKOM.



Selama ini, SSI masih melakukan perencanaan biaya dan waktu proyek sistem informasi dengan bantuan *Microsoft Excel*. Perhitungan estimasi biaya dilakukan tanpa parameter yang jelas dan penjadwalan masih belum memiliki waktu kritis. Sistem yang dilakukan ini kurang menunjang dalam membuat estimasi biaya dan dalam mengontrol waktu pembuatan sistem informasi.

Dalam membuat perhitungan estimasi biaya proyek sistem informasi, terdapat beberapa metode yang dapat digunakan, salah satunya adalah metode *Cost of Quality* (PMI, 2008:169). *Cost of Quality* adalah salah satu metode yang dapat digunakan untuk menghitung biaya sebuah proyek sistem informasi dengan berfokus pada kualitas dan tidak hanya berfokus pada biaya di dalam pengerjaan proyek tetapi juga biaya di luar pengerjaan proyek (PMI, 2008:195). Sedangkan dalam hal proses pengembangan jadwal, metode yang paling umum dilakukan adalah metode *Critical Path*. Metode *Critical Path* menghitung waktu mulai (awal maupun terlambat) dan waktu akhir untuk semua aktivitas tanpa

memperhitungkan setiap keterbatasan sumber daya dengan menjalankan analisa *forward pass* dan *backward pass* (PMI, 2008:154). Metode *Critical Path* akan menghasilkan sebuah jalur kritis yang diharapkan dapat membantu peran manajemen dalam hal pengelolaan waktu *project*.

Berdasarkan uraian diatas, SSI membutuhkan sebuah aplikasi yang dapat digunakan untuk mengelola pembuatan proyek sistem informasi terutama dalam hal estimasi biaya dan pengembangan jadwal agar dengan rencana yang baik, proyek dapat berjalan dengan lancar dan menghasilkan keuntungan bagi perusahaan yang pada akhirnya membuat pelanggan merasa puas.

1.2 Perumusan Masalah

Berdasarkan latar belakang di atas, dapat dirumuskan permasalahan yaitu:

1. Bagaimana menghasilkan estimasi biaya yang dibutuhkan dengan metode *Cost of Quality*.
2. Bagaimana menghasilkan jalur kritis dalam pembuatan proyek sistem informasi dengan metode *Critical Path*.
3. Bagaimana merancang dan membangun sistem informasi perencanaan biaya dan waktu dengan metode *Cost of Quality* dan *Critical Path*.

1.3 Pembatasan Masalah

Batasan masalah dari sistem yang dibahas adalah sebagai berikut:

1. Perencanaan biaya berfokus pada estimasi sedangkan perencanaan waktu hanya berfokus pada pengembangan jadwal.
2. Proses mengurutkan aktivitas hanya memakai hubungan *finish to start*.

3. Level pada aplikasi ini dibatasi hingga 5 tingkat.
4. *Critical Path* tidak memakai percepatan/perlambanan proses.
5. Aplikasi yang dibuat tidak menggunakan enkripsi data dan keamanan lainnya.
6. Aplikasi yang dibuat berdasarkan kasus pada bagian Solusi Sistem Informasi STIKOM Surabaya.
7. Aplikasi berbasis *desktop* menggunakan program *Microsoft Visual Studio .NET* dengan menggunakan database *Microsoft SQL Express*.

1.4 Tujuan

Berdasarkan rumusan masalah yang ada, maka tujuan dari sistem ini adalah:

1. Mengetahui estimasi biaya yang akan dikeluarkan dalam pembuatan proyek sistem informasi.
2. Mengelola jadwal kerja dalam pembuatan proyek sistem informasi.
3. Membuat aplikasi perencanaan pembuatan proyek sistem informasi yang dapat menampilkan informasi estimasi biaya dan pengelolaan jadwal.

1.5 Sistematika Penulisan

Laporan Tugas Akhir (TA) ini ditulis dengan sistematika penulisan sebagai berikut:

Bab I : Pendahuluan

Bab ini berisi tentang latar belakang diambilnya topik TA, rumusan masalah dari topik TA, batasan masalah TA, dan tujuan dari TA ini.

Bab II : Landasan Teori

Bab ini menjelaskan daur hidup sistem, manajemen proyek secara umum, perencanaan biaya dan waktu, gambaran metode yang akan digunakan yaitu *Cost of Quality* dan *Critical Path*, dan dilanjutkan dengan penjelasan tentang *database*, *testing* dan *training*.

Bab III : Metode Penelitian dan Perancangan Sistem

Bab ini berisi penjelasan tentang tahap-tahap yang dikerjakan dalam penyelesaian TA yang terdiri dari observasi pendahuluan, *interview*/wawancara, studi pustaka, identifikasi masalah dan tujuan, analisa dan perancangan sistem, pembuatan *flowchart* metode yang digunakan, *data flow diagram*, desain ERD baik *conceptual data model* maupun *physical data model*, desain antarmuka, dan desain uji coba

Bab IV : Implementasi dan Evaluasi

Bab ini berisi kebutuhan sistem, proses implementasi dari sistem dan penjelasan tentang evaluasi dari sistem yang telah dibuat.

Bab V : Penutup

Bab ini berisi kesimpulan dan saran. Saran yang dimaksud adalah saran terhadap kekurangan dari aplikasi yang ada kepada pihak lain yang ingin meneruskan topik TA ini. Tujuannya adalah agar pihak lain tersebut dapat menyempurnakan aplikasi sehingga bisa menjadi lebih baik dan berguna.

BAB II

LANDASAN TEORI

2.1 Konsep Dasar Sistem Informasi

2.1.1 Sistem

Menurut Herlambang dan Tanuwijaya, definisi sistem dapat dibagi menjadi dua pendekatan, yaitu pendekatan secara prosedur, sistem didefinisikan sebagai kumpulan dari beberapa prosedur yang mempunyai tujuan tertentu. Sedangkan berdasarkan pendekatan komponen, sistem merupakan kumpulan dari komponen-komponen yang saling berkaitan untuk mencapai tujuan tertentu.

Dalam perkembangan sistem yang ada, sistem dibedakan menjadi dua jenis, yaitu sistem terbuka dan sistem tertutup. Sistem terbuka merupakan sistem yang dihubungkan dengan arus sumber daya luar dan tidak mempunyai elemen pengendali. Sedangkan sistem tertutup tidak mempunyai elemen pengontrol dan dihubungkan pada lingkungan sekitarnya (Herlambang dan Tanuwijaya, 2005:116).

Menurut Mustakini, sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. Suatu sistem mempunyai maksud tertentu. Ada yang menyebut maksud dari suatu sistem adalah untuk mencari tujuan, dan ada yang menyebut maksud dari suatu sistem adalah untuk mencari sasaran (Mustakini, 2001: 2).

2.1.2 Siklus Hidup Sistem

Konsep siklus hidup atau *System Development Life Cycle* (SLC) merupakan konsep pengembangan yang paling banyak digunakan para

pengembang sistem informasi. Konsep ini mengambil analogi siklus kehidupan manusia yaitu, proses kelahiran, tumbuh berkembang dan akhirnya wafat. Siklus memberikan gambaran bahwa siklus demikian dapat juga diterapkan pada sistem informasi.

Menurut McLeod, Ada 5 tahap pada SDLC yaitu perencanaan, analisis, desain, penerapan, dan penggunaan. Tahap perencanaan dilakukan untuk merencanakan apa yang akan dibuat saat akan melakukan pengembangan. Tahap analisis dilakukan untuk menentukan apakah akan merancang sistem baru atau memperbaiki sistem yang lama. Pada tahap desain, proses dan data yang dibutuhkan pada sistem yang baru digambarkan. Tahap penerapan adalah memasang sistem yang baru (ada 4 cara, percontohan, serentak, bertahap dan paralel). Tahap penggunaan adalah menggunakan, mengaudit dan memelihara sistem (Herlambang dan Tanuwijaya, 2005:180).

2.2 Manajemen Proyek Sistem Informasi

Proyek adalah suatu proses pengambilalihan sementara untuk membuat suatu produk, servis atau hasil yang unik. Proyek yang benar selalu mendefinisikan awal dan akhir. Akhir dari sebuah proyek dicapai ketika objektivitas proyek telah tercapai atau ketika proyek dihentikan karena objektivitas tidak atau tidak akan tercapai, atau ketika proyek tidak lagi dibutuhkan (PMI, 2008:5).

Definisi lain menurut Gray dan Larson menyebutkan bahwa proyek adalah usaha yang kompleks, tidak rutin, yang dibatasi waktu, anggaran, sumber daya

dan spesifikasi kinerja yang dirancang untuk memenuhi kebutuhan pelanggan (Gray dan Larson, 2007:3).

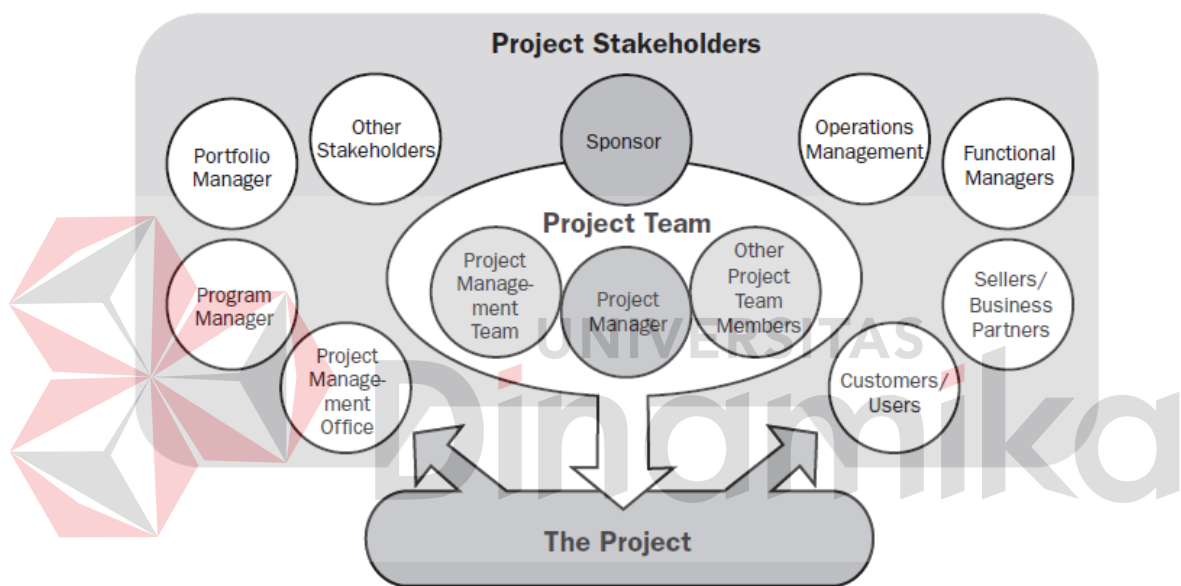
Manajemen Proyek adalah aplikasi dari pengetahuan, skill, alat, dan teknik dari aktivitas proyek untuk memenuhi kriteria proyek. Pada tabel 2.1 dapat terlihat 5 proses yang dilakukan dalam manajemen proyek yaitu inisiasi, perencanaan, eksekusi, pengawasan dan pengendalian, dan penutupan (PMI, 2008:43).

Tabel 2.1 Tahapan Dalam Manajemen Proyek

Knowledge Areas	Project Management Process Groups				
	Initiating Process Group	Planning Process Group	Executing Process Group	Monitoring & Controlling Process Group	Closing Process Group
4. Project Integration Management	4.1 Develop Project Charter	4.2 Develop Project Management Plan	4.3 Direct and Manage Project Execution	4.4 Monitor and Control Project Work 4.5 Perform Integrated Change Control	4.6 Close Project or Phase
5. Project Scope Management		5.1 Collect Requirements 5.2 Define Scope 5.3 Create WBS		5.4 Verify Scope 5.5 Control Scope	
6. Project Time Management		6.1 Define Activities 6.2 Sequence Activities 6.3 Estimate Activity Resources 6.4 Estimate Activity Durations 6.5 Develop Schedule		6.6 Control Schedule	
7. Project Cost Management		7.1 Estimate Costs 7.2 Determine Budget		7.3 Control Costs	
8. Project Quality Management		8.1 Plan Quality	8.2 Perform Quality Assurance	8.3 Perform Quality Control	
9. Project Human Resource Management		9.1 Develop Human Resource Plan	9.2 Acquire Project Team 9.3 Develop Project Team 9.4 Manage Project Team		
10. Project Communications Management	10.1 Identify Stakeholders	10.2 Plan Communications	10.3 Distribute Information 10.4 Manage Stakeholder Expectations	10.5 Report Performance	
11. Project Risk Management		11.1 Plan Risk Management 11.2 Identify Risks 11.3 Perform Qualitative Risk Analysis 11.4 Perform Quantitative Risk Analysis 11.5 Plan Risk Responses		11.6 Monitor and Control Risks	
12. Project Procurement Management		12.1 Plan Procurements	12.2 Conduct Procurements	12.3 Administer Procurements	12.4 Close Procurements

2.3 Pemegang Kepentingan Proyek

Pemegang kepentingan proyek adalah orang atau organisasi yang secara aktif terlibat di dalam proyek atau yang memiliki ketertarikan positif atau negatif dengan performa atau hasil dari proyek. Tim manajemen proyek harus mampu mengidentifikasi internal dan eksternal pemegang kepentingan dengan tujuan dapat memutuskan kebutuhan dan ekspektasi proyek. Untuk lebih jelasnya dapat dilihat pada Gambar 2.1.



Gambar 2.1 Pemegang Kepentingan Proyek

Dalam Gambar 2.1 dapat terlihat bahwa tim manajemen proyek, manajer proyek dan anggota tim proyek yang lain adalah pemegang kepentingan internal proyek sedangkan manajer portofolio, manajer program, kantor manajemen proyek, konsumen, penjual, manajer fungsi dan manajer operasional adalah pemegang kepentingan eksternal proyek.

Konsumen adalah orang atau organisasi yang akan menggunakan hasil dari proyek. Sponsor adalah orang atau grup yang memberikan sumber daya

finansial berupa tunai atau lainnya untuk sebuah proyek. Manajer portofolio bertanggung jawab pada pemerintahan tingkat atas untuk mengumpulkan proyek atau program. Program manajer adalah orang yang bertanggung jawab atas manajemen proyek untuk mengkoordinasikan dan mengontrol individu proyek. Kantor manajemen proyek adalah organisasi yang bertanggung jawab atas layanan pendukung *administrasi, training, mentoring* dan pelatihan dari manajer proyek, pendukung proyek, pengaturan sumber daya dan komunikasi tersentral. Manajer proyek adalah orang yang ditugaskan untuk menjalankan organisasi untuk mencapai tujuan proyek. Tugas dari manajer proyek adalah mengembangkan rencana manajemen proyek, menjaga proyek dalam jalur yang benar, mengidentifikasi, dan mengawasi resiko. Tim proyek adalah orang yang mengerjakan proyek namun tidak terlibat pada manajemen proyek. Tim proyek menggunakan kemampuannya untuk menyelesaikan proyek. Manajer fungsi adalah orang yang menjalankan tugas manajemen dengan administrasi seperti sumber daya manusia, finansial, akunting, dan pembelian. Manajer operasional adalah orang yang menjalankan tugas manajemen dalam area bisnis seperti pengembangan, desain, manufaktur, *testing*, atau *maintenance*. Penjual atau vendor adalah orang yang terkontrak untuk menyediakan komponen atau servis untuk kebutuhan proyek (PMI, 2008:23).

2.4 Perencanaan Biaya

Perencanaan biaya merupakan salah satu bagian manajemen biaya. Manajemen biaya membutuhkan pengertian yang lebih mendalam tentang struktur biaya suatu perusahaan. Manajer harus mampu menentukan kegiatan dan proses

jangka pendek dan jangka panjang (Hansen dan Mowen, 2000:2). Perencanaan biaya berisi 2 proses yaitu estimasi biaya dan penganggaran biaya (PMI, 2008:43).

Estimasi biaya adalah proses pengembangan perkiraan dari kebutuhan keuangan yang dibutuhkan untuk menyelesaikan aktivitas proses (PMI, 2008:168). Estimasi biaya membutuhkan masukan yaitu batasan masalah, jadwal proses, perencanaan sumber daya manusia, kemungkinan resiko, faktor pengembangan perusahaan dan aset organisasi. Teknik yang dapat digunakan dalam estimasi biaya diantaranya: *Expert Judgment, Analogous Estimating, Parametric Estimating, Bottom-up Estimating, Three-point Estimates, Reserve Analysis, Cost of Quality, Project Management Estimating Software*, dan *Vendor Bid Analysis*. Output yang dihasilkan adalah estimasi biaya aktivitas dan perubahan dokumen proyek (PMI, 2008:169). Dengan estimasi biaya, maka manajer maupun konsumen dapat mengetahui gambaran harga dari sebuah proyek yang dikerjakan.

2.5 Metode Cost of Quality

Cost of Quality adalah salah satu metode yang dapat digunakan dalam menentukan estimasi biaya proyek (PMI, 2008:169). *Cost of Quality* berisi semua biaya yang terjadi pada seluruh komponen produk dengan memperhatikan 4 hal, yaitu biaya pencegahan, biaya penilaian (untuk mencegah kegagalan) dan biaya kegagalan internal, biaya kegagalan eksternal (untuk kegagalan yang terjadi baik pada pengerjaan ataupun pada konsumen) (PMI, 2008:195). Menurut Gryna, Cost of Quality adalah biaya yang terasosiasikan dengan pencegahan, pencarian, dan pembetulan kerja yang gagal (Kaner, 1996:1).

Biaya pencegahan adalah biaya aktivitas yang didesain untuk menghindari kualitas yang buruk, salah satunya adalah kegagalan *coding*, desain yang gagal, kesalahan pada panduan pelanggan, dokumentasi yang jelek atau kode yang tidak dapat ditangani. Selain itu biaya pencegahan juga termasuk biaya *training*, dokumen, peralatan, dan waktu yang dibutuhkan dalam mengerjakan proyek. Biaya penilaian adalah biaya aktivitas yang didesain untuk mencari masalah kualitas seperti inspeksi *coding* dan beberapa tipe tes yang lain, termasuk *testing* yang gagal.

Biaya kegagalan terbagi menjadi 2 bagian, internal dan eksternal. Biaya kegagalan internal membahas biaya yang dikeluarkan karena kegagalan yang ditemukan ketika mengerjakan program. Diantaranya biaya pengerjaan kembali dan biaya yang telah dihabiskan dalam membuat sistem yang gagal. Biaya kegagalan eksternal adalah biaya yang timbul akibat kegagalan di tingkat konsumen. Biaya yang termasuk kegagalan eksternal adalah biaya ketidakstabilan program, biaya garansi, dan biaya yang muncul akibat kegagalan usaha yang diakibatkan oleh program (Kaner, 1996:1).

Dalam menghitung biaya dengan metode *Cost of Quality*, terdapat banyak contoh nyata pada perusahaan (Schiffauerova dan Thomson, 2006:10) diantaranya *United Technologies/Essex Group, British Aerospace Dynamics, ITT Europe, Ferranti Defense Systems, National Cash Register Company, ITT Corp.*, dan *Travenol Laboratories* menggunakan perhitungan biaya penilaian ditambah biaya pencegahan ditambah biaya kegagalan (PC+AC+FC). Namun beberapa perusahaan lain seperti *AT&T Bell Laboratories, Hydro Coatings, York International, Allis-Chalmers Corporation, Herbert Machine Tools, major*

electrical firm, North American Philips Consumer Electronics, Hermes Electronics, Banc One Corporation, Cascade Engineering, dan electronic manufacturer menggunakan perhitungan biaya penilaian ditambah biaya pencegahan ditambah biaya kegagalan internal ditambah biaya kegagalan eksternal (PC+AC+IFC+EFC).

Biaya dengan metode *Cost of Quality* (Kaner, 1996:2) adalah penjumlahan dari 4 elemen yaitu biaya pencegahan, biaya penilaian, biaya kegagalan internal dan biaya kegagalan eksternal sehingga dirumuskan sebagai berikut:

$$CoQ = \sum PC + \sum AC + \sum IFC + \sum EFC \dots\dots(2.1)$$

CoQ = *Cost of Quality*

PC = *Prevention Cost* (Biaya Pencegahan)

AC = *Appraisal Cost* (Biaya Penilaian)

IFC = *Internal Failure Cost* (Biaya Kegagalan Internal)

EFC = *External Failure Cost* (Biaya Kegagalan Eksternal)

Beberapa contoh parameter dari masing-masing kategori dapat dilihat pada tabel 2.2.

Tabel 2.2 Parameter Pada Metode *Cost of Quality*

<i>Prevention</i>	<i>Appraisal</i>
<ul style="list-style-type: none"> • Staff training • Requirements analysis • Early prototyping • Fault-tolerant design • Defensive programming • Usability analysis • Clear specification • Accurate internal documentation • Evaluation of the reliability of development tools (before buying them) or of other potential components of the product 	<ul style="list-style-type: none"> • Design review • Code inspection • Glass box testing • Black box testing • Training testers • Beta testing • Test automation • Usability testing • Pre-release out-of-box testing by customer service staff
<i>Internal Failure</i>	<i>External Failure</i>
<ul style="list-style-type: none"> • Bug fixes • Regression testing • Wasted in-house user time • Wasted tester time • Wasted writer time • Wasted marketer time • Wasted advertisements • Direct cost of late shipment • Opportunity cost of late shipment 	<ul style="list-style-type: none"> • Technical support calls • Preparation of support answer books • Refunds and replacement with updated product • Lost sales • PR work to soften drafts of harsh reviews • Lost customer goodwill • Warranty costs and other costs imposed by law

Sebagai contoh pada sebuah proyek sistem informasi yang akan dibuat. Jika metode ini tidak digunakan, seperti terlihat pada tabel 2.3, maka angka yang didapat adalah angka yang mentah dan tidak bisa dipertanggungjawabkan asal usul biaya tersebut.

Tabel 2.3 Tabel Perhitungan Estimasi Biaya Dengan Cara Manual

No	Kategori	Parameter	Biaya
1	Tidak Ada	Pengerjaan Program	Rp 4.000.000
2	Tidak Ada	Biaya Lain-lain	Rp 300.000

Total biaya dari perhitungan manual adalah 4.300.000 Rupiah

Namun dengan metode *Cost of Quality* pada tabel 2.4, penyajian biaya menjadi lebih lengkap, dibagi dalam parameter-parameter yang kecil dengan 4 kategori berbeda maka meski didapat hasil yang sama, tingkat kepercayaan konsumen akan lebih tinggi dibandingkan dengan hasil estimasi biaya yang dihitung secara manual.

Tabel 2.4 Tabel Perhitungan Estimasi Biaya Dengan Metode *Cost of Quality*

No	Kategori	Parameter	Biaya
1	Biaya Pencegahan	Analisis Kebutuhan	Rp 150.000
2	Biaya Pencegahan	Pengerjaan Program	Rp 2.000.000
3	Biaya Pencegahan	Desain	Rp 500.000
4	Biaya Pencegahan	Training	Rp 350.000
5	Biaya Penilaian	Penilaian Desain	Rp 100.000
6	Biaya Penilaian	Black box Testing	Rp 250.000
7	Biaya Kegagalan Internal	Pembetulan Kesalahan	Rp 200.000
8	Biaya Kegagalan External	Garansi	Rp 500.000
9	Biaya Kegagalan External	Kerugian akibat program	Rp 250.000

Total biaya dari perhitungan dengan metode adalah 4.300.000 Rupiah.

2.6 Perencanaan Waktu

Perencanaan waktu proyek adalah salah satu bagian dalam manajemen waktu. Perencanaan waktu berisi 5 proses yaitu mendefinisikan aktivitas, mengurutkan aktivitas, mengestimasi sumber daya, mengestimasi durasi, dan mengembangkan jadwal (PMI, 2008:129). Mengurutkan aktivitas adalah proses identifikasi dan dokumentasi hubungan antar aktivitas proyek. Aktivitas diurutkan dalam hubungan *logical*. Semua aktivitas kecuali awal dan akhir harus terhubung oleh minimal satu predesesor dan satu suksesor. Hubungan antar aktivitas dapat berupa 4 hal yaitu *finish to start*, *finish to finish*, *start to start* dan *start to finish*. Hubungan *finish to start* berarti aktivitas suksesor tidak dapat dimulai sebelum selesainya aktivitas predesesor, hubungan *finish to finish* berarti aktivitas suksesor tidak dapat selesai sebelum selesainya aktivitas predesesor, hubungan *start to start* berarti aktivitas suksesor tidak dapat dimulai sebelum mulainya aktivitas predesesor, dan hubungan *start to finish* berarti aktivitas suksesor tidak dapat diakhiri sebelum dimulainya aktivitas predesesor (PMI, 2008:138).

Mengembangkan jadwal adalah proses untuk menganalisis urutan aktivitas, durasi, kebutuhan sumber daya dan batasan jadwal untuk membuat jadwal proyek. Inputan dalam jadwal berupa aktivitas, durasi dan *resource*. Pengembangan jadwal membutuhkan penjelasan dan revisi dari estimasi durasi dan estimasi *resource* untuk menyetujui jadwal proyek yang diperlihatkan dari kemajuan kerja (PMI, 2008:152).

Beberapa metode yang dapat digunakan dalam mengembangkan jadwal diantaranya: *Schedule network analysis*, *Critical path method*, *Critical chain method*, *Resource leveling*, *What-if scenario analysis*, *Applying leads and lags*, *Schedule compression*, dan *Scheduling tool* (PMI, 2008:154).

2.7 Metode *Critical Path*

Metode *Critical path* bertujuan mengetahui jalur kritis dari sebuah proyek dengan menghitung waktu mulai awal, waktu mulai terlambat, waktu akhir awal, waktu akhir terlambat, dan *slack* untuk semua aktivitas tanpa memperhitungkan setiap keterbatasan sumber daya dengan menjalankan analisa *forward pass* dan *backward pass* (PMI, 2008:154).

Menurut Hillier dan Lieberman, waktu awal untuk sebuah kejadian adalah (dugaan) waktu dimana kejadian akan terjadi jika kegiatan sebelumnya dimulai se-awal mungkin. Waktu terlambat untuk suatu kejadian adalah (dugaan) waktu terakhir dimana kejadian akan terjadi, tanpa penundaan penyelesaian proyeknya di atas waktu paling awalnya. Kelambanan (*slack*) suatu kejadian adalah perbedaan antara waktu terlambat dan waktu awal.

Lintasan kritis suatu proyek adalah lintasan dalam suatu jaringan kerja sedemikian sehingga kegiatan pada lintasan ini memiliki *slack* nol (semua kegiatan dan kejadian yang memiliki *slack* nol akan terdapat dalam lintasan kritis, tetapi yang lainnya tidak) (Hillier dan Lieberman, 1994:376). Dengan kata lain, lintasan kritis mengidentifikasi semua kegiatan kritis dari proyek tersebut (Taha, 1997:80)

Analisa *forward pass* dimulai dari aktivitas yang tidak memiliki predesesor dimana waktu mulainya ditentukan terlebih dahulu dan waktu selesai aktivitas didapat dari penambahan waktu mulai dengan durasi aktivitas tersebut. Pada aktivitas yang memiliki predesesor, waktu mulai aktivitas tersebut akan mengikuti nilai dari waktu selesai aktivitas predesesornya dan waktu selesai didapat dari penambahan waktu mulai dengan durasi aktivitas. Jika aktivitas memiliki lebih dari satu predesesor, maka waktu mulai aktivitas tersebut akan mengikuti nilai maksimal dari semua waktu selesai aktivitas predesesornya dan waktu selesai didapat dari penambahan waktu mulai dengan durasi aktivitas. Proses ini dilakukan hingga semua aktivitas memiliki waktu mulai dan waktu selesai yang dinamakan waktu mulai awal dan waktu selesai awal.

Analisa *backward pass* adalah kebalikan dari analisa *forward pass*. Analisa ini dimulai dari aktivitas yang tidak memiliki suksesor. Waktu selesai didapat dari waktu selesai penghitungan secara *forward pass* dan waktu mulai dari aktivitas tersebut didapat dari pengurangan waktu selesai dengan durasi aktivitas. Pada aktivitas yang memiliki suksesor, waktu selesai aktivitas tersebut didapat dari waktu mulai aktivitas suksesornya dan waktu mulai aktivitas tersebut didapat dari pengurangan antara waktu selesai dengan durasi aktivitas. Pada aktivitas yang

memiliki lebih dari satu suksesor, waktu selesai aktivitas tersebut didapat dari nilai minimal dari semua waktu mulai aktivitas suksesornya dan waktu mulai aktivitas tersebut didapat dari pengurangan antara waktu selesai dengan durasi aktivitas. Proses ini dilakukan hingga semua aktivitas memiliki waktu mulai dan waktu selesai yang dinamakan waktu mulai terlambat dan waktu selesai terlambat.

Dari hasil *forward pass* dan *backward pass*, maka akan dicari selisihnya (*slack*). Selisih dari masing-masing aktivitas dihitung dan jika aktivitas itu memiliki *slack nol*, maka aktivitas itu berada pada jalur kritis. Namun aktivitas yang tidak memiliki *slack nol* bukanlah aktivitas yang berada pada jalur kritis. Dari hasil tersebut, manajer atau pimpinan proyek dapat melakukan kontrol yang lebih baik lagi dalam pengerjaan proyek tersebut.

Sebagai contoh, pada tabel 2.5, terdapat 9 kegiatan pada pembuatan proyek sistem. Masing-masing kegiatan memiliki pendahulu dan waktu kerja yang dihitung dalam hari. Proses dimulai pada kegiatan survey dan selesai pada kegiatan siap untuk diprogram.

Tabel 2.5 Tabel Kegiatan Dalam Proyek Sistem

Kode	Kegiatan	Pendahulu	Waktu (Hari)
A	Survey	-	4
B	Studi kelayakan	A	20
C	Analisis sistem yang ada	A	10
D	Membangun sistem	B,C	5
E	Menyiapkan spec file	D	10
F	Menyiapkan kebutuhan I/O	D	7
G	Spesifikasi program	E,F	12
H	Membuat prosedur manual	F	10
I	Siap untuk deprogram	G,H	1

Selanjutnya, dilakukan proses *forward pass* untuk menghitung tanggal awal pada masing-masing kegiatan. Tanggal mulai kegiatan yang tidak memiliki

predesesor akan mengikuti tanggal mulai proyek, tanggal mulai kegiatan yang memiliki predesesor akan mengikuti tanggal selesai predesesor, sedangkan tanggal mulai kegiatan yang memiliki lebih dari satu predesesor akan mengikuti nilai maksimum tanggal selesai dari predesesornya. Tanggal selesai dari masing-masing kegiatan didapat dari hasil penjumlahan tanggal mulai dengan durasi sehingga hasil dari proses *forward pass* dapat terlihat pada tabel 2.6.

Tabel 2.6 Tabel Hasil Proses *Forward Pass*

Kode	Kegiatan	Tanggal Mulai	Tanggal Selesai
A	Survey	1	4
B	Studi kelayakan	5	24
C	Analisis sistem yang ada	5	14
D	Membangun sistem	25	29
E	Menyiapkan spec file	30	39
F	Menyiapkan kebutuhan I/O	30	36
G	Spesifikasi program	40	51
H	Membuat prosedur manual	37	46
I	Siap untuk diprogram	52	53

Hasil dari proses *forward pass* kemudian digunakan sebagai dasar pada proses *backward pass*. Proses *backward pass* digunakan untuk mencari tanggal terlambat dari masing-masing kegiatan. Tanggal selesai kegiatan yang tidak memiliki suksesor akan mengikuti tanggal selesai proses *forward pass*, tanggal selesai kegiatan yang memiliki suksesor akan mengikuti tanggal mulai dari suksesornya, sedangkan tanggal selesai kegiatan yang memiliki lebih dari satu suksesor akan mengikuti nilai minimum tanggal mulai dari suksesornya. Tanggal mulai dari masing-masing kegiatan didapat dari hasil pengurangan tanggal selesai dengan durasi sehingga hasil dari proses *backward pass* dapat terlihat pada tabel 2.7.

Tabel 2.7 Tabel Hasil Proses *Backward Pass*

Kode	Kegiatan	Tanggal Mulai	Tanggal Selesai
A	Survey	1	4
B	Studi kelayakan	5	24
C	Analisis sistem yang ada	20	24
D	Membangun sistem	25	29
E	Menyiapkan spec file	30	39
F	Menyiapkan kebutuhan I/O	33	39
G	Spesifikasi program	40	51
H	Membuat prosedur manual	42	51
I	Siap untuk diprogram	52	53

Setelah didapat hasil dari kedua proses, langkah terakhir adalah mencari *slack* (selisih) dari proses *forward pass* dan *backward pass*. *Slack* dari proses *forward pass* dan *backward pass* dapat dilihat pada tabel 2.8.

Tabel 2.8 Tabel *Slack* Dari Proses *Forward Pass* dan *Backward Pass*

Kode	Kegiatan	Slack (Hari)
A	Survey	0
B	Studi kelayakan	0
C	Analisis sistem yang ada	15
D	Membangun sistem	0
E	Menyiapkan spec file	0
F	Menyiapkan kebutuhan I/O	3
G	Spesifikasi program	0
H	Membuat prosedur manual	5
I	Siap untuk diprogram	0

Sesuai dengan teori metode *Critical Path*, hasil dari tabel 2.8 dapat memperlihatkan kegiatan-kegiatan yang terletak pada waktu kritis yaitu kegiatan dengan *slack* nol. Jadi kegiatan yang berada pada jalur kritis adalah kegiatan survey, studi kelayakan, membangun sistem, menyiapkan spec file, spesifikasi program, dan siap untuk diprogram. Dengan mengetahui kegiatan yang terletak pada waktu kritis, manajer atau project leader dapat lebih memberi perhatian kepada kegiatan-kegiatan tersebut.

2.8 Database

Menurut Marlinda, *database* adalah suatu susunan/kumpulan data operasional lengkap dari suatu organisasi/perusahaan yang diorganisir/dikelola dan disimpan secara terintegrasi dengan menggunakan metode tertentu menggunakan komputer sehingga mampu menyediakan informasi optimal yang diperlukan pemakainya (Marlinda, 2004:1).

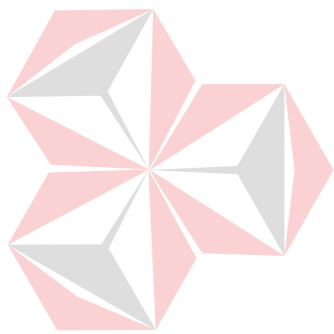
Penyusunan satu *database* digunakan untuk mengatasi masalah-masalah pada penyusunan data yaitu redundansi dan inkonsistensi data, kesulitan pengaksesan data, isolasi data untuk standarisasi, *multiple user* (banyak pemakai), masalah keamanan (*security*), masalah integrasi (kesatuan), dan masalah data *independence* (kebebasan data).

2.9 Testing

Menurut Romeo, *testing software* adalah proses mengoperasikan *software* dalam suatu kondisi yang dikendalikan, untuk verifikasi apakah telah berlaku sebagaimana telah ditetapkan (menurut spesifikasi), mendeteksi *error*, dan validasi apakah spesifikasi yang telah ditetapkan sudah memenuhi keinginan atau kebutuhan dari pengguna yang sebenarnya.

Verifikasi adalah pengecekan atau pengetesan entitas-entitas, termasuk *software*, untuk pemenuhan dan konsistensi dengan melakukan evaluasi hasil terhadap kebutuhan yang telah ditetapkan (*Are we building the system right ?*). Validasi melihat kebenaran sistem, apakah proses yang telah ditulis dalam spesifikasi adalah apa yang sebenarnya diinginkan atau dibutuhkan oleh pengguna (*Are we building the right system?*).

Testing seharusnya berorientasi untuk membuat kesalahan secara intensif, untuk menentukan apakah suatu hal tersebut terjadi bilamana tidak seharusnya terjadi atau suatu hal tersebut tidak terjadi dimana seharusnya mereka ada (Romeo, 2003:3).



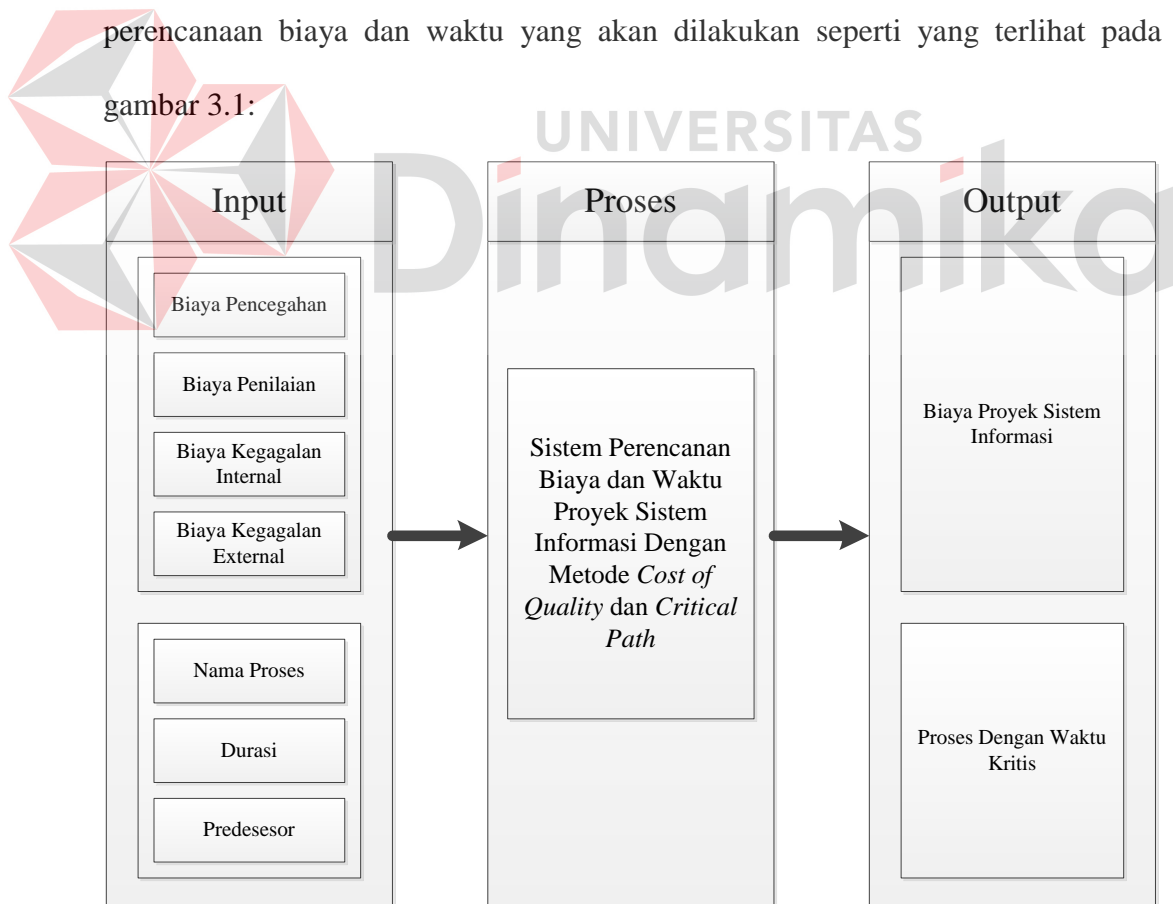
UNIVERSITAS
Dinamika

BAB III

METODE PENELITIAN DAN PERANCANGAN SISTEM

3.1 Metode Penelitian

Model yang digunakan pada rancang bangun aplikasi untuk perencanaan biaya dan waktu proyek sistem informasi ini menggunakan metode *Cost of Quality* dan *Critical Path*. Metode *Cost of Quality* dipilih sebab penghitungan biaya ditetapkan berfokus pada kualitas produk yang dihasilkan. Sedangkan metode *Critical Path* dipilih sebab mampu menganalisa titik yang harus diprioritaskan dalam pembuatan proyek. Berikut adalah diagram blok proses perencanaan biaya dan waktu yang akan dilakukan seperti yang terlihat pada gambar 3.1:



Gambar 3.1 Diagram Blok Proses Perencanaan Biaya dan Waktu

Tahap pertama adalah mengumpulkan input-input yang akan digunakan dalam sistem. Dalam metode *Cost of Quality*, terdapat 4 input, yaitu biaya pencegahan, biaya penilaian, biaya kegagalan internal dan biaya kegagalan eksternal. Sedangkan metode *Critical Path* memiliki 3 input yaitu nama proses, durasi dan predesesor/pendahulu.

Tahap kedua adalah proses yang akan dikerjakan pada sistem informasi tersebut yaitu, sistem manajemen biaya dan waktu proyek sistem informasi dengan metode *Cost of Quality* dan *Critical Path*. Tahap ketiga adalah *output* yang didapat dalam sistem ini yaitu biaya proyek sistem informasi dan proses dengan waktu kritis.

Adapun langkah-langkah awal yang akan dilakukan dalam pembuatan aplikasi untuk perencanaan biaya dan waktu yaitu sebagai berikut:

a. Wawancara

Cara ini dilakukan untuk mengetahui permasalahan-permasalahan yang terjadi di bagian SSI STIKOM Surabaya berkaitan dengan pembuatan sistem informasi, pengelolaan sumber daya, penentuan biaya dan waktu proses, dan pengawasan yang dilakukan selama pembuatan sistem informasi. Selain itu, cara ini dilakukan guna memperoleh kebutuhan-kebutuhan sistem dan keinginan pegawai SSI dalam pembuatan aplikasi nantinya.

b. Studi literatur

Langkah ini dilakukan dengan tujuan untuk memperoleh pengetahuan yang lebih dalam mengenai manajemen proyek sistem informasi. Diantaranya adalah manajemen biaya dan manajemen waktu. Metode yang dipelajari antara lain *Cost of Quality* dan *Critical Path*. *Cost of Quality* dipilih karena

mampu mendeskripsikan biaya menurut golongannya masing-masing dan *Critical Path* karena paling umum digunakan dalam hal manajemen waktu.

Setelah melakukan pengumpulan data, kegiatan selanjutnya yang dilakukan adalah melakukan analisis sistem, desain sistem, pembuatan program (*coding*), *testing*, dan implementasi. Dalam analisis sistem, hal-hal yang dilakukan antara lain: membuat bagan alir metode, *system flow* dan *Data Flow Diagram* (DFD). Untuk perancangan sistem, hal-hal yang dilakukan yaitu membuat *Entity Relationship Diagram* (ERD) serta desain *input output*. Untuk *coding*, yang dilakukan adalah memastikan *Microsoft Visual Studio .NET* yang digunakan sebagai bahasa pemrograman serta *Microsoft SQL Express* sebagai basis data berjalan dengan baik dan lancar untuk pembuatan program. Dalam tahap *testing* dan implementasi, yang dilakukan adalah melakukan *deployment* terhadap program yang telah dibuat dan melakukan uji coba kepada pemakai program.

3.2 Identifikasi Masalah

Dalam membuat proyek, perencanaan biaya sangat diperlukan. Salah satunya adalah penentuan harga. Biaya yang tepat membuat pihak perusahaan dan pelanggan akan merasa puas. Meski begitu, menentukan biaya haruslah dilandaskan pada suatu metode tertentu agar hasil perhitungan dapat dipertanggung jawabkan.

Selain merencanakan biaya, masalah lain yang sering dimiliki dalam membuat proyek adalah merencanakan waktu. Merancang jadwal adalah salah satu perencanaan waktu yang dapat dibuat. Dengan begitu banyak aktivitas dan masing-masing aktivitas memiliki syarat yang harus ditempuh, akan menciptakan celah waktu di antara banyaknya proses. Jika tidak dilakukan perancangan jadwal

yang tepat, dikhawatirkan akan terjadi kemoloran pada jadwal kerja semua aktivitas. Dalam hal ini, menentukan jalur kritis dimana merupakan jalur terpenting dalam pembuatan proses karena pada aktivitas pada jalur ini tidak melebihi batas waktu yang ditentukan.

Berdasarkan permasalahan tersebut, dibutuhkan sistem untuk perencanaan biaya dan waktu pada setiap pembuatan sistem informasi. Aplikasi ini akan menampilkan informasi biaya proyek dan jalur kritis aktivitas dalam sebuah proses pembuatan sistem informasi tersebut. Diharapkan dengan adanya aplikasi ini, pengerjaan proyek sistem informasi dapat berjalan dengan baik.

3.3 Analisa Kebutuhan

Dengan adanya aplikasi untuk perencanaan biaya dan waktu proyek sistem informasi, diharapkan dapat membantu manajer dalam menjalankan proyek. Selain itu, aplikasi ini diharapkan dapat memberikan kepercayaan kepada konsumen akan biaya yang diestimasikan.

Metode *Cost of Quality* dan *Critical Path* yang digunakan dalam aplikasi untuk perencanaan biaya dan waktu proyek sistem informasi diharapkan dapat membantu dalam proses perhitungan estimasi biaya dan perancangan jadwal yang dibuat.

Setelah dilakukan wawancara pada pihak SSI, dapat diambil beberapa masukan dalam sistem yaitu:

1. SSI memiliki 4 tipe pengguna yaitu *administrator*, manajer, *market*, dan *project resource*. *Administrator*, manajer dan *market* adalah pemegang kepentingan eksternal proyek sedangkan *project resource* adalah

pemegang kepentingan internal proyek. *Administrator* bertugas menangani *master user*, manajer menjadi pemimpin bagian yang bertugas melakukan persetujuan akan project, membuat biaya, dan menutup *project*. Manajer juga mendapat hasil laporan dari project. *Market* bertugas menangani *master customer* dan juga membuat *project* baru. *Project resource* bertugas mengerjakan *project*. *Market* berubungan dengan konsumen.

2. Project Resource dapat dibagi lagi menjadi 4 tipe yakni *project leader*, *project team*, *project tester*, dan *project trainer*. *Project leader* bertugas memimpin *project*, membuat jadwal dan sumber daya dari *project* dan menyetujui pekerjaan *project team*, *project tester*, dan *project trainer*..

Penetapan *project leader* ditentukan oleh manajer. *Project team* bertugas mengerjakan program, desain, dan semua yang menjadi pekerjaan *project*.

Project tester bertugas mencoba hasil dari pekerjaan *project team* dan *project trainer* bertugas melatih pelanggan hasil dari aplikasi yang dibuat oleh *project team* dan dicoba oleh *project train*.

3. Level pada masing-masing project ada 3 tahap yakni *project*, *task* dan *sub task* (aktivitas).
4. Sistem *framework* dari manajemen proyek sistem informasi dari PMI yang digunakan oleh SSI adalah manajemen biaya proyek dan manajemen waktu proyek meliputi pembuatan estimasi biaya, mendefinisikan aktivitas, mengurutkan aktivitas, mengestimasi sumber daya (manusia maupun waktu) aktivitas, dan mengembangkan jadwal.
5. Pada SSI, Cost of Quality yang digunakan adalah sebagai berikut:

Tabel 3.1 Tabel Parameter *Cost of Quality* yang digunakan SSI

Biaya Pencegahan	Biaya Penilaian
<ul style="list-style-type: none"> • Pelatihan Internal • Analisis Kebutuhan • Prototipe • Penyesuaian Desain • Programming • Analisa Kebutuhan • Spesifikasi Software • Dokumentasi Internal • Evaluasi Pekerjaan • Kesalahan Coding • Kesalahan Desain • Investasi • Pelatihan Kualitas • Operasional • Pelatihan Pada Konsumen 	<ul style="list-style-type: none"> • Pengecekan Desain • Pengecekan Coding • Black box testing • Beta testing • Ujicoba kegunaan • Ujicoba Dari Pelanggan • Acceptance testing • Pengecekan Staff
Biaya Kegagalan Eksternal	Biaya Kegagalan Eksternal
<ul style="list-style-type: none"> • Pembetulan Bug • Pembuangan Waktu • Pengerjaan Kembali 	<ul style="list-style-type: none"> • Pemanggilan Teknisi • Upgrade Produk • Garansi • Komplain Diluar Garansi • Kehilangan Reputasi

Untuk menjalankan aplikasi untuk perencanaan biaya dan waktu proyek sistem informasi, diperlukan dukungan *software* dan *hardware* sebagai berikut :

Software:

1. Pada *Server*:

- a. Sistem operasi menggunakan *Windows XP Professional Service Pack* tiga atau lebih.
- b. *Database* untuk pengolahan data menggunakan *Microsoft SQL Server 2005* dengan *SQL Browser* yang berjalan dengan baik.
- c. *Microsoft .NET Framework 2.0* atau di atasnya.

2. Pada *Client*:

- a. Sistem operasi menggunakan *Windows XP Professional Service Pack* tiga atau lebih.
- b. *Database* untuk pengolahan data menggunakan *Microsoft SQL Server 2005* dengan *SQL Browser* yang berjalan dengan baik.\

- c. *Microsoft .NET Framework 2.0* atau di atasnya.
- d. *DevExpress 9*.

Hardware

1. Pada *Server*:

- a. Processor Intel Pentium IV dengan kecepatan 1.5 GHz atau lebih.
- b. *Memory* 512 Mb atau lebih.
- c. *Harddisk* 10 Gb atau lebih.
- d. Koneksi jaringan yang bisa menghubungkan komputer *server* dengan komputer *client*.
- e. Monitor, mouse, keyboard yang berjalan dengan baik.

2. Pada *Client*:

- a. Processor Intel Pentium IV dengan kecepatan 1 GHz atau lebih.
- b. *Memory* 1 Gb atau lebih.
- c. *Harddisk* 15 Gb atau lebih.
- d. Koneksi jaringan yang bisa menghubungkan komputer *client* dengan komputer *server*.
- e. Monitor dengan resolusi 1280 x 800.
- f. Mouse dan keyboard yang berjalan dengan baik.

3.3 Analisa dan Perancangan Sistem

Langkah ini dilakukan untuk membuat analisa dan rancangan dari sistem yang akan dibuat. Rancangan ini dibuat dalam bentuk diagram aliran data, yang digambarkan dengan menggunakan *Power Designer 6.0* dan *Microsoft Office Visio 2010*. Diagram aliran data ini menggambarkan arus data yang ada dalam

aplikasi secara terstruktur dan jelas, serta menggambarkan proses-proses yang terjadi pada sistem sehingga dapat menjadi sarana dokumentasi sistem yang baik.

Sistem yang akan dirancang ini bertujuan membantu manajer dalam menentukan biaya proyek berdasarkan parameter-parameter yang terdapat dalam metode *Cost of Quality*. Biaya-biaya tersebut akan ditotal untuk menghasilkan biaya total. Dari outputnya, pihak *market* akan memberikan hasil biaya kepada pelanggan.

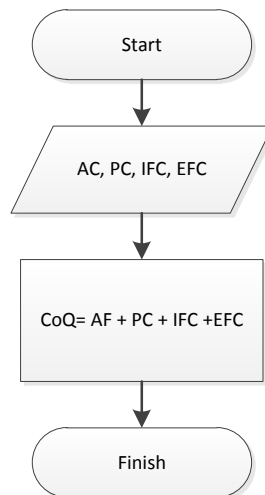
Selain itu, sistem juga dapat menampilkan informasi aktivitas mana yang terletak pada jalur kritis. Dari *output* yang dihasilkan, pemimpin proyek akan lebih berfokus terhadap aktivitas-aktivitas tersebut sehingga proyek akan selesai tepat waktu.

Sistem ini juga dapat menampilkan laporan tentang kondisi proyek tertentu dan juga laporan mengenai kondisi masing-masing anggota tim SSI. Dengan *output* yang dihasilkan, manajer akan mengetahui kondisi proyek dan kondisi masing-masing anggota timnya.

3.3.1 Flowchart

A. Flowchart metode *Cost of Quality*

Terdapat 4 masukan dalam metode *Cost of Quality*, yaitu AC (*Appraisal Cost*) atau biaya penilaian, PC (*Prevention Cost*) atau biaya pencegahan, IFC (*Internal Failure Cost*) atau biaya kegagalan internal, dan EFC (*External Failure Cost*) atau biaya kegagalan eksternal. Masing-masing parameter dari masing-masing inputan kemudian dijumlah dan menjadi biaya estimasi dengan metode *Cost of Quality*. Untuk lebih jelas dapat dilihat pada gambar 3.2.

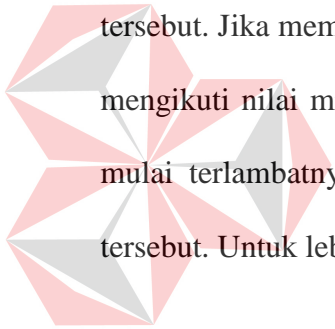


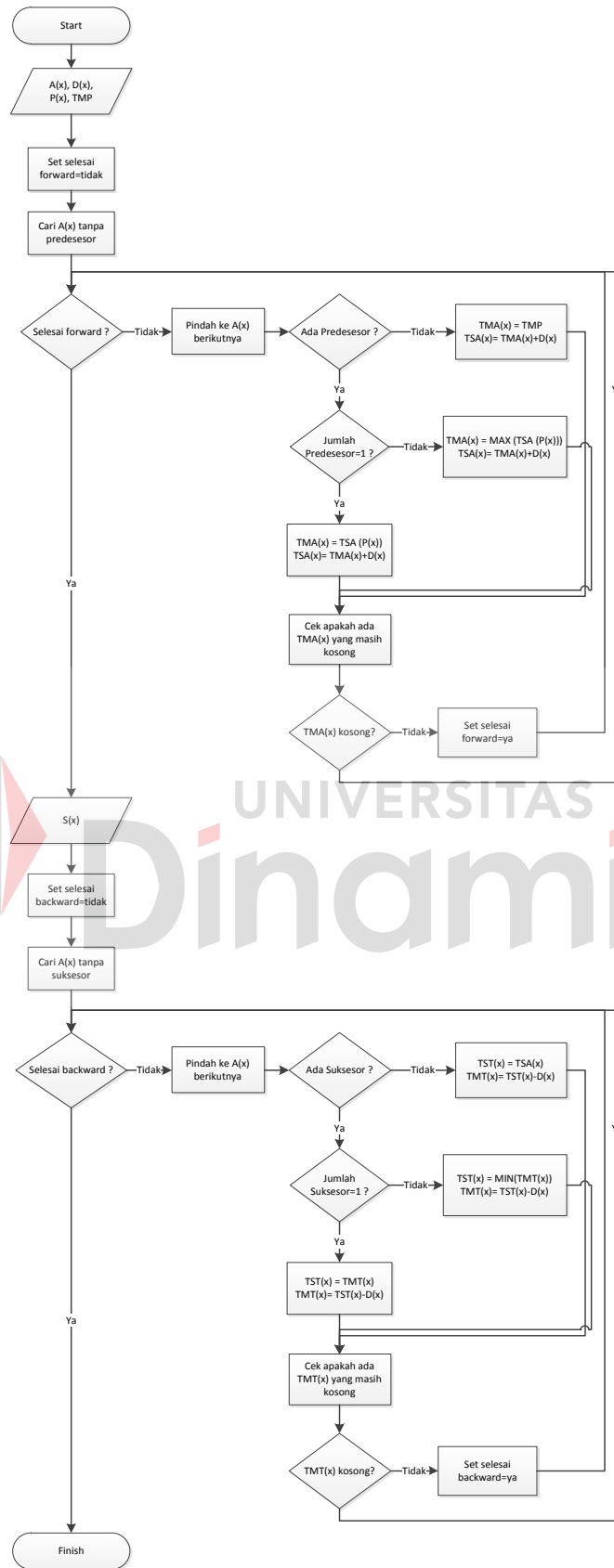
Gambar 3.2 Flowchart metode Cost of Quality

B. Flowchart metode Critical Path

Dalam menjalankan metode *Critical Path* dibutuhkan 2 langkah yaitu melakukan *forward pass* dan *backward pass*. Tahap *forward pass* membutuhkan masukan yaitu aktivitas, durasi, predesesor dan tanggal mulai proyek. Langkah berikutnya adalah memulai perulangan hingga semua nilai tanggal mulai awal terisi semua. Didalam perulangan, dilakukan percabangan, apakah aktivitas itu memiliki predesesor, jika tidak ada, maka tanggal mulai awalnya adalah tanggal mulai proyek dan tanggal selesai awalnya adalah tanggal mulai awal ditambah durasi aktivitas tersebut. Jika aktivitas itu memiliki predesesor, maka dilakukan percabangan lagi apakah predesesornya lebih dari satu. Jika tidak lebih dari satu predesesor, maka tanggal mulai awalnya adalah tanggal selesai awal predesesornya dan tanggal selesai awalnya adalah tanggal mulai awal ditambah durasi aktivitas tersebut. Jika memiliki lebih dari satu predesesor, maka tanggal mulai awalnya mengikuti nilai maksimum dari tanggal selesai awal predesesornya, dan tanggal selesai awalnya adalah tanggal mulai awal ditambah durasi aktivitas tersebut.

Tahap berikutnya adalah tahap *backward pass*. Tahap ini membutuhkan masukan lagi yaitu suksesor. Langkah berikutnya adalah memulai perulangan hingga semua nilai tanggal mulai terlambat terisi semua. Didalam perulangan, dilakukan percabangan, apakah aktivitas itu memiliki suksesor, jika tidak ada, maka tanggal selesai terlambatnya adalah tanggal selesai awal dan tanggal mulai terlambatnya adalah tanggal selesai terlambat dikurangi durasi aktivitas tersebut. Jika aktivitas itu memiliki suksesor, maka dilakukan percabangan lagi apakah suksesornya lebih dari satu. Jika tidak lebih dari satu suksesor, maka tanggal selesai terlambatnya adalah tanggal mulai terlambat suksesornya dan tanggal mulai terlambatnya adalah tanggal selesai terlambat dikurangi durasi aktivitas tersebut. Jika memiliki lebih dari satu suksesor, maka tanggal selesai terlambatnya mengikuti nilai minimum dari tanggal mulai terlambat suksesornya, dan tanggal mulai terlambatnya adalah tanggal selesai terlambat dikurangi durasi aktivitas tersebut. Untuk lebih jelas dapat dilihat pada gambar 3.3.





Gambar 3.3 Flowchart metode Critical Path

Keterangan:

$A(x)$ = Aktivitas x

$D(x)$ = Durasi aktivitas x

$P(x)$ = Predesesor aktivitas x

TMP= Tanggal mulai proyek

$TMA(x)$ = Tanggal mulai awal aktivitas x

$TSA(x)$ = Tanggal selesai awal aktivitas x

$S(x)$ = Suksesor aktivitas x

$TMT(x)$ = Tanggal mulai terlambat aktivitas x

$TST(x)$ = Tanggal selesai terlambat aktivitas x

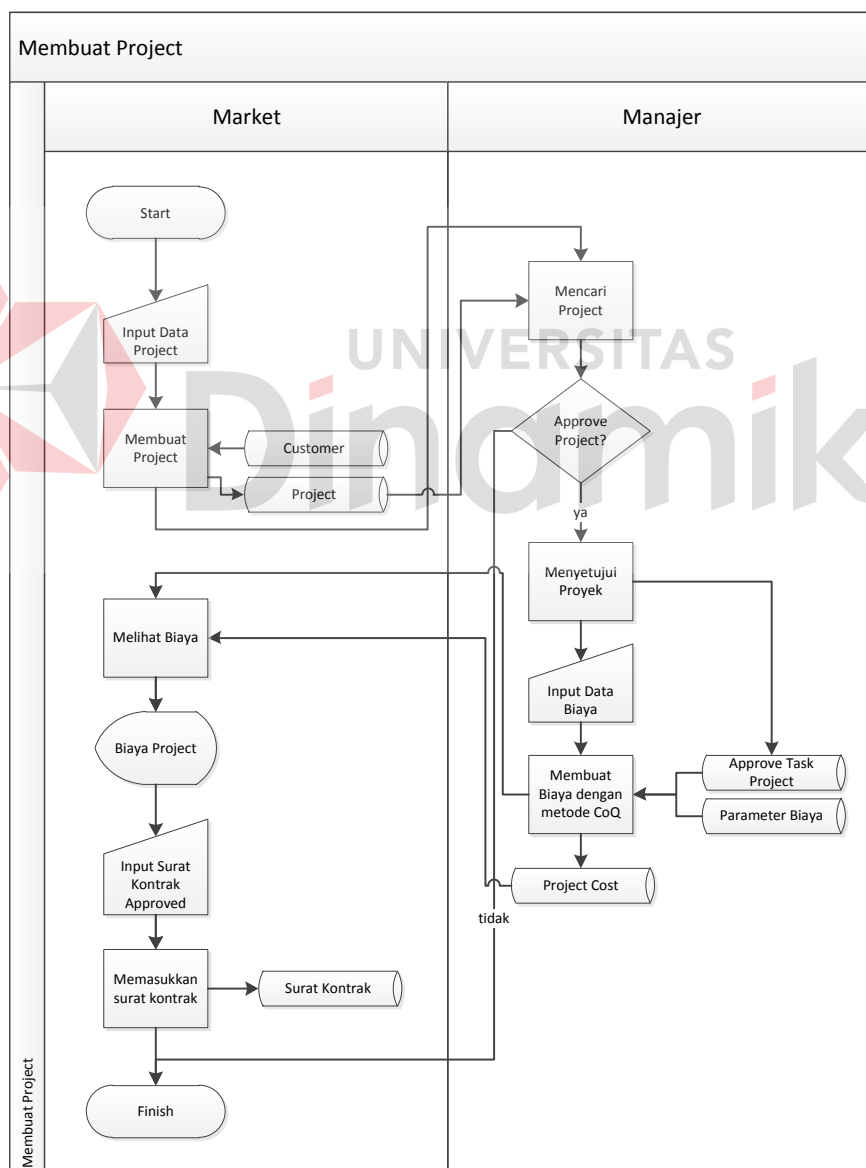
3.3.2 *System flow*

System flow adalah suatu bagian yang menunjukkan arus pekerjaan secara menyeluruh dari suatu sistem dimana bagian ini menjelaskan urutan prosedur-prosedur yang ada di dalam sistem. *System flow* yang dibuat akan terbagi menjadi empat bagian yang merupakan proses utama dalam sistem ini, yaitu membuat proyek, merancang jadwal kerja proyek, menjalankan jadwal kerja proyek dan penutupan proyek.

A. *System flow* membuat proyek

Bagan alir program untuk membuat proyek dimulai dengan memasukkan data proyek dari *customer* ke dalam sistem dan disimpan. Data yang ada kemudian dilihat oleh manajer untuk kemudian disetujui. Setelah disetujui, manajer dapat membuat rincian biaya menurut parameter biaya.

Biaya yang diinputkan memiliki empat golongan: biaya pencegahan, biaya penilaian, biaya kegagalan internal dan biaya kegagalan eksternal. Masing-masing golongan memiliki parameter. Manajer akan memilih jenis parameter dan sumber daya (pekerja dan waktu). Hasil biaya akan dimunculkan pada bagian *market* dan akan dibuatkan surat kontrak. Begitu surat kontrak tersebut disetujui oleh *customer* maka *market* akan menginputkan surat kontrak tersebut dalam sistem. Bagan alir program untuk membuat proyek dapat dilihat pada gambar 3.4.



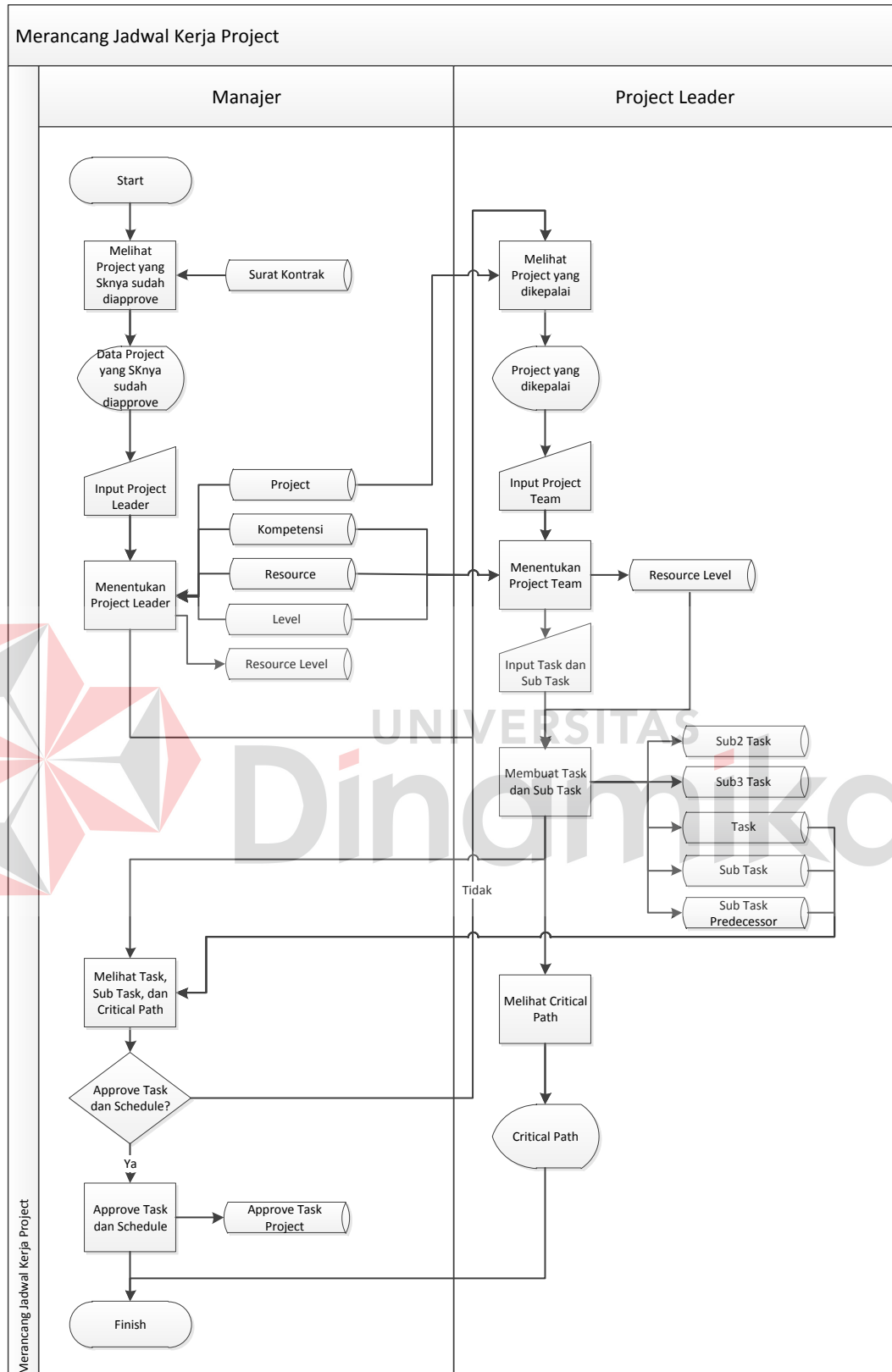
Gambar 3.4 Bagan Alir Program untuk Membuat Proyek

B. *System flow* merancang jadwal kerja proyek

Bagan alir program untuk merancang jadwal kerja proyek dimulai dari sisi manajer dengan melihat daftar proyek yang dapat dikerjakan (surat kontrak telah disetujui oleh pelanggan). Manajer kemudian akan menentukan siapa pimpinan proyek tersebut.

Setelah pimpinan proyek terpilih, maka pimpinan proyek akan membuat daftar anggota proyek tersebut berdasarkan kemampuan masing-masing dan akan menduduki jabatan *team*, *tester*, dan *trainer*. Pemimpin proyek lalu membuat *task* yang masing-masing akan memiliki *sub task*. *Sub task* harus memiliki *resource* (seorang *project team*, seorang *project tester*, dan seorang *project trainer*), durasi, dan pendahulu (predesesor).

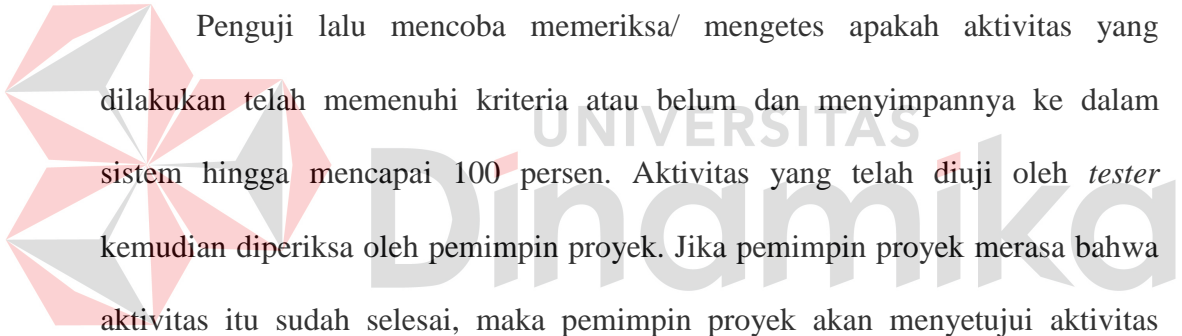
Setelah pemimpin proyek selesai menyusun *task*, *sub task*, dan *resource project*, susunan ini diperiksa oleh manajer. Jika tidak bermasalah, maka manajer akan menyetujui susunan *task*, *sub task* dan *resource project*, namun jika manajer tidak menyetujui susunan *task* dan *sub task*, maka kepala proyek harus membuat susunan yang baru hingga disetujui manajer atau proyek itu dibatalkan. Selain melihat susunan *task* dan *sub task*, manajer maupun pemimpin proyek dapat melihat aktivitas yang berada pada jalur kritis yang dibuat berdasarkan metode *Critical Path*. Bagan alir program untuk merancang jadwal kerja proyek dapat dilihat pada gambar 3.5.



Gambar 3.5 Bagan Alir Program untuk Merancang Jadwal Kerja Proyek

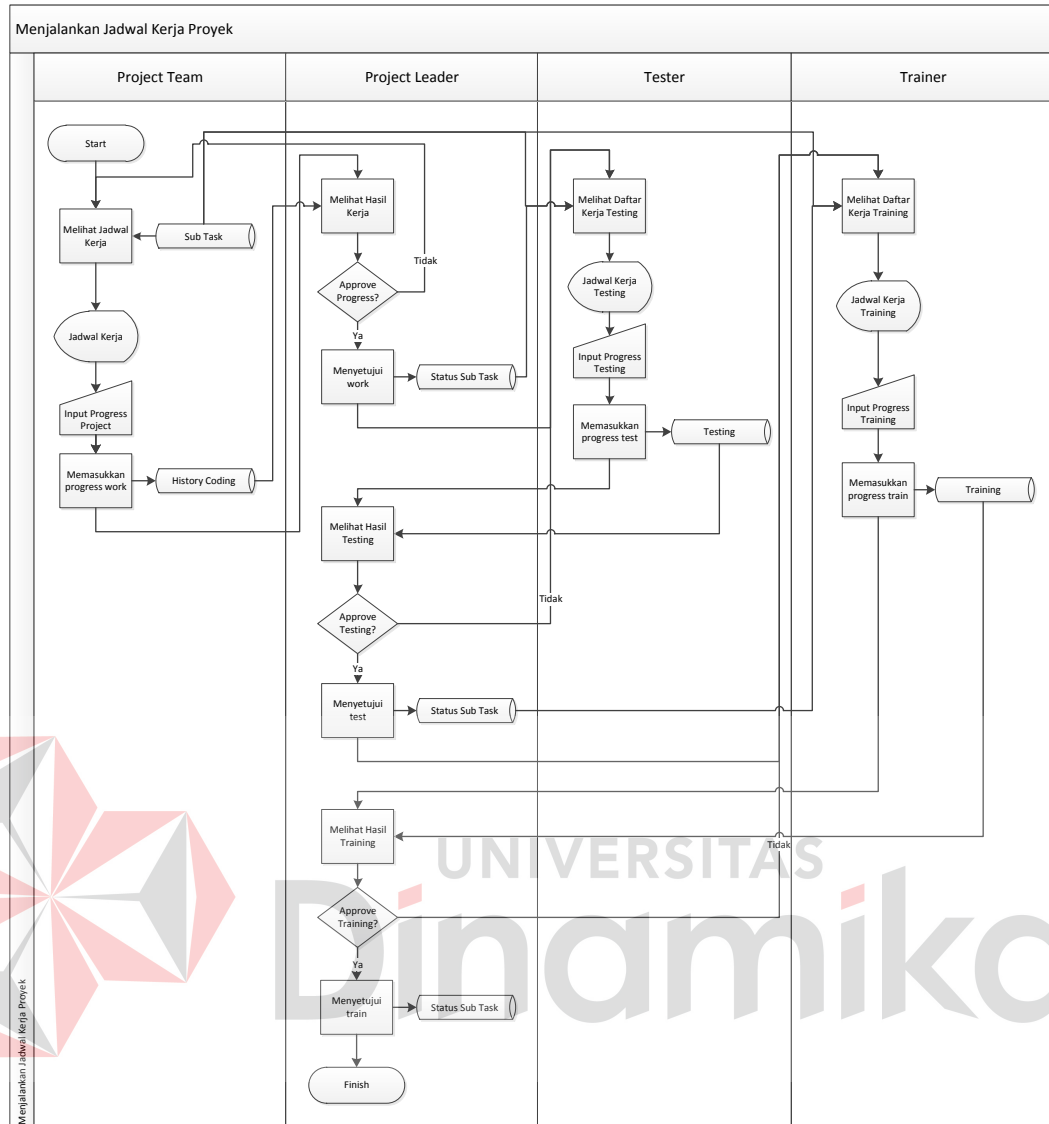
C. *System flow* menjalankan jadwal kerja proyek

Bagan alir program untuk menjalankan jadwal kerja proyek dimulai dengan melihat daftar aktivitas proyek yang dapat dikerjakan oleh anggota proyek (*project team*). Anggota proyek kemudian memasukkan *progress* aktivitas ke dalam sistem hingga mencapai 100 persen atau selesai. Dari sisi pemimpin proyek (*project leader*), aktivitas yang telah selesai itu lalu diperiksa, apakah sudah selesai atau belum. Jika pemimpin proyek merasa bahwa aktivitas itu sudah selesai, maka pemimpin proyek akan menyetujui aktivitas tersebut untuk diuji oleh penguji (*tester*), namun jika tidak, pemimpin proyek akan mengubah hasil kerja *project team*.



Penguji lalu mencoba memeriksa/ mengetes apakah aktivitas yang dilakukan telah memenuhi kriteria atau belum dan menyimpannya ke dalam sistem hingga mencapai 100 persen. Aktivitas yang telah diuji oleh *tester* kemudian diperiksa oleh pemimpin proyek. Jika pemimpin proyek merasa bahwa aktivitas itu sudah selesai, maka pemimpin proyek akan menyetujui aktivitas tersebut diberikan kepada pembeli/konsumen untuk dilatih oleh pelatih (*trainer*), jika tidak, pemimpin proyek akan mengubah hasil kerja *tester*.

Setelah pelatih selesai mengerjakan tugasnya, dia lalu menyimpan hasilnya ke dalam sistem dan akan diperiksa oleh pemimpin proyek hingga pemimpin proyek menyetujui hasil latihan yang diberikan oleh *trainer*. Bagan alir program untuk menjalankan jadwal kerja proyek dapat dilihat pada gambar 3.6.

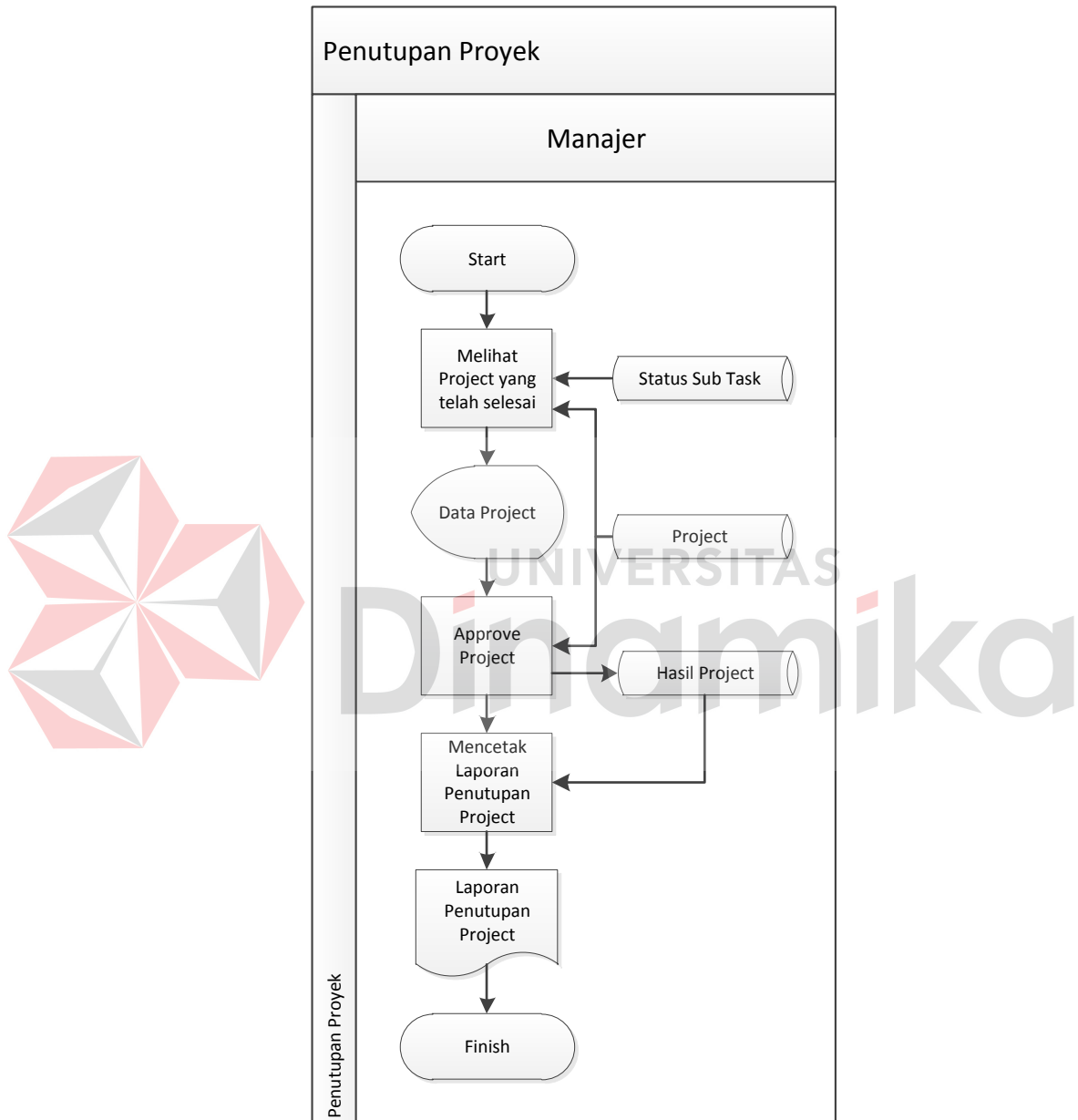


Gambar 3.6 Bagan Alir Program untuk Menjalankan Jadwal Kerja Proyek

D. System flow penutupan proyek

Bagan alir program untuk penutupan proyek dimulai ketika manajer melihat status proyek yang semua aktivitasnya (*sub task*) telah selesai. Manajer kemudian menyatakan proyek itu selesai dan disimpan dalam sistem. Setelah itu, manajer akan mendapatkan laporan penutupan proyek.

Sedangkan dari sisi *market*, proyek selesai itu akan dilihat dan diberitahukan kepada pelanggan. Bagan alir program untuk penutupan proyek dapat dilihat pada gambar 3.7.

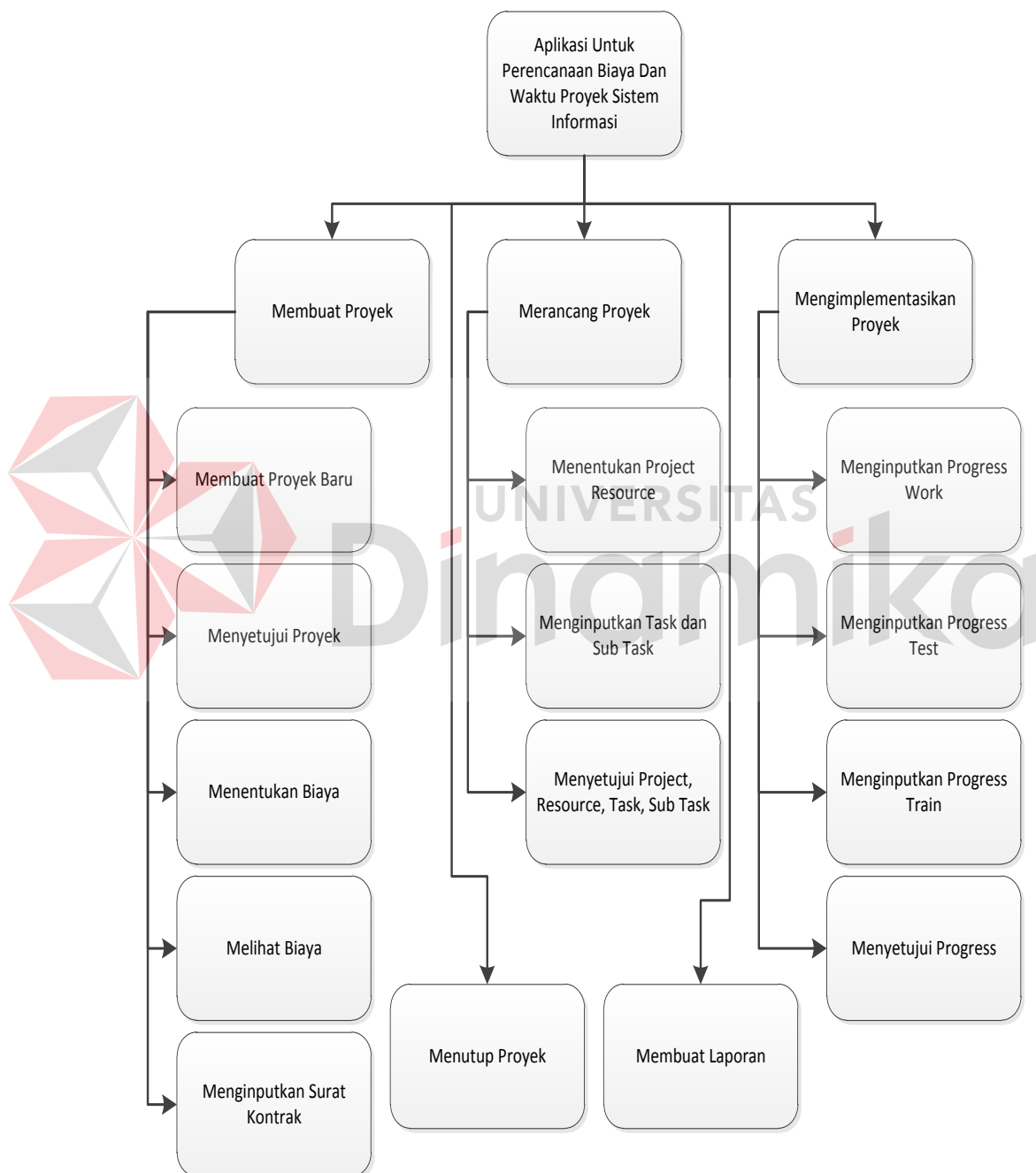


Gambar 3.7 Bagan Alir Program untuk Penutupan Proyek

3.4 Data Flow Diagram (DFD)

3.4.1 Diagram berjenjang

Diagram berjenjang aplikasi untuk perencanaan biaya dan waktu proyek sistem informasi yang akan dibangun ini dapat dilihat pada Gambar 3.8.

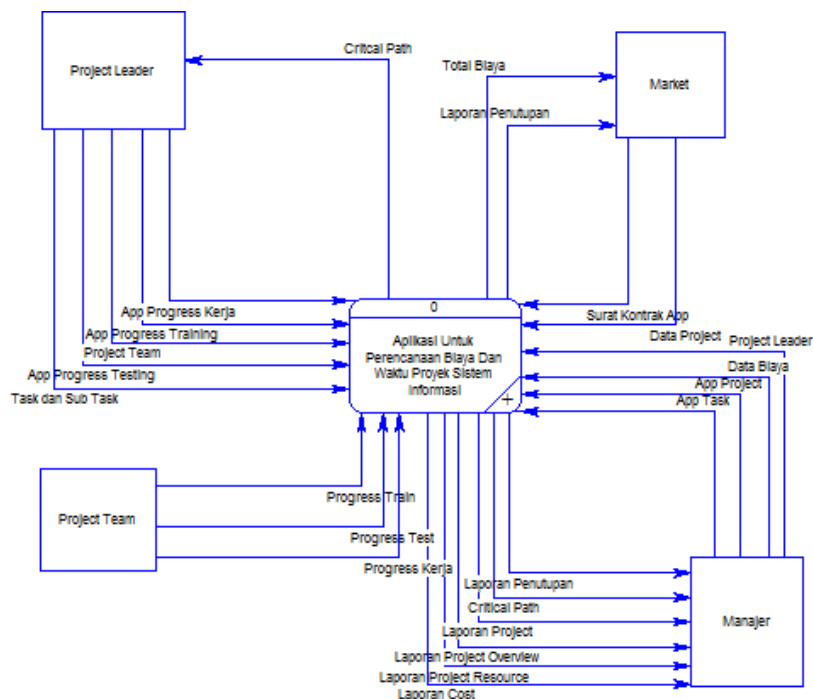


Gambar 3.8 Diagram Berjenjang

3.4.2 Context diagram

Pada *context diagram* Aplikasi Untuk Perencanaan Biaya Dan Waktu Proyek Sistem Informasi ini terdapat empat buah entitas, yaitu manajer, *market*, *project leader*, dan *project team*. Pada sistem ini, manajer memberikan *input* berupa data siapa yang menjadi *project leader*, data biaya proyek tersebut, *approve* terhadap *project*, *approve* terhadap *task*. *Market* memberikan *input* data yang berhubungan dengan pelanggan seperti data proyek yang akan dibuat dan surat kontrak yang telah disetujui oleh pelanggan. *Project leader* memberikan *input* yang berhubungan dengan pekerjaan proyek seperti data *task* dan *sub task*, *project team* yang akan mengerjakan *sub task*, *approve progress* kerja, *approve progress testing*, *approve progress training*. *Project team* akan memberikan data berupa *progress* yang dilakukan seperti *progress* kerja, *progress test*, dan *progress train*.

Dari semua data itu, sistem akan memberikan *output* berupa informasi kepada manajer, *market* dan *project leader*. Manajer akan menerima informasi berupa laporan yaitu laporan *project*, laporan *project overview*, laporan *project resource*, laporan *cost* proyek, laporan penutupan proyek, dan berupa informasi yaitu *critical path* dari sebuah *project*. *Market* akan menerima informasi yang berhubungan dengan pelanggan seperti total biaya proyek yang nantinya akan dibuatkan surat kontrak oleh *market* dan laporan penutupan proyek. *Project leader* akan menerima informasi *critical path* dari sebuah *project*. *Context diagram* aplikasi untuk perencanaan biaya dan waktu proyek sistem informasi dapat dilihat pada gambar 3.9.



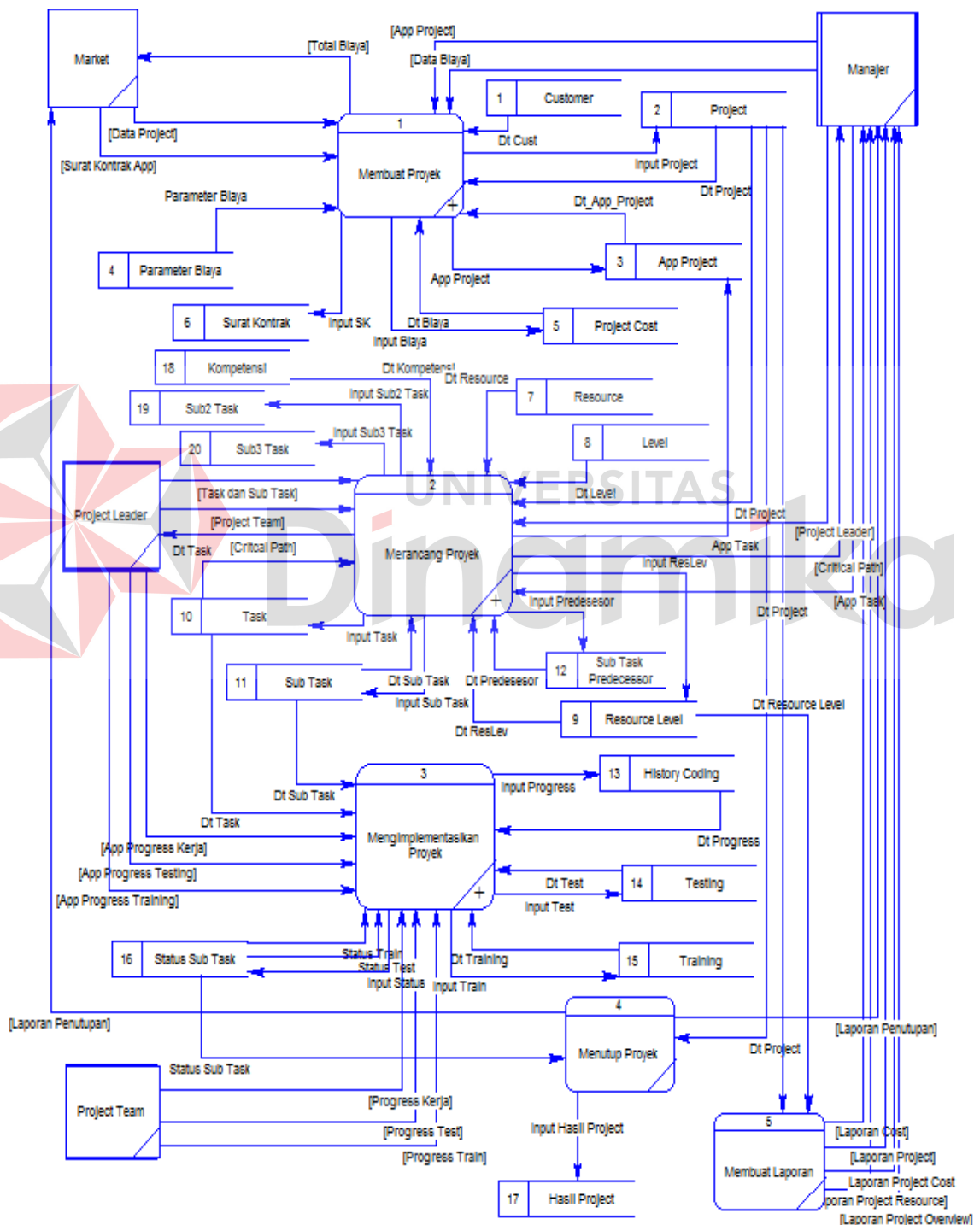
Gambar 3.9 Context Diagram Aplikasi Untuk Perencanaan Biaya Dan Waktu

Proyek Sistem Informasi

3.4.3 DFD level 0 aplikasi untuk perencanaan biaya dan waktu

Aplikasi untuk perencanaan biaya dan waktu akan terbagi dalam 5 subsistem, yaitu membuat proyek, merancang proyek, mengimplementasikan proyek, menutup proyek dan membuat laporan. Subsistem membuat proyek merupakan proses *input* data proyek yang berasal dari *market* dan manajer dan akan menghasilkan *output* biaya. Subsistem berikutnya adalah merancang proyek. Pada subsistem ini, manajer dan *project leader* akan menentukan siapa saja yang akan mengerjakan proyek dan jadwal rencana pengerjaan proyek. Subsistem ini akan menghasilkan *output* berupa *critical path*. Subsistem berikutnya adalah subsistem mengimplementasikan proyek. Pada subsistem ini, *project leader* dan *project team* akan mengisi *progress* dari pengerjaan proyek. Subsistem setelah itu adalah subsistem menutup proyek yang akan memberikan laporan penutupan

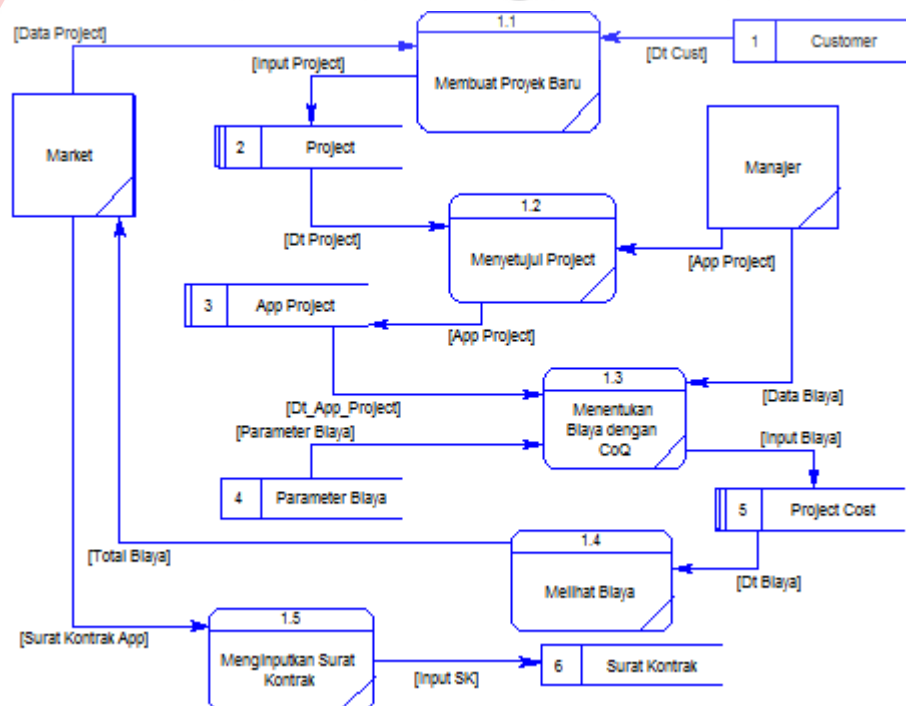
proyek. Subsistem terakhir adalah subsistem laporan yang menghasilkan *output* laporan yaitu laporan *project*, laporan *project overview*, laporan *project resource*, laporan *cost* proyek kepada manajer. Untuk lebih jelasnya, DFD level 0 aplikasi untuk perencanaan biaya dan waktu dapat dilihat pada Gambar 3.10.



Gambar 3.10 DFD Level 0 Aplikasi Untuk Perencanaan Biaya dan Waktu

3.4.4 DFD level 1 subsistem membuat proyek

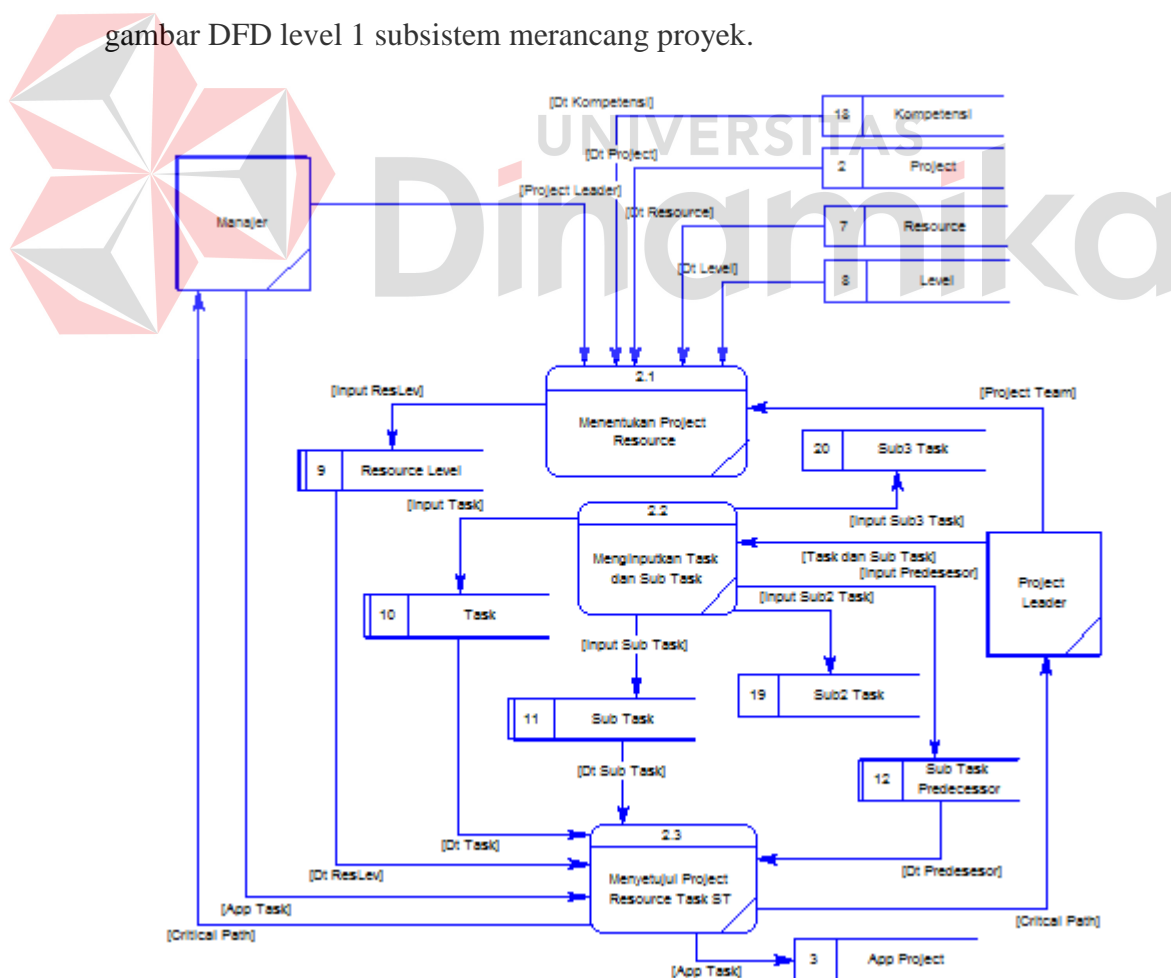
Pada subsistem membuat proyek ini, proses yang pertama dilakukan adalah *input* proyek baru. Data proyek di-*input*-kan oleh *market*, kemudian sistem akan menyimpannya ke dalam tabel proyek. Setelah disimpan, tabel itu akan diakses oleh manajer untuk disetujui pada proses menyetujui *project* dan disimpan pada tabel *app project*. Proses berikutnya adalah menentukan biaya. *Input* dari proses ini berasal dari data-data parameter biaya, data dari *app project*, dan data biaya yang diberikan oleh manajer. Hasil proses ini akan disimpan pada tabel *project cost* yang kemudian akan digunakan oleh proses melihat biaya yang akan dibaca oleh *market*. *Market* kemudian membuat surat kontrak yang akan diberikan kepada pelanggan dan menginputkan surat kontrak yang telah disetujui pada proses menginputkan surat kontrak dan disimpan pada tabel surat kontrak. Untuk lebih jelasnya, Gambar 3.11 adalah gambar DFD level 1 subsistem membuat proyek.



Gambar 3.11 DFD Level 1 Subsistem Membuat Proyek

3.4.5 DFD level 1 subsistem merancang proyek

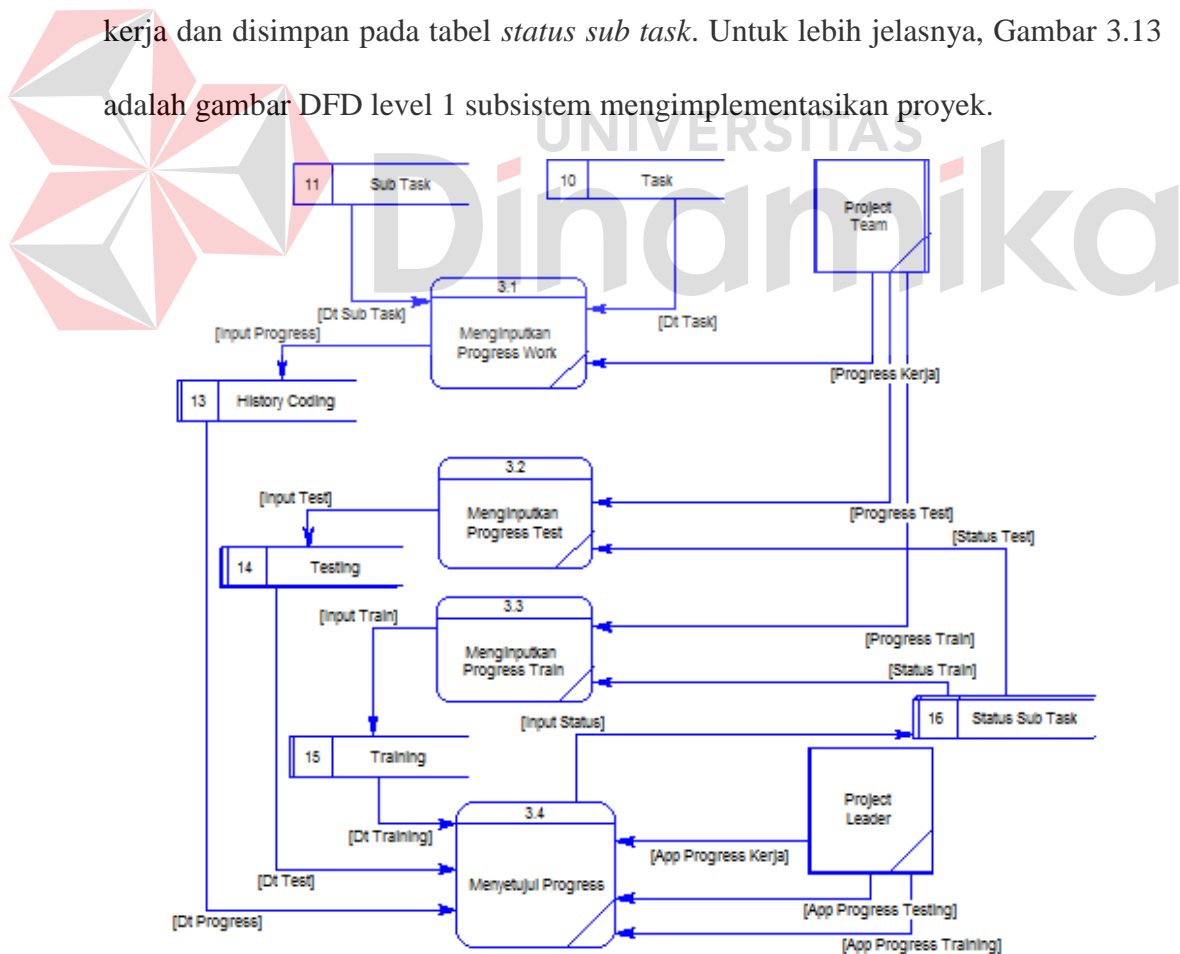
Subsistem merancang proyek ini dimulai dari manajer yang akan memberikan *input* siapa pemimpin dari sebuah *project* pada proses menentukan *project resource*. Pada proses yang sama, *project leader* akan memberikan *input* anggota yang akan mengerjakan *project* yang akan disimpan pada tabel *resource level*. *Project leader* juga akan menentukan *task* dan *sub task* pada proses menginputkan *task* dan *sub task* yang disimpan pada tabel *task*, *sub task* dan *predecessor sub task*. Proses terakhir adalah menyetujui *task*, *sub task* dan *resource* dari *project*. *Input* dari proses ini adalah persetujuan manajer dan disimpan pada tabel *app project*. Untuk lebih jelasnya, Gambar 3.12 adalah gambar DFD level 1 subsistem merancang proyek.



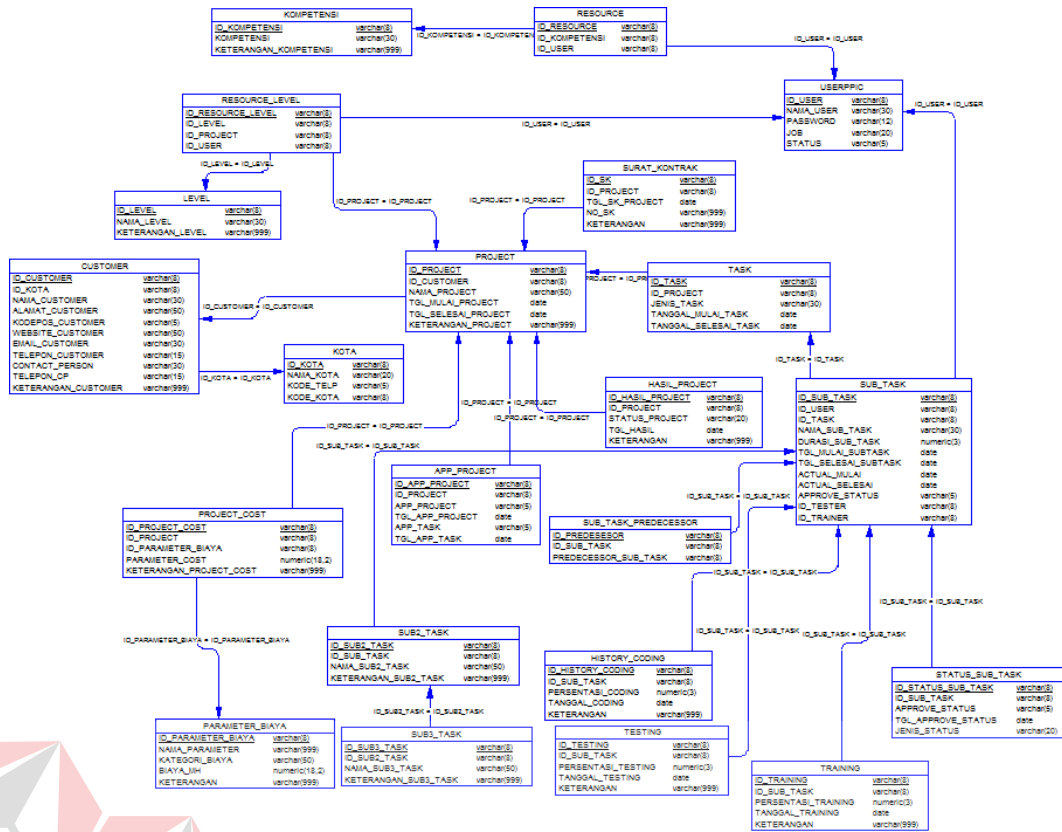
Gambar 3.12 DFD Level 1 Subsistem Merancang Proyek

3.4.6 DFD level 1 subsistem mengimplementasikan proyek

Subsistem mengimplementasikan proyek ini dimulai dari *project team* yang akan memberikan *input progress* kerja pada sebuah proses menginputkan *progress work* dan disimpan pada tabel *history coding*. Proses berikutnya adalah menginputkan *progress test*. Proses ini mendapatkan *input* data dari tabel *status sub task* dan *progress test* dari *project team* dan disimpan pada tabel *history test*. Proses setelah itu adalah proses menginputkan *progress train*. Seperti proses sebelumnya, *input* dari proses ini adalah data dari tabel *status sub task* dan *progress train* dari *project team* dan disimpan pada tabel *history train*. Proses terakhir adalah menyetujui proses dimana *project leader* menyetujui *progress* kerja dan disimpan pada tabel *status sub task*. Untuk lebih jelasnya, Gambar 3.13 adalah gambar DFD level 1 subsistem mengimplementasikan proyek.



Gambar 3.13 DFD Level 1 Subsistem Mengimplementasikan Proyek



Gambar 3.15 Physical Data Model

3.5.3 Struktur tabel

Berikut ini akan diuraikan struktur tabel yang nantinya digunakan dalam pembuatan sistem.

1. Tabel UserPPIC

Primary key : Id_User

Foreign key : -

Fungsi : Menyimpan data user pada sistem ini

Tabel 3.2 Tabel UserPPIC

No	Kolom	Type Data	Keterangan
1	<u>Id_User</u>	Varchar (8)	Kode user
2	Nama_User	Varchar (30)	-
3	Password	Varchar (12)	Password user
4	Job	Varchar (20)	Jabatan user

Tabel 3.2 Tabel UserPPIC (Lanjutan)

No	Kolom	Tipe Data	Keterangan
5	Status	Varchar (5)	Status user (aktif/pasif)

2. Tabel Kota

Primary key : Id_Kota

Foreign key : -

Fungsi : Menyimpan data kota

Tabel 3.3 Tabel Kota

No	Kolom	Tipe Data	Keterangan
1	<u>Id_Kota</u>	Varchar (8)	Kode kota
2	Nama_Kota	Varchar (10)	Nama kota
3	Kode_Telp	Varchar (5)	Kode telepon kota
4	Kode_Kota	Varchar (8)	Kode area kota

3. Tabel Customer

Primary key : Id_Customer

Foreign key : Id_Kota *references* ke Id_Kota pada tabel Kota

Fungsi : Menyimpan data pelanggan

Tabel 3.4 Tabel Customer

No	Kolom	Tipe Data	Keterangan
1	<u>Id_Customer</u>	Varchar (8)	Kode pelanggan
2	Nama_Customer	Varchar (30)	Nama pelanggan
3	Alamat_Customer	Varchar (50)	Alamat pelanggan
4	KodePos_Customer	Varchar (50)	Kode pos pelanggan
5	Id_Kota	Varchar (50)	Kode kota pelanggan
6	Website_Customer	Varchar (50)	Website pelanggan
7	Email_Customer	Varchar (50)	Email pelanggan
8	Telepon_Customer	Varchar (50)	Telepon pelanggan
9	Contact Person	Varchar (50)	Nama yang dihubungi
10	Telepon_CP	Varchar (50)	Telepon yang dihubungi
11	Keterangan_Customer	Varchar (50)	Keterangan pelanggan

4. Tabel Project

Primary key : Id_Project

Foreign key : Id_Customer *references* ke Id_Customer pada tabel
Customer

Fungsi : Menyimpan data *project*

Tabel 3.5 Tabel Project

No	Kolom	Tipe Data	Keterangan
1	<u>Id_Project</u>	Varchar (8)	Kode project
2	Id_Customer	Varchar (8)	Kode pelanggan
3	Nama_Project	Varchar (10)	Nama project
4	Tgl_Mulai_Project	Date	Tanggal mulai project
5	Tgl_Selesai_Project	Date	Tanggal selesai project
6	Keterangan_Project	Varchar (MAX)	Keterangan project

5. Tabel Surat_Kontrak

Primary key : Id_SK

Foreign key : Id_Project *references* ke Id_Project pada tabel Project

Fungsi : Menyimpan data surat kontrak dari sebuah *project*

Tabel 3.6 Tabel Surat_Kontrak

No	Kolom	Tipe Data	Keterangan
1	<u>Id_SK</u>	Varchar (8)	Kode surat kontrak
2	Id_Project	Varchar (8)	Kode project
3	Tgl_SK_Project	Date	Tanggal surat kontrak
4	No_SK	Varchar (MAX)	Nomor surat kontrak
5	Keterangan	Varchar (MAX)	Keterangan surat kontrak

6. Tabel Parameter_Biaya

Primary key : Id_Parameter_Biaya

Foreign key : -

Fungsi : Menyimpan data parameter dan harga per parameter

Tabel 3.7 Tabel Parameter_Biaya

No	Kolom	Tipe Data	Keterangan
1	<u>Id_Parameter_Biaya</u>	Varchar (8)	Kode parameter biaya
2	Nama_Parameter	Varchar (MAX)	Nama parameter biaya
3	Kategori_Biaya	Varchar (50)	Kategori biaya
4	Biaya_MH	Number (18,2)	Biaya per <i>man hour</i>
5	Keterangan	Varchar (MAX)	Keterangan parameter

7. Tabel Level

Primary key : Id_Level

Foreign key : -

Fungsi : Menyimpan data level

Tabel 3.8 Tabel Level

No	Kolom	Tipe Data	Keterangan
1	<u>Id_Level</u>	Varchar (8)	Kode Level
2	Nama_Level	Varchar (30)	Nama Level
3	Keterangan_Level	Varchar (MAX)	Keterangan Level

8. Tabel Kompetensi

Primary key : Id_Kompetensi

Foreign key : -

Fungsi : Menyimpan data kompetensi

Tabel 3.9 Tabel Kompetensi

No	Kolom	Tipe Data	Keterangan
1	<u>Id_Kompetensi</u>	Varchar (8)	Kode kompetensi
2	Kompetensi	Varchar (30)	Nama kompetensi
3	Keterangan_Kompetensi	Varchar (MAX)	Keterangan kompetensi

9. Tabel Task

Primary key : Id_Task

Foreign key : *Id_Project references* ke *Id_Project* pada tabel Project

Fungsi : Menyimpan data *task*

Tabel 3.10 Tabel Task

No	Kolom	Tipe Data	Keterangan
1	<u>Id_Task</u>	Varchar (8)	Kode Task
2	Jenis_Task	Varchar (30)	Nama Task
3	Tanggal_Mulai_Task	Date	Tanggal mulai task
4	Tanggal_Selesai_Task	Date	Tanggal selesai task
5	Id_Project	Varchar (8)	Kode project

10. Tabel Sub_Task

Primary key : *Id_Sub_Task*

Foreign key : *Id_Task references* ke *Id_Task* pada tabel Task

Id_User references ke *Id_User* pada tabel UserPPIC

Fungsi : Menyimpan data *sub task*

Tabel 3.11 Tabel Sub_Task

No	Kolom	Tipe Data	Keterangan
1	<u>Id_Sub_Task</u>	Varchar (8)	Kode sub task
2	Id_Task	Varchar (8)	Kode task
3	Nama_Sub_Task	Varchar (30)	Nama sub task
4	Durasi_Sub_Task	Number (3)	Durasi sub task
5	Tgl_Mulai_SubTask	Date	Tanggal mulai sub task
6	Tgl_Selesai_SubTask	Date	Tanggal selesai sub task
7	Id_User	Varchar (8)	Kode user
8	Actual_Mulai	Date	Tanggal aktual mulai sub task
9	Actual_Selesai	Date	Tanggal aktual selesai sub task
10	Approve_Status	Varchar (5)	Persetujuan status sub task
11	Id_User_Test	Varchar (8)	Kode user test
12	Id_User_Train	Varchar (8)	Kode user train

11. Tabel Resource

Primary key : *Id_Resource*

Foreign key : *Id_User references* ke *Id_User* pada tabel UserPPIC

Id_Kompetensi *references* ke *Id_Kompetensi* pada tabel
Kompetensi

Fungsi : Menyimpan data *resource*

Tabel 3.12 Tabel Resource

No	Kolom	Tipe Data	Keterangan
1	<u>Id_Resource</u>	Varchar (8)	Kode resource
2	Id_User	Varchar (8)	Kode user
3	Id_Kompetensi	Varchar (8)	Kode kompetensi

12. Tabel Resource_Level

Primary key : *Id_Resource_Level*

Foreign key : *Id_Project* *references* ke *Id_Project* pada tabel Project

Id_User *references* ke *Id_User* pada tabel UserPPIC

Id_Level *references* ke *Id_Level* pada tabel Level

Fungsi : Menyimpan data *resource* yang memiliki level pada
project



UNIVERSITAS
Dinamika

Tabel 3.13 Tabel Resource_Level

No	Kolom	Tipe Data	Keterangan
1	<u>Id_Resource_Level</u>	Varchar (8)	Kode resource level
2	Id_Project	Varchar (8)	Kode project
3	Id_User	Varchar (8)	Kode user
4	Id_Level	Varchar (8)	Kode level

13. Tabel Project_Cost

Primary key : *Id_Project_Cost*

Foreign key : *Id_Project* *references* ke *Id_Project* pada tabel Project

Id_Parameter_Biaya *references* ke *Id_Parameter_Biaya*
pada tabel Parameter_Biaya

Fungsi : Menyimpan data biaya setiap *project*

Tabel 3.14 Tabel Project_Cost

No	Kolom	Tipe Data	Keterangan
1	<u>Id Project Cost</u>	Varchar (8)	Kode biaya project
2	Id_Project	Varchar (8)	Kode project
3	Id_Parameter_Biaya	Varchar (8)	Kode parameter
4	Parameter_Cost	Number (18,0)	Biaya parameter
5	Keterangan_Project_Cost	Varchar (MAX)	Keterangan biaya

14. Tabel Hasil_Project

Primary key : Id_Hasil_Project

Foreign key : Id_Project *references* ke Id_Project pada tabel Project

Fungsi : Menyimpan hasil *project*

Tabel 3.15 Tabel Hasil_Project

No	Kolom	Tipe Data	Keterangan
1	<u>Id Hasil Project</u>	Varchar (8)	Kode hasil project
2	Id_Project	Varchar (8)	Kode project
3	Status_Project	Varchar (20)	Status project
4	Tgl_Hasil	Date	Tanggal hasil project
5	Keterangan	Varchar (MAX)	Keterangan hasil project

15. Tabel App_Project

Primary key : Id_App_Project

Foreign key : Id_Project *references* ke Id_Project pada tabel Project

Fungsi : Menyimpan hasil persetujuan *project* dan *task*

Tabel 3.16 Tabel App_Project

No	Kolom	Tipe Data	Keterangan
1	<u>Id App Project</u>	Varchar (8)	Kode app project
2	Id_Project	Varchar (8)	Kode project
3	App_Project	Varchar (5)	Persetujuan project
4	Tgl_App_Project	Date	Tanggal app project

Tabel 3.16 Tabel App_Project (Lanjutan)

No	Kolom	Tipe Data	Keterangan
5	App_Task	Varchar (5)	Persetujuan task
6	Tgl_App_Task	Date	Tanggal app task

16. Tabel Sub_Task_Predecessor

Primary key : Id_Predecessor

Foreign key : Id_Sub_Task *references* ke Id_Sub_Task pada tabel
Sub_Task

Fungsi : Menyimpan predesesor/pendahulu dari *sub task*

Tabel 3.17 Tabel Sub_Task_Predecessor

No	Kolom	Tipe Data	Keterangan
1	<u>Id_Predecessor</u>	Varchar (8)	Kode predesesor
2	<u>Id_Sub_Task</u>	Varchar (8)	Kode sub task
3	Predecessor_Sub_Task	Varchar (20)	Pendahulu sub task

17. Tabel History_Coding

Primary key : Id_History_Coding

Foreign key : Id_Sub_Task *references* ke Id_Sub_Task pada tabel
Sub_Task

Fungsi : Menyimpan *progress* kerja *coding*

Tabel 3.18 Tabel History_Coding

No	Kolom	Tipe Data	Keterangan
1	<u>Id_History_Coding</u>	Varchar (8)	Kode history coding
2	<u>Id_Sub_Task</u>	Varchar (8)	Kode sub task
3	Persentasi_Coding	Number (3,0)	Persentasi progress
4	Tanggal_Coding	Date	Tanggal persentasi
5	Keterangan	Varchar (MAX)	Keterangan progress

18. Tabel Testing

Primary key : Id_Testing

Foreign key : Id_Sub_Task *references* ke Id_Sub_Task pada tabel
Sub_Task

Fungsi : Menyimpan *progress* kerja *testing*

Tabel 3.19 Tabel Testing

No	Kolom	Tipe Data	Keterangan
1	<u>Id_Testing</u>	Varchar (8)	Kode history testing
2	Id_Sub_Task	Varchar (8)	Kode sub task
3	Persentasi_Testing	Number (3,0)	Persentasi progress
4	Tanggal_Testing	Date	Tanggal persentasi
5	Keterangan	Varchar (MAX)	Keterangan progress



19. Tabel Training

Primary key : Id_Training

Foreign key : Id_Sub_Task *references* ke Id_Sub_Task pada tabel
Sub_Task

Fungsi : Menyimpan *progress* kerja *training*

Tabel 3.20 Tabel Training

No	Kolom	Tipe Data	Keterangan
1	<u>Id_Training</u>	Varchar (8)	Kode history training
2	Id_Sub_Task	Varchar (8)	Kode sub task
3	Persentasi_Training	Number (3,0)	Persentasi progress
4	Tanggal_Training	Date	Tanggal persentasi
5	Keterangan	Varchar (MAX)	Keterangan progress

20. Tabel Status_Sub_Task

Primary key : Id_Status_Sub_Task

Foreign key : Id_Sub_Task *references* ke Id_Sub_Task pada tabel

Sub_Task

Fungsi : Menyimpan persetujuan untuk *coding, testing, training* sebuah *sub task*

Tabel 3.21 Tabel Status_Sub_Task

No	Kolom	Tipe Data	Keterangan
1	<u>Id_Status_Sub_Task</u>	Varchar (8)	Kode status
2	Id_Sub_Task	Varchar (8)	Kode sub task
3	Approve_Status	Varchar (5)	Persetujuan status
4	Tgl_Approve_Status	Date	Tanggal approve status
5	Jenis_Status	Varchar (20)	Jenis status

21. Tabel Sub2_Task



Primary key : Id_Sub2_Task

Foreign key : Id_Sub_Task *references* ke Id_Sub_Task pada tabel Sub_Task

Fungsi : Menyimpan data level dibawah tabel sub task

Tabel 3.22 Tabel Sub2_Task

No	Kolom	Tipe Data	Keterangan
1	<u>Id_Sub2_Task</u>	Varchar (8)	Kode sub2 task
2	Id_Sub_Task	Varchar (8)	Kode sub task
3	Nama_Sub2_Task	Varchar (50)	Nama Sub2 Task
4	Keterangan_Sub2_Task	Varchar (999)	Keterangan Sub2 Task

22. Tabel Sub3_Task

Primary key : Id_Sub3_Task

Foreign key : Id_Sub2_Task *references* ke Id_Sub2_Task pada tabel Sub2_Task

Fungsi : Menyimpan data dua level dibawah tabel sub task

Tabel 3.23 Tabel Sub3_Task

No	Kolom	Tipe Data	Keterangan
1	<u>Id_Sub3_Task</u>	Varchar (8)	Kode sub3 task
2	Id_Sub2_Task	Varchar (8)	Kode sub2 task
3	Nama_Sub3_Task	Varchar (50)	Nama Sub3 Task
4	Keterangan_Sub3_Task	Varchar (999)	Keterangan Sub3 Task

3.6 Desain *Input/Output*

Setelah melakukan perancangan basis data, tahap selanjutnya adalah membuat desain *input/output*. Desain ini berguna untuk menggambarkan interaksi antara pengguna dengan sistem yang dibuat.

3.6.1 Desain *form login*

Form login ini digunakan untuk melakukan validasi terhadap *user* sebelum masuk ke dalam sistem yang dibuat. Pada *form* ini, *user* harus memasukkan *id user* pada kolom *id user* dan *password* pada kolom *password*. Untuk lebih jelasnya, desain *form login* ini dapat dilihat pada Gambar 3.16.

The diagram shows a login form layout. It consists of two text input fields stacked vertically. The first field is labeled 'Id User' and the second is labeled 'Password'. Below these fields is a rectangular button labeled 'Login'. The entire form is enclosed in a thin black border.

Gambar 3.16 Desain *Form Login*

Fungsi objek pada *form login* adalah sebagai berikut:

Tabel 3.24 Fungsi Objek *Form Login*

Nama Objek	Tipe Objek	Fungsi
Id User	<i>Textbox</i>	Mengisi <i>id user</i>
Password	<i>Textbox</i>	Mengisi <i>password</i>
Login	<i>Button</i>	Masuk kedalam sistem

3.6.2 Desain form master user

Form master user ini digunakan oleh administrator untuk menyimpan dan mengubah *user* / status *user*. Desain *form master user* ini dapat dilihat pada Gambar 3.17.

Id User	Nama	Job	Status

Gambar 3.17 Desain Form Master User

Fungsi objek pada *form master user* adalah sebagai berikut:

Tabel 3.25 Fungsi Objek Form Master User

Nama Objek	Tipe Objek	Fungsi
NIK	<i>Textbox</i>	Mengisi NIK/id <i>user</i>
User Name	<i>Textbox</i>	Mengisi nama <i>user</i>
Password	<i>Textbox</i>	Mengisi <i>password</i>
Re-Type Password	<i>Textbox</i>	Mengisi ulang <i>password</i>
Job	<i>Combobox</i>	Memilih posisi dalam sistem
Save	<i>Button</i>	Menyimpan data <i>user</i>
Reset	<i>Button</i>	Mengosongkan semua <i>field</i> di <i>form</i>
Tabel User	<i>Datagrid</i>	Menampilkan data <i>user</i>

3.6.3 Desain form menu utama market

Form menu utama *market* ini digunakan oleh *marketing* untuk melihat *project*, memasuki *form master project*, *form master sk*, *form master cost*, dan

form hasil *project*. Desain *form* menu utama *market* ini dapat dilihat pada Gambar 3.18.

				Master Project	In Progress	V
Nama Project	Manajer App	No. SK	Status	Cost	Cancel	Keterangan
XXX	V	XXX	In Progress	Lihat Cost	Cancel Project	Keterangan

Gambar 3.18 Desain *Form* Menu Utama *Market*

Fungsi objek pada *form* menu utama *market* adalah sebagai berikut:

Tabel 3.26 Fungsi Objek *Form* Menu Utama *Market*

Nama Objek	Type Objek	Fungsi
Master Project	Button	Memasuki form <i>Master Project</i>
Status	Combobox	Menyortir berdasarkan status
Tabel Project	Datagrid	Menampilkan data <i>project</i>
No. SK	Link	Melihat <i>master</i> surat kontrak
Status	Link	Melihat detail proyek
Cost	Link	Melihat data biaya
Cancel	Link	Membatalkan proyek
Keterangan	Link	Mengetahui keterangan proyek

3.6.4 Desain *form* master kota

Form master kota ini digunakan oleh *marketing* untuk menambah dan mengubah data kota. Master kota juga digunakan agar mengetahui kode telepon dari masing-masing pelanggan. Desain *form* master kota ini dapat dilihat pada Gambar 3.19.

Kode Kota	<input type="text"/>
Nama Kota	<input type="text"/>
Kode Telp	<input type="text"/>
<input type="button" value="Save"/>	<input type="button" value="Reset"/>

Kode Kota	Nama Kota	Kode Telp

Gambar 3.19 Desain *Form Master Kota*

Fungsi objek pada *form master kota* adalah sebagai berikut:

Tabel 3.27 Fungsi Objek *Form Master Kota*

Nama Objek	Tipe Objek	Fungsi
Kode Kota	<i>Textbox</i>	Mengisi kode kota
Nama Kota	<i>Textbox</i>	Mengisi nama kota
Kode Telp	<i>Textbox</i>	Mengisi kode telp
Save	<i>Button</i>	Menyimpan data kota
Reset	<i>Button</i>	Mengosongkan semua <i>field</i> di <i>form</i>
Tabel Kota	<i>Datagrid</i>	Menampilkan data kota

3.6.5 Desain *form master customer*

Form master customer ini digunakan oleh *marketing* untuk menambah atau mengubah data *customer*. Desain *form master customer* ini dapat dilihat pada Gambar 3.20.

Nama Customer	Alamat Customer	Telepon	Contact Person

Gambar 3.20 Desain *Form Master Customer*

Fungsi objek pada *form master customer* adalah sebagai berikut:

Tabel 3.28 Fungsi Objek *Form Master Customer*

Nama Objek	Tipe Objek	Fungsi
Id Customer	<i>Label</i>	Tempat kode pelanggan
Nama Customer	<i>Textbox</i>	Mengisi nama pelanggan
Alamat Customer	<i>Textbox</i>	Mengisi alamat pelanggan
Kode Post	<i>Textbox</i>	Mengisi kode pos pelanggan
Kota	<i>Combobox</i>	Memilih kota pelanggan
Telp	<i>Textbox</i>	Mengisi nomor telepon pelanggan
Email	<i>Textbox</i>	Mengisi email pelanggan
Website	<i>Textbox</i>	Mengisi website pelanggan
Contact Person	<i>Textbox</i>	Mengisi nama yang bisa dihubungi
Telp CP	<i>Textbox</i>	Mengisi nomor telepon yang bisa dihubungi
Catatan	<i>Textbox</i>	Mengisi catatan tentang pelanggan
Save	<i>Button</i>	Menyimpan data pelanggan
Reset	<i>Button</i>	Mengosongkan semua <i>field</i> di <i>form</i>
Tabel Customer	<i>Datagrid</i>	Menampilkan data pelanggan

3.6.6 Desain form master project

Form master project ini digunakan oleh marketing untuk menambah dan mengubah data project. Desain form master project ini dapat dilihat pada Gambar 3.21.

The form contains the following elements:

- Id Project:
- Nama Project:
- Nama Customer:
- Tanggal Mulai:
- Tanggal Selesai:
- Keterangan:
-

Nama Project	Nama Customer	Tanggal Mulai	Tanggal Selesai

Gambar 3.21 Desain Form Master Project

Fungsi objek pada form master project adalah sebagai berikut:

Tabel 3.29 Fungsi Objek Form Master Project

Nama Objek	Tipe Objek	Fungsi
Id Project	Label	Tempat kode proyek
Nama Project	Textbox	Mengisi nama proyek
Nama Customer	Textbox	Tempat nama pelanggan
Find	Button	Mencari nama pelanggan
Tanggal Mulai	Date	Mengisi tanggal mulai proyek
Tanggal Selesai	Date	Mengisi tanggal selesai proyek
Keterangan	Textbox	Mengisi keterangan proyek
Save	Button	Menyimpan data proyek
Reset	Button	Mengosongkan semua field di form
Tabel Customer	Datagrid	Menampilkan data pelanggan

3.6.7 Desain form pencarian customer

Form pencarian *customer* digunakan oleh marketing untuk mencari nama *customer* yang akan ditambahkan dalam *master project*. Desain *form* pencarian *customer* ini dapat dilihat pada Gambar 3.22.

Id Customer	Nama Customer

Gambar 3.22 Desain *Form* Pencarian *Customer*

Fungsi objek pada *form* pencarian *customer* adalah sebagai berikut:

Tabel 3.30 Fungsi Objek *Form* Pencarian *Customer*

Nama Objek	Tipe Objek	Fungsi
Tabel Customer	<i>Datagrid</i>	Menampilkan id dan nama pelanggan

3.6.8 Desain form surat kontrak

Form surat kontrak ini digunakan oleh *marketing* untuk memasukkan isi surat kontrak yang sudah disetujui. Desain *form* surat kontrak ini dapat dilihat pada Gambar 3.23.

Id Surat Kontrak	<input type="text"/>
Id Project	<input type="text"/>
Tanggal Surat Kontrak	<input type="text"/> v
Nomor Surat Kontrak	<input type="text"/>
Keterangan	<input type="text"/>
	<input type="button" value="Save"/>

Gambar 3.23 Desain *Form* Surat Kontrak

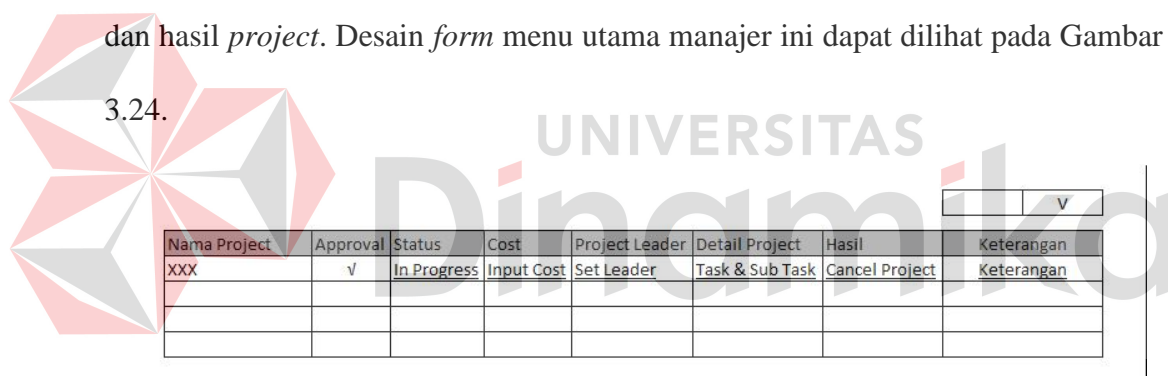
Fungsi objek pada *form* surat kontrak adalah sebagai berikut:

Tabel 3.31 Fungsi Objek *Form* Surat Kontrak

Nama Objek	Tipe Objek	Fungsi
Id Surat Kontrak	<i>Label</i>	Tempat id surat kontrak
Id Project	<i>Label</i>	Menampilkan kode proyek
Tanggal Surat Kontrak	<i>Date</i>	Mengisi tanggal surat kontrak
Nomor Surat Kontrak	<i>Text</i>	Mengisi nomor surat kontrak
Keterangan	<i>Text</i>	Mengisi keterangan surat kontrak
Save	<i>Button</i>	Menyimpan data surat kontrak

3.6.9 Desain *form* menu utama manajer

Form menu manajer ini digunakan manajer oleh melihat *project*, menyetujui *project*, memasuki *form master* biaya, *resource level*, *approve task*, dan hasil *project*. Desain *form* menu utama manajer ini dapat dilihat pada Gambar 3.24.



Nama Project	Approval	Status	Cost	Project Leader	Detail Project	Hasil	Keterangan
XXX	v	In Progress	Input Cost	Set Leader	Task & Sub Task	Cancel Project	Keterangan

Gambar 3.24 Desain *Form* Menu Utama Manajer

Fungsi objek pada *form* menu utama manajer adalah sebagai berikut:

Tabel 3.32 Fungsi Objek *Form* Menu Utama Manajer

Nama Objek	Tipe Objek	Fungsi
Status	<i>Combobox</i>	Menyortir berdasarkan status
Data Project	<i>Datagrid</i>	Menampilkan data proyek
Approval	<i>Check</i>	Menyetujui sebuah proyek
Status	<i>Link</i>	Mengetahui status proyek
Cost	<i>Link</i>	Mengisi biaya proyek
Project Leader	<i>Link</i>	Memilih pimpinan proyek
Detail Project	<i>Link</i>	Mengetahui detail proyek
Hasil	<i>Link</i>	Membatalkan/Menutup proyek
Keterangan	<i>Link</i>	Mengetahui keterangan proyek

3.6.10 Desain *form master* kompetensi

Form master kompetensi ini digunakan oleh manajer untuk menambah atau mengubah data kompetensi. Desain *form master* kompetensi ini dapat dilihat pada Gambar 3.25.

The image shows a web form for managing competencies. It includes three text input fields labeled 'Id Kompetensi', 'Nama Kompetensi', and 'Keterangan'. Below the inputs are two buttons: 'Save' and 'Reset'. At the bottom of the form is a data grid with the header 'Nama Kompetensi' and four empty rows. A large watermark for 'UNIVERSITAS Dinamika' is overlaid on the image.

Gambar 3.25 Desain *Form Master* Kompetensi

Fungsi objek pada *form master* kompetensi adalah sebagai berikut:

Tabel 3.33 Fungsi Objek *Form Master* Kompetensi

Nama Objek	Tipe Objek	Fungsi
Id Kompetensi	<i>Label</i>	Tempat kode kompetensi
Nama Kompetensi	<i>Text</i>	Mengisi nama kompetensi
Keterangan	<i>Text</i>	Mengisi keterangan kompetensi
Save	<i>Button</i>	Menyimpan data kompetensi
Reset	<i>Button</i>	Mengosongkan semua <i>field</i> di <i>form</i>
Tabel Customer	<i>Datagrid</i>	Menampilkan data kompetensi

3.6.11 Desain *form master* parameter

Form master parameter ini digunakan oleh manajer untuk menambah atau mengubah data parameter. Desain *form master* parameter ini dapat dilihat pada Gambar 3.26.

Nama Parameter	Kategori Biaya	Biaya Per MH

Gambar 3.26 Desain *Form Master Parameter*

Fungsi objek pada *form master Parameter* adalah sebagai berikut:

Tabel 3.34 Fungsi Objek *Form Master Parameter*

Nama Objek	Tipe Objek	Fungsi
Id Parameter Biaya	<i>Label</i>	Tempat kode parameter biaya
Nama Parameter	<i>Text</i>	Mengisi nama parameter biaya
Kategori Biaya	<i>Combobox</i>	Memilih kategori biaya
Biaya per man hour	<i>Text</i>	Mengisi biaya per orang dan waktu
Keterangan	<i>Text</i>	Mengisi keterangan parameter biaya
Save	<i>Button</i>	Menyimpan data parameter biaya
Reset	<i>Button</i>	Mengosongkan semua <i>field</i> di <i>form</i>
Tabel Parameter	<i>Datagrid</i>	Menampilkan data parameter biaya

3.6.12 Desain *form master level*

Form master level ini digunakan oleh manajer untuk menambah atau mengubah data level. Desain *form master level* ini dapat dilihat pada Gambar 3.27.

Gambar 3.27 Desain *Form Master Level*

Fungsi objek pada *form master level* adalah sebagai berikut:

Tabel 3.35 Fungsi Objek *Form Master Level*

Nama Objek	Tipe Objek	Fungsi
Id Level	<i>Label</i>	Tempat kode <i>level</i>
Nama Level	<i>Text</i>	Mengisi nama <i>level</i>
Keterangan	<i>Text</i>	Mengisi keterangan <i>level</i>
Save	<i>Button</i>	Menyimpan data <i>level</i>
Reset	<i>Button</i>	Mengosongkan semua <i>field</i> di <i>form</i>
Tabel Level	<i>Datagrid</i>	Menampilkan data <i>level</i>

3.6.13 Desain *form master resource*

Form master resource ini digunakan oleh manajer untuk menambah atau mengubah data *resource*. *Resource* adalah seorang *user* yang telah memiliki kompetensi. Desain *form master resource* ini dapat dilihat pada Gambar 3.28.

Id Resource

Nama Resource

Kompetensi

Nama Kompetensi	Action
XXX	Delete

Nama User	Nama Kompetensi

Gambar 3.28 Desain *Form Master Resource*

Fungsi objek pada *form master resource* adalah sebagai berikut:

Tabel 3.36 Fungsi Objek *Form Master Resource*

Nama Objek	Tipe Objek	Fungsi
Id Resource	<i>Label</i>	Tempat kode <i>resource</i>
Nama Resource	<i>Text</i>	Tempat nama <i>resource</i>
Find	<i>Button</i>	Mencari nama <i>resource</i>
Kompetensi	<i>Text</i>	Tempat nama kompetensi
Add	<i>Button</i>	Menambah kompetensi pada seorang <i>user</i>
Tabel Daftar Kompetensi	<i>Datagrid</i>	Menampilkan data kompetensi
Delete	<i>Link</i>	Menghapus kompetensi pada seorang <i>user</i>
Save	<i>Button</i>	Menyimpan data <i>resource</i>
Reset	<i>Button</i>	Mengosongkan semua <i>field</i> di <i>form</i>
Tabel Resource	<i>Datagrid</i>	Menampilkan data <i>resource</i>

3.6.14 Desain *form* pencarian *user*

Form pencarian *user* ini digunakan oleh manajer dan *project leader* untuk mencari nama *user* pada *form resource*, *resource level* dan *sub task*. Desain *form* pencarian *user* ini dapat dilihat pada Gambar 3.29.

Id User	Nama User

Gambar 3.29 Desain *Form Pencarian User*

Fungsi objek pada *form pencarian user* adalah sebagai berikut:

Tabel 3.37 Fungsi Objek *Form Pencarian User*

Nama Objek	Tipe Objek	Fungsi
Tabel User	<i>Datagrid</i>	Menampilkan data <i>user</i>

3.6.15 Desain *form master cost*

Form master cost ini digunakan oleh manajer untuk memasukkan biaya *project* dan oleh *marketing* untuk melihat biaya. Desain *form master cost* ini dapat dilihat pada Gambar 3.30.

Nama Project				
Kategori		V		
Nama Parameter	Biaya	Man	Hour	Total Biaya
Save				

Gambar 3.30 Desain *Form Master Cost*

Fungsi objek pada *form master cost* adalah sebagai berikut:

Tabel 3.38 Fungsi Objek *Form Master Cost*

Nama Objek	Tipe Objek	Fungsi
Nama Project	<i>Text</i>	Tempat nama proyek
Kategori	<i>Combobox</i>	Memilih nama kategori biaya
Tabel Biaya	<i>Datagrid</i>	Menampilkan data biaya
Save	<i>Button</i>	Menyimpan data biaya

3.6.16 Desain *form resource level*

Form resource level ini digunakan oleh manajer dan *project leader* untuk membentuk tim yang akan mengerjakan proyek. Manajer akan memilih *project leader* dan *project leader* akan memilih *project team*. Desain *form resource level* ini dapat dilihat pada Gambar 3.31.

The screenshot shows a web form titled 'Form Resource Level'. On the left is a sidebar with a red and white geometric logo. The main form area contains the following elements:

- Input field for 'Id Resource Level'.
- Input field for 'Id Project'.
- Input field for 'Nama User' with a 'Find' button to its right.
- Input field for 'Level' with a dropdown arrow and a 'v' indicator.
- A 'Save' button below the 'Level' field.
- A table with two columns: 'Nama User' and 'Nama Level'. The table has four rows, with the first row being a header and the others being empty.

Gambar 3.31 Desain *Form Resource Level*

Fungsi objek pada *form resource level* adalah sebagai berikut:

Tabel 3.39 Fungsi Objek *Form Resource Level*

Nama Objek	Tipe Objek	Fungsi
Id Resource Level	<i>Label</i>	Tempat kode <i>resource level</i>
Id Project	<i>Text</i>	Tempat kode proyek
Nama User	<i>Text</i>	Tempat nama <i>user</i>
Find	<i>Button</i>	Mencari <i>user</i>
Level	<i>Combobox</i>	Memilih level pada proyek
Save	<i>Button</i>	Menyimpan data <i>resource level</i>
Tabel Resource Level	<i>Datagrid</i>	Menampilkan data <i>resource level</i>

3.6.17 Desain form view and app task

Form view and app task ini digunakan oleh manajer untuk menyetujui *task* dan *sub task* yang ada dan melihat *ganttt chart*. *Project leader* menggunakannya untuk melihat *ganttt chart*. Desain *form view and app task* dapat dilihat pada Gambar 3.32.

Project Leader

Nama Task	Sub Task	Tgl Mulai	Tgl Selesai	Act Mulai	Act Selesai	Project Team	Tester	Trainer	Biaya	Act Biaya

Gambar 3.32 Desain *Form View and App Task*

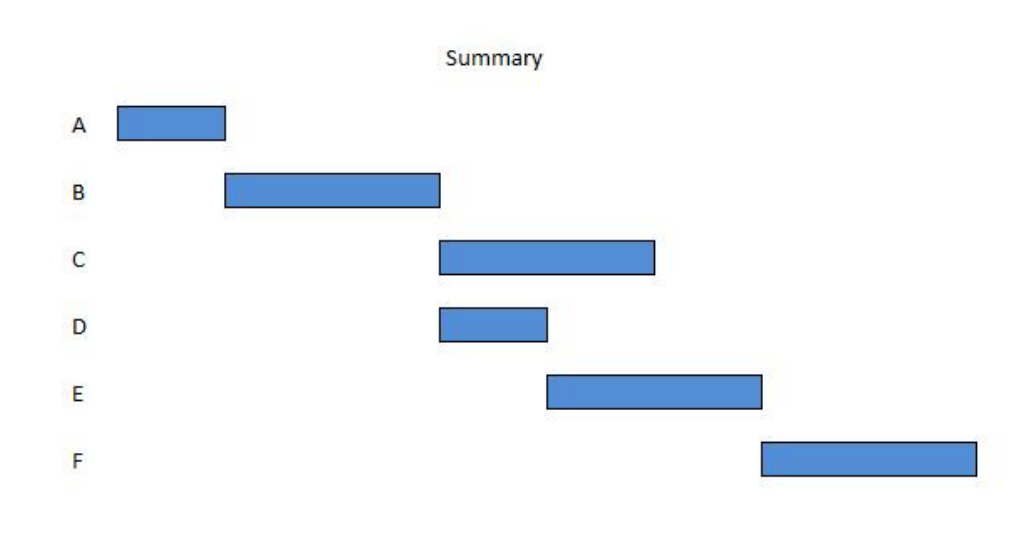
Fungsi objek pada *form view and app task* adalah sebagai berikut:

Tabel 3.40 Fungsi Objek *Form View and App Task*

Nama Objek	Tipe Objek	Fungsi
Project Leader	<i>Label</i>	Tempat nama pemimpin proyek
Tabel Proyek	<i>Datagrid</i>	Menampilkan data proyek
Approve Task	<i>Button</i>	Menyetujui <i>task</i> dari proyek
Summary	<i>Button</i>	Melihat <i>sub task critical path</i>

3.6.18 Desain form summary

Form summary ini digunakan oleh manajer dan *project leader* untuk melihat *ganttt chart* dari sebuah *project* dan *ganttt chart critical path* dari sebuah proyek. Desain *form summary* ini dapat dilihat pada Gambar 3.33.



Gambar 3.33 Desain *Form Summary*

Fungsi objek pada *form summary* adalah sebagai berikut:

Tabel 3.41 Fungsi Objek *Form Summary*

Nama Objek	Tipe Objek	Fungsi
Gantt Chart	<i>Chart</i>	Menampilkan gantt chart dari sebuah proyek

3.6.19 Desain *form review project*

Form review project ini digunakan oleh *project leader* dan *project team* untuk mengetahui *project* mana yang akan dikerjakan. Desain *form review project* ini dapat dilihat pada Gambar 3.34.

Nama Project	Status

Gambar 3.34 Desain *Form Review Project*

Fungsi objek pada *form review project* adalah sebagai berikut:

Tabel 3.42 Fungsi Objek *Form Review Project*

Nama Objek	Tipe Objek	Fungsi
Tabel Project	<i>Datagrid</i>	Menampilkan data proyek

3.6.20 Desain form project leader taskboard

Form project leader taskboard ini digunakan oleh *project leader* untuk membuat *task*, memasuki *form resource level*, *master sub task*, *approve sub task*, dan *view and app task*. Desain *form project leader taskboard* ini dapat dilihat pada Gambar 3.35.

Gambar 3.35 Desain *Form Project Leader Taskboard*

Fungsi objek pada *form project leader taskboard* adalah sebagai berikut:

Tabel 3.43 Fungsi Objek *Form Project Leader Taskboard*

Nama Objek	Tipe Objek	Fungsi
Resource Level	<i>Button</i>	Mengubah data <i>resource level</i>
Id Project	<i>Textbox</i>	Tempat kode proyek
Tanggal Project	<i>Textbox</i>	Tempat tanggal proyek
Id Task	<i>Label</i>	Tempat kode <i>task</i>
Nama Task	<i>Textbox</i>	Mengisi nama <i>task</i>
Tanggal Mulai	<i>Date</i>	Mengisi tanggal mulai <i>task</i>
Tanggal Selesai	<i>Date</i>	Mengisi tanggal selesai <i>task</i>
Save	<i>Button</i>	Menyimpan data <i>task</i>
Reset	<i>Button</i>	Mengosongkan semua <i>field</i> di <i>task</i>
Approve Sub Task	<i>Button</i>	Melihat <i>form approve sub task</i>

Tabel 3.43 Fungsi Objek *Form Project Leader Taskboard* (Lanjutan)

Nama Objek	Tipe Objek	Fungsi
Tabel Task	<i>Datagrid</i>	Menampilkan data <i>task</i>
Buat Sub Task	<i>Link</i>	Membuat <i>sub task</i>
View Summary and Cost	<i>Button</i>	Melihat <i>form summary</i>

3.6.21 Desain *form master sub task*

Form master sub task ini digunakan oleh *project leader* untuk membuat *sub task*, mengubah *sub task*, menjabarkan *sub task* ke level yang lebih detail, menambah predesesor, dan menghapus predesesor dari sebuah *sub task*. Desain *form master sub task* ini dapat dilihat pada Gambar 3.36.

Gambar 3.36 Desain *Form Master Sub Task*

Fungsi objek pada *form master sub task* adalah sebagai berikut:

Tabel 3.44 Fungsi Objek *Form Master Sub Task*

Nama Objek	Tipe Objek	Fungsi
Id Project	<i>Textbox</i>	Tempat kode proyek
Id Task	<i>Textbox</i>	Tempat kode <i>task</i>

Tabel 3.44 Fungsi Objek *Form Master Sub Task* (Lanjutan)

Nama Objek	Tipe Objek	Fungsi
Id Sub Task	<i>Label</i>	Tempat kode <i>sub task</i>
Nama Sub Task	<i>Textbox</i>	Mengisi nama <i>sub task</i>
Durasi	<i>Textbox</i>	Mengisi durasi <i>sub task</i>
Down Level	<i>Button</i>	Melihat <i>form down level sub task</i>
Nama Project Team	<i>Textbox</i>	Tempat nama <i>project team</i>
Find User	<i>Button</i>	Mencari nama <i>project team</i>
Nama User Tester	<i>Textbox</i>	Tempat nama <i>project tester</i>
Find Tester	<i>Button</i>	Mencari nama <i>project tester</i>
Nama User Trainer	<i>Textbox</i>	Tempat nama <i>project trainer</i>
Find Trainer	<i>Button</i>	Mencari nama <i>project trainer</i>
Predesesor	<i>Textbox</i>	Tempat predesesor
Find Predesesor	<i>Button</i>	Mencari predesesor
Add	<i>Button</i>	Menambah predesesor
Tabel Predesesor	<i>Datagrid</i>	Melihat data predesesor
Delete	<i>Link</i>	Menghapus data predesesor
Save	<i>Button</i>	Menyimpan data <i>sub task</i>
Reset	<i>Button</i>	Mengosongkan semua <i>field</i> di <i>form</i>
Tabel Sub Task	<i>Datagrid</i>	Melihat data <i>sub task</i>

3.6.22 Desain *form* pencarian predesesor *sub task*

Form pencarian predesesor *sub task* ini digunakan oleh *project leader* untuk mencari kemungkinan *sub task* yang bisa menjadi predesesor dari *sub task* yang lain. Desain *form* pencarian predesesor *sub task* ini dapat dilihat pada Gambar 3.37.

The image shows a rectangular form with a title bar at the top containing the text 'Nama Sub Task'. Below the title bar, there are four horizontal input fields, each separated by a thin line, intended for entering data.

Gambar 3.37 Desain *Form* Pencarian Predesesor *Sub Task*

Fungsi objek pada *form* pencarian predesesor *sub task* adalah sebagai berikut:

Tabel 3.45 Fungsi Objek *Form* Pencarian Predecessor *Sub Task*

Nama Objek	Tipe Objek	Fungsi
Tabel Predecessor	<i>Datagrid</i>	Melihat data <i>predecessor sub task</i>

3.6.23 Desain *form down level sub task*

Form down level sub task ini digunakan oleh *project leader* untuk mengisi keterangan yang lebih spesifik dari *sub task* dan *project team* untuk melihat detail *sub task*. Desain *form down level sub task* ini dapat dilihat pada Gambar 3.38.

Gambar 3.38 Desain *Form Down Level Sub Task*

Fungsi objek pada *form down level sub task* adalah sebagai berikut:

Tabel 3.46 Fungsi Objek *Form Down Level Sub Task*

Nama Objek	Tipe Objek	Fungsi
Id Sub Task	<i>Textbox</i>	Tempat kode <i>sub task</i>
Id Detail Sub Task	<i>Label</i>	Tempat kode <i>detail sub task</i>
Nama Detail	<i>Textbox</i>	Mengisi nama <i>detail sub task</i>
Keterangan	<i>Textbox</i>	Mengisi keterangan <i>detail sub task</i>
Save	<i>Button</i>	Menyimpan data <i>detail sub task</i>
Reset	<i>Button</i>	Mengosongkan semua <i>field</i> di <i>detail sub task</i>
Delete	<i>Button</i>	Menghapus <i>detail sub task</i>
Sub Detail	<i>Button</i>	Melihat <i>sub detail sub task</i>
Tabel Detail Sub Task	<i>Datagrid</i>	Melihat data <i>detail sub task</i>
Id Sub Detail Sub Task	<i>Label</i>	Tempat kode <i>sub detail sub task</i>
Nama Sub Detail	<i>Textbox</i>	Mengisi nama <i>sub detail sub task</i>
Keterangan	<i>Textbox</i>	Mengisi keterangan <i>sub detail sub task</i>

Tabel 3.46 Fungsi Objek *Form Down Level Sub Task* (Lanjutan)

Nama Objek	Tipe Objek	Fungsi
Save	<i>Button</i>	Menyimpan data <i>sub detail sub task</i>
Reset	<i>Button</i>	Mengosongkan semua <i>field</i> di <i>sub detail sub task</i>
Delete	<i>Button</i>	Menghapus <i>sub detail sub task</i>
Tabel Sub Detail Sub Task	<i>Datagrid</i>	Melihat data <i>sub detail sub task</i>

3.6.24 Desain form project team taskboard

Form project team taskboard ini digunakan oleh *project team* untuk melihat pekerjaan yang harus dikerjakan dan memasuki *form history work*. Desain *form project team taskboard* ini dapat dilihat pada Gambar 3.39.

Gambar 3.39 Desain *Form Project Team Taskboard*

Fungsi objek pada *form project team taskboard* adalah sebagai berikut:

Tabel 3.47 Fungsi Objek *Form Project Team Taskboard*

Nama Objek	Tipe Objek	Fungsi
Id Project	<i>Textbox</i>	Tempat kode <i>project</i>
Tabel Sub Task	<i>Datagrid</i>	Melihat data sub task
Progress	<i>Progressbar</i>	Mengisi data <i>progress</i>
Predesesor	<i>Link</i>	Melihat predesesor sub task

3.6.25 Desain form history work

Form history work ini digunakan oleh *project team* untuk memasukkan inputan kerja. Desain *form history work* ini dapat dilihat pada Gambar 3.40.

The image shows a web form titled 'Form History Work'. It contains the following elements:

- Four text input fields labeled: 'Id History', 'Nama Sub Task', 'Persentasi', and 'Keterangan'.
- Two buttons labeled 'Save' and 'Reset' positioned below the input fields.
- A table at the bottom with two columns: 'Persentasi' and 'Tanggal'. The table has four empty rows for data entry.

Gambar 3.40 Desain *Form History Work*

Fungsi objek pada *form history work* adalah sebagai berikut:

Tabel 3.48 Fungsi Objek *Form History Work*

Nama Objek	Tipe Objek	Fungsi
Id History	<i>Label</i>	Tempat kode <i>history work</i>
Nama Sub Task	<i>Textbox</i>	Tempat nama <i>sub task</i>
Persentasi	<i>Textbox</i>	Mengisi data persentasi <i>work</i>
Keterangan	<i>Textbox</i>	Mengisi keterangan <i>history work</i>
Save	<i>Button</i>	Menyimpan data <i>history work</i>
Reset	<i>Button</i>	Mengosongkan semua <i>field</i> di <i>form</i>
Tabel History Work	<i>Datagrid</i>	Melihat data <i>history work</i>

3.6.26 Desain *form project test taskboard*

Form project test taskboard ini digunakan oleh *project test* untuk melihat pekerjaan yang harus dikerjakan dan memasuki *form history test*. Desain *form project test taskboard* ini dapat dilihat pada Gambar 3.41.

Id Project

Nama Sub Task	Tgl Mulai	Tgl Selesai	Approve Test	Progress	Progress (dihara
XXX	XX/XX/XXXX	XX/XX/XXXX	√	<div style="width: 100%; height: 10px; background-color: #007bff;"></div>	<div style="width: 100%; height: 10px; background-color: #007bff;"></div>

Gambar 3.41 Desain *Form Project Test Taskboard*

Fungsi objek pada *form project test taskboard* adalah sebagai berikut:

Tabel 3.49 Fungsi Objek *Form Project Test Taskboard*

Nama Objek	Tipe Objek	Fungsi
Id Project	<i>Textbox</i>	Tempat kode <i>project</i>
Tabel Sub Task	<i>Datagrid</i>	Melihat data sub task
Progress	<i>Progressbar</i>	Mengisi data <i>progress</i>

3.6.27 Desain *form history test*

Form history test ini digunakan oleh *project test* untuk memasukkan inputan kerja *test*. Desain *form history test* ini dapat dilihat pada Gambar 3.42.

Id History

Nama Sub Task

Persentasi

Keterangan

Persentasi	Tanggal Testing

Gambar 3.42 Desain *Form History Test*

Fungsi objek pada *form history test* adalah sebagai berikut:

Tabel 3.50 Fungsi Objek *Form History Test*

Nama Objek	Tipe Objek	Fungsi
Id History	<i>Label</i>	Tempat kode <i>history test</i>
Nama Sub Task	<i>Textbox</i>	Tempat nama <i>sub task</i>
Persentasi	<i>Textbox</i>	Mengisi data persentasi <i>test</i>
Keterangan	<i>Textbox</i>	Mengisi keterangan <i>history test</i>
Save	<i>Button</i>	Menyimpan data <i>history test</i>
Reset	<i>Button</i>	Mengosongkan semua <i>field</i> di <i>form</i>
Tabel History Test	<i>Datagrid</i>	Melihat data <i>history test</i>

3.6.28 Desain *form project train taskboard*

Form project train taskboard ini digunakan oleh *project train* untuk melihat pekerjaan yang harus dikerjakan dan memasuki *form history train*. Desain *form project train taskboard* ini dapat dilihat pada Gambar 3.43.

Nama Sub Task	Tgl Mulai	Tgl Selesai	Approve Train	Progress	Progress (dihara)
xxx	xx/xx/xxxx	xx/xx/xxxx	√	<div style="width: 25%; background-color: blue;"></div>	<div style="width: 25%; background-color: white;"></div>

Gambar 3.43 Desain *Form Project Train Taskboard*

Fungsi objek pada *form project train taskboard* adalah sebagai berikut:

Tabel 3.51 Fungsi Objek *Form Project Train Taskboard*

Nama Objek	Tipe Objek	Fungsi
Id Project	<i>Textbox</i>	Tempat kode <i>project</i>
Tabel Sub Task	<i>Datagrid</i>	Melihat data <i>sub task</i>
Progress	<i>Progressbar</i>	Mengisi data <i>progress</i>

3.6.29 Desain form history train

Form history train ini digunakan oleh *project train* untuk memasukkan inputan kerja *train*. Desain *form history train* ini dapat dilihat pada Gambar 3.44.

Gambar 3.44 Desain Form History Train

Fungsi objek pada *form history train* adalah sebagai berikut:

Tabel 3.52 Fungsi Objek Form History Train

Nama Objek	Tipe Objek	Fungsi
Id History	Label	Tempat kode <i>history train</i>
Nama Sub Task	Textbox	Tempat nama <i>sub task</i>
Persentasi	Textbox	Mengisi data persentasi <i>train</i>
Keterangan	Textbox	Mengisi keterangan <i>history train</i>
Save	Button	Menyimpan data <i>history train</i>
Reset	Button	Mengosongkan semua <i>field</i> di <i>form</i>
Tabel History Train	Datagrid	Melihat data <i>history train</i>

3.6.30 Desain form approve sub task

Form approve sub task ini digunakan oleh *project leader* untuk memasuki *form view history* dan menyetujui pekerjaan masing-masing anggota timnya. Desain *form approve sub task* ini dapat dilihat pada Gambar 3.45.

Id Project

Nama Sub Task	Tgl Sub Task	Progress Work	App Work	Progress Test	App Test	Progress Train	App Train	Progress (dihara
ABC	xx/xx/xxxx-xx/xx/xxxx	<div style="width: 100%; height: 10px; background-color: #00aaff;"></div>	√	<div style="width: 100%; height: 10px; background-color: #00aaff;"></div>	√	<div style="width: 100%; height: 10px; background-color: #00aaff;"></div>	√	<div style="width: 100%; height: 10px; background-color: #00aaff;"></div>

Gambar 3.45 Desain *Form Approve Sub Task*

Fungsi objek pada *form approve sub task* adalah sebagai berikut:

Tabel 3.53 Fungsi Objek *Form Approve Sub Task*

Nama Objek	Tipe Objek	Fungsi
Id Project	<i>Textbox</i>	Tempat kode proyek
Tabel Sub Task	<i>Datagrid</i>	Melihat data <i>sub task</i>
Progress Work	<i>Progressbar</i>	Melihat persentasi <i>work</i>
App Work	<i>Check</i>	Menyetujui <i>work</i>
Progress Test	<i>Progressbar</i>	Melihat persentasi <i>test</i>
App Test	<i>Check</i>	Menyetujui <i>test</i>
Progress Train	<i>Progressbar</i>	Melihat persentasi <i>train</i>
App Train	<i>Check</i>	Menyetujui <i>train</i>

3.6.31 Desain *form view history*

Form view history ini digunakan oleh *project leader* untuk melihat kerja

anggota timnya. Desain *form view history* ini dapat dilihat pada Gambar 3.46.

Nama Sub Task

Nama User

Persentasi %

Tanggal

Keterangan

Edit Delete

Persentasi	Tanggal

Gambar 3.46 Desain *Form View History*

Fungsi objek pada *form view history* adalah sebagai berikut:

Tabel 3.54 Fungsi Objek *Form View History*

Nama Objek	Tipe Objek	Fungsi
Nama Sub Task	<i>Textbox</i>	Tempat nama <i>sub task</i>
Nama User	<i>Textbox</i>	Tempat nama <i>user</i>
Persentasi	<i>Textbox</i>	Tempat persentasi <i>work/ test/ train</i>
Tanggal	<i>Textbox</i>	Tempat tanggal persentasi
Keterangan	<i>Textbox</i>	Tempat keterangan
Edit	<i>Button</i>	Mengubah data <i>history</i>
Delete	<i>Button</i>	Menghapus data <i>history</i>
Tabel Persentasi	<i>Datagrid</i>	Melihat data <i>history</i>

3.6.32 Desain *form* hasil *project*

Form hasil *project* ini digunakan oleh *marketing* dan manajer untuk menyetujui atau membatalkan *project*. Desain *form* hasil *project* ini dapat dilihat pada Gambar 3.47.

The image shows a user interface for a 'Form Hasil Project'. On the left, there is a decorative logo consisting of several overlapping triangles in shades of red and white. To the right of the logo, the form contains the following elements:

- Id Hasil Project**: A single-line text input field.
- Id Project**: A single-line text input field.
- Status Project**: A single-line text input field.
- Tanggal Status**: A single-line text input field.
- Keterangan**: A larger, multi-line text input area.
- Save**: A rectangular button located below the 'Keterangan' field.

A large, semi-transparent watermark for 'UNIVERSITAS Dinamika' is visible across the right side of the form.

Gambar 3.47 Desain *Form* Hasil *Project*

Fungsi objek pada *form* hasil *project* adalah sebagai berikut:

Tabel 3.55 Fungsi Objek *Form Hasil Project*

Nama Objek	Tipe Objek	Fungsi
Id Hasil Project	<i>Label</i>	Tempat kode hasil proyek
Id Project	<i>Textbox</i>	Tempat kode proyek
Status Project	<i>Textbox</i>	Tempat status proyek
Tanggal Status	<i>Textbox</i>	Tempat tanggal hasil proyek
Keterangan	<i>Textbox</i>	Mengisi keterangan hasil proyek
Save	<i>Button</i>	Menyimpan data hasil proyek

3.6.33 Desain *form* laporan

Form laporan ini digunakan oleh manajer untuk melihat laporan-laporan yang berhubungan dengan proyek dari sisi biaya dan sumber daya *resource*.

Desain *form* laporan ini dapat dilihat pada Gambar 3.48.

Gambar 3.48 Desain *Form* Laporan

Fungsi objek pada *form* laporan adalah sebagai berikut:

Tabel 3.56 Fungsi Objek *Form* Laporan

Nama Objek	Tipe Objek	Fungsi
Jenis Laporan	<i>Combobox</i>	Mencari jenis laporan
Nama Proyek	<i>Combobox</i>	Mencari nama proyek
Ok	<i>Button</i>	Menampilkan laporan
Tahun	<i>Combobox</i>	Memilih tahun
Laporan	<i>Crystal Report</i>	Tempat Report

3.7 Desain Uji Coba

Setelah melakukan desain sistem, tahap selanjutnya adalah melakukan desain uji coba. Tujuan dari desain uji coba ini adalah untuk mengetahui apakah aplikasi yang telah dibuat ini sudah berjalan dengan baik dan sesuai dengan tujuan atau *output* yang diharapkan. Desain uji coba ini akan dilakukan dengan menggunakan metode *black box testing*.

Skenario yang akan dilakukan sebagai berikut, project yang dibuat bernama Sistem Informasi Akuntansi PT. XYZ, konsumen bernama Andrew dan dimulai pada tanggal 2 Agustus 2010 hingga 14 November 2010. Manajer kemudian membuat biaya dari proyek tersebut dengan beberapa parameter yaitu analisis kebutuhan, prototipe awal, programming yang meminimalkan error, memeriksa desain, testing berbasis *black box*, pembersihan *bug*, dan biaya garansi. Setelah biaya didapat, maka *market* dapat membuat surat kontrak yang diberikan kepada konsumen.

Selanjutnya, manajer menentukan pemimpin proyek dan pemimpin proyek akan membentuk tim yang berisikan anggota-anggota untuk menyelesaikan proyek tersebut. Pemimpin proyek juga membuat susunan aktivitas yang nantinya dikerjakan oleh anggota tim. Pada proyek ini, pemimpin proyeknya adalah Ali dan anggota timnya adalah Eric dan Sigit. Aktivitas yang dikerjakan nantinya adalah membuat master barang, membuat transaksi jual beli, membuat laporan neraca dan membuat laporan penutupan.

Setelah aktivitas selesai dibuat, maka manajer akan menyetujui urutan kerja. Aktivitas yang sudah disetujui akan dikerjakan oleh masing-masing user dan hasil dari kerja itu akan diinputkan user ke dalam sistem. Hasil kerja itu akan

dilihat oleh pimpinan proyek, dan jika tidak setuju, project leader dapat merubahnya sendiri. Pada proyek ini, inputan diberikan kepada aktivitas membuat master barang. *User* akan memasukkan persentasi 15 persen yang kemudian dicoba diubah menjadi 10 persen. Pimpinan proyek yang melihat akan mengubah persentasi menjadi 5 persen dikarenakan desain yang ada masih kacau.

Setelah semua proyek selesai dikerjakan, maka manajer akan menutup proyek dan sistem akan menghasilkan laporan penutupan proyek. Adapun uji coba yang akan dilakukan adalah sebagai berikut:

3.7.1 Desain uji coba membuat proyek baru

A. Desain uji coba memasuki sistem

Seorang user dapat masuk ke dalam ketika memasuki *form login*. Di *form login*, seorang *user* harus menginputkan id dan *password* pada. Uji coba yang akan dilakukan pada *form login* dapat dilihat pada Tabel 3.58.

Tabel 3.57 Data User

Nama Field	Data 1	Data 2
Id User	001	002
Password	admin	1234

Tabel 3.58 Desain Uji Coba *Form Login*

Test Case	Tujuan	Input	Output Yang Diharapkan
1	Mengenali <i>user</i> memasukkan id <i>user</i> dan <i>password</i> yang sesuai	Id <i>user</i> dan <i>password</i> yang valid	<i>User</i> masuk ke dalam sistem
2	Mengenali <i>user</i> memasukkan id <i>user</i> dan <i>password</i> yang tidak sesuai	Id <i>user</i> dan <i>password</i> yang tidak valid	Muncul pesan peringatan

B. Desain uji coba membuat proyek baru

Setelah masuk ke dalam sistem, seorang market dapat membuat proyek baru pada form *master project*. Uji coba yang akan dilakukan pada *form master project* dapat dilihat pada Tabel 3.60.

Tabel 3.59 Data Proyek

Nama Field	Data 1	Data 2
Nama Project	SIA PT. XYZ	SIM PT. ABC
Nama Customer	Andrew	Ahmad
Tgl Mulai Project	02-08-2010	01-04-2011
Tgl Selesai Project	14-11-2010	30-06-2010
Keterangan Project	Membutuhkan neraca, laporan penutupan yang bagus	Membutuhkan sistem manajemen yang baik

Tabel 3.60 Desain Uji Coba *Form Master Project*

Test Case	Tujuan	Input	Output Yang Diharapkan
3	Sistem tidak dapat dikelabui saat memasukkan nilai tanggal selesai yang lebih dulu dari nilai tanggal awal	Memasukkan nilai tanggal selesai yang lebih dulu dari nilai tanggal awal	Nilai tanggal selesai berubah menjadi tanggal awal
4	Menyimpan data <i>project</i>	Data <i>project</i> , tombol <i>save</i>	Project baru terbuat
5	Mengubah data <i>project</i>	Data <i>project</i> , tombol <i>edit</i>	Project berubah sesuai dengan perubahan yang dilakukan
6	Mengosongkan semua inputan	Tombol <i>reset</i>	Semua inputan menjadi kosong

C. Desain uji coba membuat biaya proyek

Setelah proyek dibuat, manajer lalu menghitung estimasi biaya dengan metode Cost of Quality yang berdasarkan pada beberapa parameter dalam 4 kategori. Uji coba yang akan dilakukan pada *form master cost* dapat dilihat pada Tabel 3.62.

Tabel 3.61 Data Biaya

Nama Field	Data 1	Data 2
Requirements Analysis	12.000.000	14.000.000
Early Prototyping	5.000.000	0
Defensive Programming	15.000.000	22.000.000
Design Review	2.500.000	4.000.000
Black box Testing	1.500.000	2.000.000
Bug fixes	6.000.000	6.500.000
Techinal Support Call	0	4.900.000
Warrantly	25.200.000	32.000.000

Tabel 3.62 Desain Uji Coba *Form Master Cost*

Test Case	Tujuan	Input	Output Yang Diharapkan
7	Mendapatkan estimasi biaya	Memilih parameter, menginputkan data biaya dan menekan tombol save	Estimasi biaya diketahui

Estimasi biaya dari program lalu akan dibandingkan dengan estimasi biaya yang didapat dari sistem lama SSI yaitu estimasi biaya tanpa adanya metode *Cost of Quality*. Desain uji coba perbedaan dari estimasi biaya dengan metode dan tanpa metode akan dapat dilihat pada Tabel 3.63.

Tabel 3.63 Desain Uji Coba Perbandingan Dengan Dan Tanpa Metode

Estimasi Biaya dengan CoQ	Estimasi Biaya tanpa CoQ	Selisih
Xxx	yyy	xxx-yyy

D. Desain uji coba menentukan pemimpin proyek

Setelah sebuah proyek mendapatkan persetujuan dari pelanggan dengan keluarnya surat kontrak, maka manajer akan menentukan project leader (pimpinan proyek) pada setiap proyek yang telah disetujui. Langkah ini dilakukan pada *form resource level*. Uji coba yang akan dilakukan pada *form resource level* dapat dilihat pada Tabel 3.65.

Tabel 3.64 Data *Resource Level*

Nama Field	Data 1	Data 2
Nama Proyek	SIA PT. XYZ	SIM PT. ABC
Nama Pemimpin Proyek	Ali	Samuel

Tabel 3.65 Desain Uji Coba *Form Resource Level*

Test Case	Tujuan	Input	Output Yang Diharapkan
8	Menentukan <i>project leader</i>	Nama proyek dan nama pemimpin proyek	<i>Project leader</i> berhasil ditetapkan dan pada <i>project overview</i> dari <i>project leader</i> akan muncul <i>project</i> tersebut

3.7.2 Desain uji coba merancang proyek

A. Desain uji coba menentukan anggota proyek

Setelah ditentukan oleh manajer, seorang project leader dapat memilih anggota timnya yang akan membantunya mengerjakan sebuah proyek. Langkah ini dilakukan pada *form resource level* dimana seorang project leader tidak dapat memilih dirinya sendiri. Uji coba yang akan dilakukan pada *form resource level* dapat dilihat pada Tabel 3.67.

Tabel 3.66 Data *Resource Level*

Nama Field	Data 1	Data 2
Nama Proyek	SIA PT. XYZ	SIA PT. XYZ
Nama <i>Resource</i>	Eric	Sigit
Nama <i>Level</i>	Programmer	Trainer

Tabel 3.67 Desain Uji Coba *Form Resource Level*

Test Case	Tujuan	Input	Output Yang Diharapkan
1	Menentukan <i>resource project</i>	Nama proyek, nama <i>resource</i> dan nama <i>level</i>	<i>Resource project</i> berhasil ditambahkan dan pada <i>project overview</i> dari <i>user</i> akan muncul <i>project</i> tersebut

B. Desain uji coba *master sub task*

Dalam mengurutkan aktivitas (*sub task*), ada beberapa hal dapat diperhatikan, salah satunya adalah setiap sub task (kecuali awal dan terakhir), harus memiliki minimal satu predesesor dan satu suksesor. Uji coba yang akan dilakukan pada *form master sub task* dapat dilihat pada Tabel 3.69.

Tabel 3.68 Data Jalur

Contoh Jalur	Gambar	Keterangan
Jalur 1	<pre> graph LR A[A] --> B[B] B --> C[C] B --> D[D] </pre>	Memiliki 2 akhir
Jalur 2	<pre> graph LR A[A] --> C[C] B[B] --> C[C] C --> D[D] </pre>	Memiliki 2 awal
Jalur 3	<pre> graph LR A[A] --> C[C] B[B] </pre>	Tidak Tersambung

Keterangan:

A: Aktivitas membuat master barang (Durasi: 7 hari)

B: Aktivitas membuat transaksi jual beli (Durasi: 5 hari)

C: Aktivitas membuat laporan neraca (Durasi: 10 hari)

D: Aktivitas membuat laporan penutupan (Durasi: 3 hari)

Tabel 3.69 Desain Uji Coba *Form Master Sub Task*

Test Case	Tujuan	Input	Output Yang Diharapkan
2	Mencegah aktivitas yang memiliki 2 akhir	Jalur 1	Memunculkan error dengan keterangan aktivitas memiliki 2 akhir
3	Mencegah aktivitas yang memiliki 2 awal	Jalur 2	Memunculkan error dengan keterangan aktivitas memiliki 2 awal
4	Mencegah aktivitas yang tidak memiliki awal maupun akhir	Jalur 3	Memunculkan error dengan keterangan aktivitas yang tidak jelas

C. Desain uji coba menampilkan *critical path*

Setelah *sub task* selesai disusun, maka seorang *project leader* atau manajer dapat melihat *critical path* pada *form summary*. Uji coba yang akan dilakukan pada *form summary* dapat dilihat pada Tabel 3.71.

Tabel 3.70 Data Jalur Normal

Contoh Jalur	Gambar	Durasi (Hari)
Jalur 1	<pre> graph LR A[A] --> B[B] A --> C[C] B --> D[D] C --> D[D] </pre>	A: 2, B: 5, C: 4, D: 2

Keterangan:

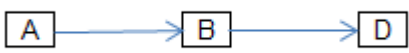
A: Aktivitas membuat master barang (Predesesor: tidak ada)

B: Aktivitas membuat transaksi jual beli (Predesesor: Aktivitas A)

C: Aktivitas membuat laporan neraca (Predesesor: Aktivitas A)

D: Aktivitas membuat laporan penutupan (Predesesor: Aktivitas B dan C)

Tabel 3.71 Desain Uji Coba *Form Summary*

Test Case	Tujuan	Input	Output Yang Diharapkan
5	Menampilkan gantt chart	Jalur 1	

3.7.3 Desain uji coba mengimplementasikan proyek

A. Desain uji coba memasukkan persentasi

Project yang telah disetujui oleh manajer kemudian dikerjakan oleh *user*.

Seorang *user* dapat menginputkan persentasi pekerjaannya dalam sistem dan akan disetujui oleh *project leader*. Langkah ini dilakukan pada form *history work*, *test*,

atau *train*. Uji coba yang akan dilakukan pada form *history work* dapat dilihat pada Tabel 3.73.

Tabel 3.72 Data *History*

Nama Field	Data 1	Data 2
Nama Sub Task	Membuat master barang	Membuat master barang
Tanggal	02-05-2010	05-05-2010
Persentasi	15	10
Keterangan	Desain selesai	Pencarian selesai

Tabel 3.73 Desain Uji Coba *Form History Work*

Test Case	Tujuan	Input	Output Yang Diharapkan
1	Mencegah <i>user</i> memasukkan persentasi dibawah nilai yang sudah ada	Data <i>history</i>	Muncul <i>error</i> akibat persentasi yang diinputkan tidak cocok

B. Desain uji coba mengontrol proyek

Pekerjaan user yang selesai dapat dikontrol oleh *project leader* salah satunya adalah memeriksa apakah pekerjaan user tersebut betul-betul sudah selesai. Langkah ini dilakukan pada *form history*. Uji coba yang akan dilakukan pada *form history* dapat dilihat pada Tabel 3.75.

Tabel 3.74 Data *History*

Nama Field	Data 1
Nama Sub Task	Membuat master barang
Tanggal	12-05-2010
Persentasi	5
Keterangan	Desain masih kacau

Tabel 3.75 Desain Uji Coba *Form History*

Test Case	Tujuan	Input	Output Yang Diharapkan
2	Mengubah data <i>user</i>	Data history	Data berubah dan pada project team, data tersebut dapat terlihat
3	Menghapus data <i>user</i>	Tombol Delete	Data terhapus dalam sistem, namun anggota proyek tidak dapat melihatnya.

3.7.4 Desain uji coba menutup proyek

A. Desain uji coba menutup proyek

Setelah semua *sub task* selesai ditraining kepada konsumen dan disetujui oleh project leader, langkah terakhir adalah menutup proyek. Langkah ini hanya dapat dilakukan oleh manajer dan dilakukan pada *form* hasil proyek. Uji coba yang akan dilakukan pada *form* hasil proyek dapat dilihat pada Tabel 3.77.

Tabel 3.76 Data Hasil Proyek

Nama Field	Data 1	Data 2
Nama Proyek	SIA PT. XYZ	SIM PT. ABC
Tanggal	20-05-2011	22-05-2011
Status	Completed	Completed
Keterangan	Konsumen puas	Akan dipikirkan kemungkinan membuat perkembangan proyek

Tabel 3.77 Desain Uji Coba *Form* Hasil Proyek

Test Case	Tujuan	Input	Output Yang Diharapkan
4	Menutup proyek	Data hasil proyek	Proyek berhasil ditutup dan memunculkan laporan penutupan

BAB IV

IMPLEMENTASI DAN EVALUASI

4.1 Kebutuhan Sistem

Untuk dapat menjalankan sistem yang dibuat ini diperlukan perangkat keras dan perangkat lunak dengan spesifikasi tertentu. Adapun kebutuhan perangkat keras dan perangkat lunak untuk sistem ini adalah sebagai berikut:

4.1.1 Kebutuhan perangkat keras

Sistem yang dibuat ini akan ditempatkan pada sebuah *server*. Kebutuhan minimal perangkat keras yang harus dipenuhi untuk *server* agar sistem berjalan dengan baik dan lancar adalah sebagai berikut:

- a. Processor Intel Pentium IV dengan kecepatan 1.5 GHz atau lebih.
- b. *Memory* 512 Mb atau lebih.
- c. *Harddisk* 10 Gb atau lebih.
- d. Koneksi jaringan yang bisa menghubungkan komputer *server* dengan komputer *client*.
- e. Monitor, *mouse*, *keyboard* yang berjalan dengan baik.

Kebutuhan minimal ini diperlukan untuk kebutuhan instalasi *Microsoft SQL Server* sebagai *database* pada *server*. Sedangkan kebutuhan minimal untuk *client* untuk instalasi aplikasi ini adalah sebagai berikut:

- a. Processor Intel Pentium IV dengan kecepatan 1 GHz atau lebih.
- b. *Memory* 1 Gb atau lebih.
- c. *Harddisk* 15 Gb atau lebih.

- d. Koneksi jaringan yang bisa menghubungkan komputer *client* dengan komputer *server*.
- e. Monitor dengan resolusi 1280 x 800.
- f. *Mouse* dan *keyboard* yang berjalan dengan baik.

4.1.2 Kebutuhan perangkat lunak

Kebutuhan perangkat lunak yang harus ada pada *server* agar sistem dapat berjalan dengan baik adalah sebagai berikut:

- a. Sistem operasi menggunakan *Windows XP Professional Service Pack* tiga atau lebih.
- b. *Database* untuk pengolahan data menggunakan *Microsoft SQL Server 2005* dengan *SQL Browser* yang berjalan dengan baik.
- c. *Microsoft .NET Framework 2.0* atau di atasnya.

Sedangkan kebutuhan perangkat lunak untuk *client* agar sistem dapat berjalan dengan baik adalah sebagai berikut:

- a. Sistem operasi menggunakan *Windows XP Professional Service Pack* tiga atau lebih.
- b. *Database* untuk pengolahan data menggunakan *Microsoft SQL Server 2005* dengan *SQL Browser* yang berjalan dengan baik.
- c. *Microsoft .NET Framework 2.0* atau di atasnya.
- d. *DevExpress 9*.

4.2 Pembuatan Program

Program atau aplikasi ini dibuat dengan menggunakan bahasa pemrograman VB.NET. Alasan penggunaan VB.NET sebagai bahasa

pemrograman adalah karena mudah dipelajari, mudah digunakan dan mudah diimplementasikan. Dalam pembuatan kode program, perangkat lunak yang digunakan adalah *Microsoft Visual Studio 2005*. Selain itu, aplikasi ini juga menggunakan *Microsoft SQL Server* untuk keperluan basis datanya. *Microsoft SQL Server* dipilih sebagai aplikasi basis data karena keamanannya yang baik, mampu menangani data yang besar, dan cocok dipasangkan dengan *Microsoft Visual Studio*. Karena bahasa pemrograman yang digunakan adalah VB.NET, maka ekstensi *file* yang dihasilkan adalah *.exe* sehingga dapat langsung dijalankan.

4.3 Implementasi Sistem

Setelah kebutuhan perangkat keras dan perangkat lunak telah terpenuhi, maka tahap selanjutnya adalah melakukan implementasi sistem yang telah dibuat.

4.3.1 Menu Utama Sistem

Menu utama adalah *form* yang pertama kali muncul ketika program dijalankan. Untuk dapat masuk ke dalam sistem, *user* harus melakukan validasi terlebih dahulu pada *form login*. Untuk lebih jelasnya, *form* menu utama dapat dilihat pada Gambar 4.1.



Gambar 4.1 *Form* Menu Utama

4.3.2 Form Login

Form login ini digunakan untuk melakukan validasi terhadap *user* sebelum masuk ke dalam sistem yang dibuat. Pada *form* ini, *user* harus memasukkan id *user* pada kolom id *user* dan *password* pada kolom *password*. Jika *user* memasukkan id *user* dan *password* yang tepat, maka *user* akan masuk ke *form* sesuai jabatannya. Jika salah, maka akan keluar pesan peringatan. Untuk lebih jelasnya, *form login* dapat dilihat pada Gambar 4.2.



Gambar 4.2 *Form Login*

4.3.3 Form master user

Form master user ini digunakan oleh *administrator* untuk menyimpan dan mengubah *user* / status *user*. Untuk membuat *user* baru, *administrator* dapat memasukkan data *user* pada kolom NIK / id *user*, *user name*, *password*, *re-type password* dan pada *job* dari *user* yang dibuat kemudian menekan tombol *save*. Untuk mengubah *user*, *administrator* dapat memilih *user* yang ingin diubah kemudian menekan tombol *edit*. Untuk mengubah status *user*, *administrator* dapat mencentang status *user*. *Form master user* ini dapat dilihat pada Gambar 4.3.

Master User

NIK

User Name

Password

Re-type Password

Job

Drag a column header here to group by that column

	Id User	Nama User	Job User	Status ... Δ
>	001	Admin	Administrator	<input checked="" type="checkbox"/>
	002	Manajer	Manager	<input checked="" type="checkbox"/>
	003	Nia	Market	<input checked="" type="checkbox"/>
	020393	Tan Amelia	User	<input checked="" type="checkbox"/>
	060622	Nia Chayanisari	User	<input checked="" type="checkbox"/>
	300	Buyung	User	<input checked="" type="checkbox"/>
	303	Bobby	User	<input checked="" type="checkbox"/>
	305	Hendra	User	<input checked="" type="checkbox"/>
	890026	Pantiawati S...	User	<input checked="" type="checkbox"/>

Gambar 4.3 Form Master User

4.3.4 Form menu utama market

Form menu utama market ini digunakan oleh marketing untuk melihat project, memasuki form master project, form master sk, form master cost, dan form hasil project. Untuk memasuki form master project, marketing dapat menekan tombol master project. Untuk memasuki form master sk, marketing dapat menekan link “Tambahkan SK”. Untuk memasuki form master cost, marketing dapat menekan link “Lihat Cost”. Untuk dapat memasuki form hasil project, marketing dapat menekan link “Cancel Project”. Untuk lebih jelasnya, form menu utama market ini dapat dilihat pada Gambar 4.4.

Selamat Datang, Nia

Master Project In Progress

Nama Project	Manajer ...	No. SK	Status	Cost	Cancel	Keterangan
> SI Perpustakaan Universitas 45 Surabaya	<input checked="" type="checkbox"/>	SK-2404113	In Progress	Lihat Cost	Cancel Project	Keterangan
Sistem Informasi Perguruan Tinggi Maju Terus	<input type="checkbox"/>	Tambahkan SK	In Progress	Lihat Cost	Cancel Project	Keterangan
Sistem Informasi Rumah Sakit Mitra Keluarga	<input checked="" type="checkbox"/>	Tambahkan SK	In Progress	Lihat Cost	Cancel Project	Keterangan
Sistem Informasi Perguruan Tinggi STIE Wi...	<input checked="" type="checkbox"/>	SK-2205112	In Progress	Lihat Cost	Cancel Project	Keterangan
Sistem Informasi Perpustakaan STIKOM	<input checked="" type="checkbox"/>	Tambahkan SK	In Progress	Lihat Cost	Cancel Project	Keterangan

Gambar 4.4 Form Menu Utama Market

4.3.5 Form master kota

Form master kota ini digunakan oleh marketing untuk menambah dan mengubah data kota. Untuk menambah kota, marketing dapat memasukkan data kota ke dalam inputan yang tersedia kemudian menekan tombol *save*. Untuk mengubah, marketing dapat memilih kota kemudian menekan tombol *edit*. Untuk lebih jelasnya, form master kota ini dapat dilihat pada Gambar 4.5.

Master Kota

Kode Kota

Nama Kota

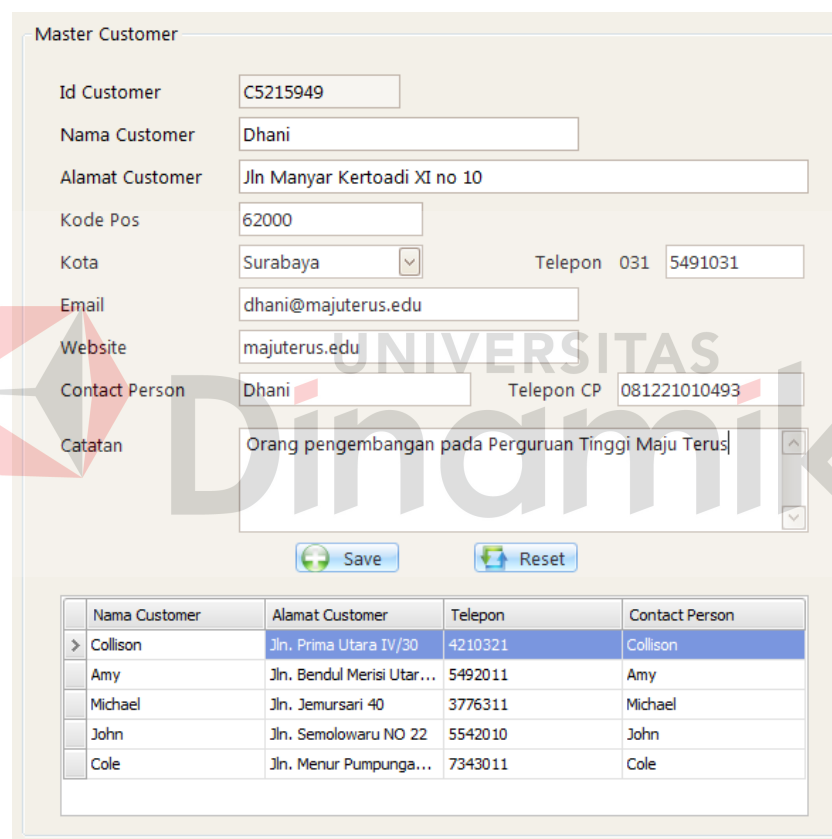
Kode Telp

Kode Kota	Nama Kota	Kode Telp
> C202	Jakarta	021
C211	Surabaya	031
C213	Bandung	022

Gambar 4.5 Form Master Kota

4.3.6 Form master customer

Form master customer ini digunakan oleh *marketing* untuk menambah atau mengubah data *customer*. Untuk menambah *customer*, *marketing* dapat memasukkan data *customer* dan menekan tombol *save*. Jika ingin mengubahnya, *marketing* dapat memilih *customer* yang ingin diubah kemudian menekan tombol *edit*. Untuk lebih jelasnya, *form master customer* ini dapat dilihat pada Gambar 4.6.



Master Customer

Id Customer: C5215949

Nama Customer: Dhani

Alamat Customer: Jln Manyar Kertoadi XI no 10

Kode Pos: 62000

Kota: Surabaya

Telepon 031: 5491031

Email: dhani@majuterus.edu

Website: majuterus.edu

Contact Person: Dhani

Telepon CP: 081221010493

Catatan: Orang pengembangan pada Perguruan Tinggi Maju Terus

Save Reset

	Nama Customer	Alamat Customer	Telepon	Contact Person
>	Collison	Jln. Prima Utara IV/30	4210321	Collison
	Amy	Jln. Bendul Merisi Utar...	5492011	Amy
	Michael	Jln. Jemursari 40	3776311	Michael
	John	Jln. Semolowaru NO 22	5542010	John
	Cole	Jln. Menur Pumpunga...	7343011	Cole

Gambar 4.6 Form Master Customer

4.3.7 Form master project

Form master project ini digunakan oleh *marketing* untuk menambah dan mengubah data *project*. Untuk menambah *project*, *marketing* dapat memasukkan data *project*, memilih nama *customer*, dan menekan tombol *save*. Jika ingin mengubah data *project*, *marketing* dapat memilih *project* yang ingin diubah dan

menekan tombol *edit*. Untuk lebih jelasnya, *form master project* ini dapat dilihat pada Gambar 4.7.

Nama Project	Nama Customer	Tanggal Mulai	Tanggal Selesai
> SI Perpustakaan ...	Collison	5/1/2011	6/17/2011
Sistem Informasi ...	Cole	4/1/2011	5/24/2011
Sistem Informasi ...	John	6/1/2011	9/30/2011
Sistem Informasi ...	Michael	6/1/2011	6/30/2011
Sistem Informasi ...	Michael	5/30/2011	7/31/2011

Gambar 4.7 *Form Master Project*

4.3.8 *Form pencarian customer*

Form pencarian customer digunakan oleh *marketing* untuk mencari nama *customer* yang akan ditambahkan dalam *master project*. Untuk lebih jelasnya, *form pencarian customer* ini dapat dilihat pada Gambar 4.8.

Id Customer	Nama Customer
> C0201855	Collison
C0373754	Amy
C4775739	Michael
C5215949	Dhani
C6615918	John
C7615023	Cole

Gambar 4.8 *Form Pencarian Customer*

4.3.9 Form surat kontrak

Form surat kontrak ini digunakan oleh *marketing* untuk memasukkan isi surat kontrak yang sudah disetujui. Untuk lebih jelasnya, *form* surat kontrak ini dapat dilihat pada Gambar 4.9.

The image shows a screenshot of a web-based form titled "Master SK". The form has a light beige background and contains the following fields:

- Id Surat Kontrak:** A text input field containing the value "S3823746".
- Id Project:** A text input field containing the value "P2420237".
- Tanggal Surat Kontrak:** A date input field containing "4/14/2010" and a small downward arrow icon.
- Nomor Surat Kontrak:** A text input field containing the value "SK-0414201003".
- Keterangan:** A text area containing the text "Berharap sistem seperti sicyca, sistem untuk AAK, Keuangan dan HRD".

At the bottom center of the form, there is a blue button with a green plus icon and the text "Save".

Gambar 4.9 Form Surat Kontrak

4.3.10 Form menu utama manajer

Form menu utama manajer ini digunakan manajer oleh melihat *project*, menyetujui *project*, memasuki *form master* biaya, *resource level*, *approve task*, dan hasil *project*. Untuk menyetujui *project*, manajer dapat mencentang status *approval*. Untuk memasuki *form master* biaya, manajer dapat menekan *link* "Input Cost" *master project*. Untuk memasuki *form resource level*, manajer dapat menekan *link* "Set Leader". Untuk memasuki *form approve task*, manajer dapat menekan *link* "Task & Sub Task". Untuk dapat memasuki *form* hasil *project*, manajer dapat menekan *link* "Cancel Project" atau *link* "Closing Project". Untuk lebih jelasnya, *form* menu utama manajer ini dapat dilihat pada Gambar 4.10.

Selamat Datang Manager!

In Progress

Nama Project	Approval	Status	Cost	Project Leader	Detail Project	Hasil	Keterangan
> SI Perpustakaan Universita...	<input checked="" type="checkbox"/>	In Progress	Input Cost	Set Leader	Task & Sub Task	Cancel Project	Keterangan
Sistem Informasi Perguruan...	<input type="checkbox"/>	In Progress	Input Cost	Set Leader	Task & Sub Task	Cancel Project	Keterangan
Sistem Informasi Rumah Sa...	<input checked="" type="checkbox"/>	In Progress	Input Cost	Set Leader	Task & Sub Task	Cancel Project	Keterangan
Sistem Informasi Perguruan...	<input checked="" type="checkbox"/>	In Progress	Input Cost	Set Leader	Task & Sub Task	Cancel Project	Keterangan
Sistem Informasi Perpustak...	<input checked="" type="checkbox"/>	In Progress	Input Cost	Set Leader	Task & Sub Task	Cancel Project	Keterangan

Gambar 4.10 Form Menu Utama Manajer

4.3.11 Form master kompetensi

Form master kompetensi ini digunakan oleh manajer untuk menambah atau mengubah data kompetensi. Untuk menambah kompetensi, manajer dapat memasukkan data kompetensi dan menekan tombol *save*. Jika ingin mengubah kompetensi, manajer dapat memilih kompetensi dan menekan tombol *edit*. Untuk lebih jelasnya, *form master* kompetensi ini dapat dilihat pada Gambar 4.11.

Master Kompetensi

Id Kompetensi: K4785362

Nama Kompetensi: PHP

Keterangan:

Nama Kompetensi
> Oracle
Pemrograman Berbasis Obyek
Pemrograman Visual
Access
SQL

Gambar 4.11 Form Master Kompetensi

4.3.12 *Form master parameter*

Form master parameter ini digunakan oleh manajer untuk menambah atau mengubah data parameter. Untuk menambah parameter, manajer dapat memasukkan data parameter dan menekan tombol *save*. Jika ingin mengubah parameter, manajer dapat memilih parameter dan menekan tombol *edit*. Untuk lebih jelasnya, *form master parameter* ini dapat dilihat pada Gambar 4.12.

Nama Parameter	Kategori Biaya	Biaya per MH
Staff Training	Biaya Pencegahan	5000
Requirements Analysis	Biaya Pencegahan	11000
Early Prototyping	Biaya Pencegahan	11000
Fault Tolerant Design	Biaya Pencegahan	8000
Defensive Programming	Biaya Pencegahan	11000

Gambar 4.12 *Form Master Parameter*

4.3.13 *Form master level*

Form master level ini digunakan oleh manajer untuk menambah atau mengubah data *level*. Untuk menambah *level*, manajer dapat memasukkan data *level* dan menekan tombol *save*. Jika ingin mengubah *level*, manajer dapat memilih *level* dan menekan tombol *edit*. Untuk lebih jelasnya, *form master level* ini dapat dilihat pada Gambar 4.13.

Master Level

Id Level: L8976112

Nama Level: Code Worker

Keterangan:

Save Reset

Nama Level
Designer Project
Tester
Database Administrator
Project Leader
Trainer

Gambar 4.13 *Form Master Level*

4.3.14 *Form master resource*

Form master resource ini digunakan oleh manajer untuk menambah atau mengubah data *resource*. Untuk menambah *resource*, manajer dapat memasukkan data *resource*, memilih nama *user*, memilih dan menambahkan kompetensi dan menekan tombol *save*. Jika ingin mengubah *resource*, manajer dapat memilih *resource* dan menekan tombol *edit*. Untuk lebih jelasnya, *form master resource*

ini dapat dilihat pada Gambar 4.14.

Master Resource

Id Resource: R5594511

Nama Resource: Pantjawati Sudarmaningtyas Find

Kompetensi: Add

Daftar Kompetensi

Nama Kompetensi	Action
Oracle	Delete
MySQL	Delete
Power Designer	Delete
PHP	Delete
UML	Delete

Save Reset

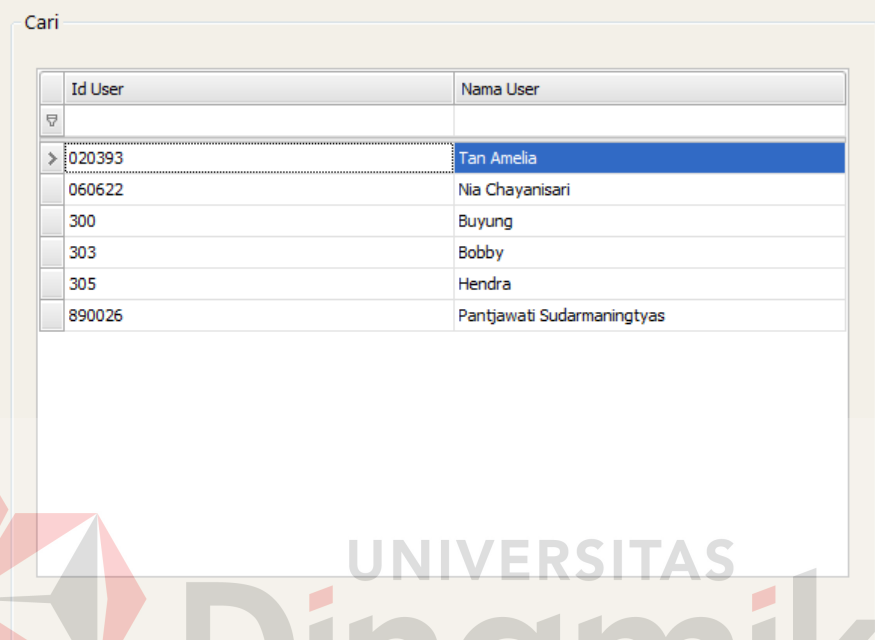
Drag a column header here to group by that column

Nama User	Nama Kompetensi
Tan Amelia	Visio
Kate	Access
Frank	Oracle
Tan Amelia	SQL

Gambar 4.14 *Form Master Resource*

4.3.15 Form pencarian user

Form pencarian user ini digunakan oleh manajer dan *project leader* untuk mencari nama *user* pada *form resource*, *resource level* dan *sub task*. Untuk lebih jelasnya, *form pencarian user* ini dapat dilihat pada Gambar 4.15.



Id User	Nama User
020393	Tan Amelia
060622	Nia Chayanisari
300	Buyung
303	Bobby
305	Hendra
890026	Pantjawati Sudarmaningtyas

Gambar 4.15 *Form Pencarian User*

4.3.16 Form master cost

Form master cost ini digunakan oleh manajer untuk memasukkan biaya *project* dan oleh *marketing* untuk melihat biaya. Untuk mengisi biaya, manajer dapat memilih 2 pilihan, mengisi *man* dan *hour* atau langsung mengisi total biaya setelah itu menekan tombol *save*. Untuk lebih jelasnya, *form master cost* ini dapat dilihat pada Gambar 4.16.

Master Cost

Nama Project:

Kategori:

Drag a column header here to group by that column

Nama Parameter	Biaya	Man	Hour	Total Biaya
> Defensive Programming	11000	5	680	37400000
Early Prototyping	11000	5	8	440000
Staff Training	5000			
Design Errors	11000	5	16	880000
Accurate Internal Documentation	3500			
Rework	11000	5	56	3080000
Technical Support Calls	11000	3	800	26400000
Lost Reputation	11000			
Refunds and Replacement with upd...	22000			
Warranty Cost	6600			60120000
Design Review	11000	1	80	880000
				160,320,000.00

Gambar 4.16 Form Master Cost

4.3.17 Form resource level

Form resource level ini digunakan oleh manajer dan *project leader* untuk membentuk tim yang akan mengerjakan proyek. Manajer akan memilih *project leader* dan *project leader* akan memilih *project team*. Untuk lebih jelasnya, *form resource level* ini dapat dilihat pada Gambar 4.17.

Resource Level

Id Resource Level: R8730312

Id Project: P2420237

Nama User: Tan Amelia

Level: Project Leader

Buttons: Find, Save

Drag a column header here to group by that column

Nama User	Nama Level
-----------	------------

Gambar 4.17 Form Resource Level

4.3.18 Form view and app task

Form view and app task ini digunakan oleh manajer untuk menyetujui *task* dan *sub task* yang ada dan melihat *gant chart*. *Project leader* menggunakannya untuk melihat *gant chart*. Untuk menyetujui *task* dan *sub task*, manajer dapat menekan tombol *approve task*. Jika ingin melihat *gant chart*, manajer atau *project leader* dapat menekan tombol *summary*. Untuk lebih jelasnya, *form view and app task* dapat dilihat pada Gambar 4.18.

View and App Task in Project Sistem Informasi Perguruan Tinggi Maju Terus

Project Leader : Tan Amelia

Drag a column header here to group by that column

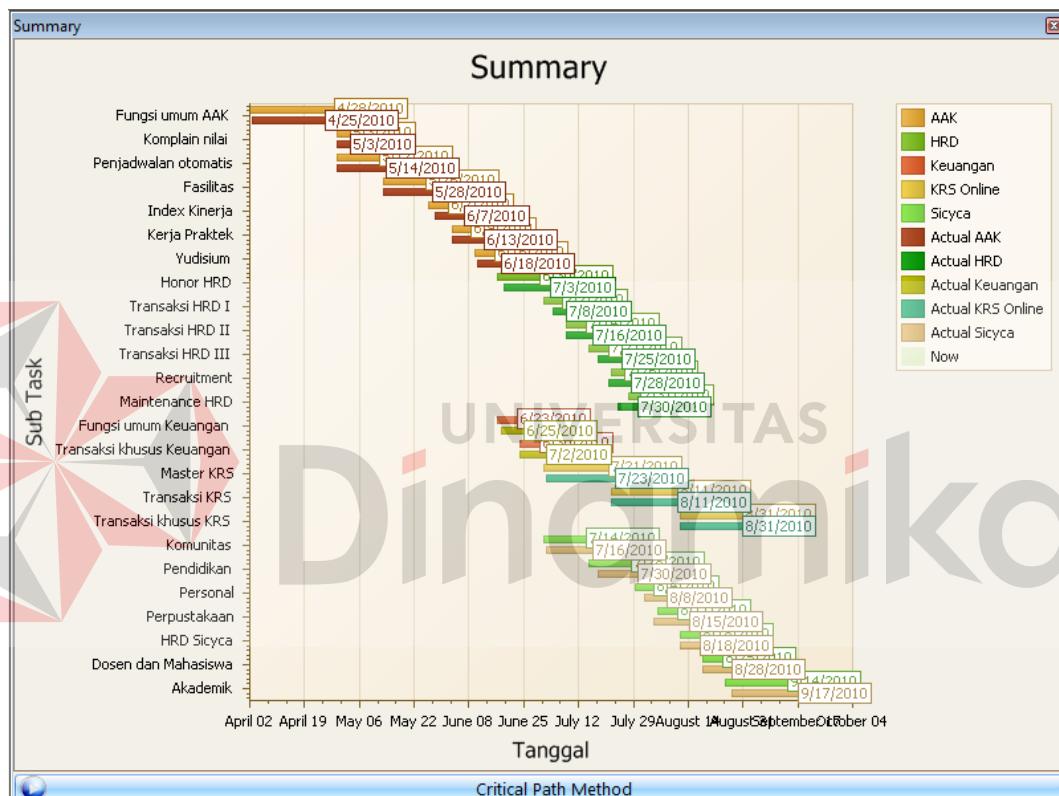
Nama Task	Sub Task	Tgl Mulai	Tgl Selesai	Act Mulai	Act Selesai	Project ...	Tester	Trainer	Biaya	Act Biaya	Predesor
> KRS Online	Membuat M...	4/15/2010	4/29/2010			Pantjawi ...	Pantjawi ...	Pantjawi ...	4396350	120753080	Lihat Predes...
KRS Online	Membuat T...	4/30/2010	5/9/2010			Pantjawi ...	Pantjawi ...	Pantjawi ...	2930900	116356730	Lihat Predes...
Sicyca Kapr...	Komunitas	5/6/2010	5/19/2010			Hendra	Pantjawi ...	Hendra	4103260	114598190	Lihat Predes...
KRS Online	Transaksi k...	5/10/2010	5/19/2010			Pantjawi ...	Pantjawi ...	Pantjawi ...	2930900	113425830	Lihat Predes...
Sicyca Mah...	Komunitas	5/11/2010	5/17/2010			Pantjawi ...	Pantjawi ...	Pantjawi ...	2051630	113132740	Lihat Predes...
Sicyca Mah...	Pendidikan	5/18/2010	5/31/2010			Pantjawi ...	Pantjawi ...	Pantjawi ...	4103260	111081110	Lihat Predes...
Sicyca Mah...	Personal	5/18/2010	6/6/2010			Pantjawi ...	Pantjawi ...	Pantjawi ...	5861800	111081110	Lihat Predes...
Sicyca Dos...	Komunitas	5/20/2010	6/2/2010			Nia Chayan...	Pantjawi ...	Nia Chayan...	4103260	110494930	Lihat Predes...
Sicyca Kapr...	Dosen	5/20/2010	5/26/2010			Hendra	Pantjawi ...	Hendra	2051630	110494930	Lihat Predes...
									160,320,2...	4,013,281,...	

Buttons: Approve Task, Summary

Gambar 4.18 Form View and App Task

4.3.19 Form summary

Form summary ini digunakan oleh manajer dan *project leader* untuk melihat *ganttt chart* dari sebuah *project* dan *ganttt chart critical path*. Untuk melihat *critical path*, manajer atau *project leader* dapat menekan tombol *critical path method*. Untuk lebih jelasnya, *form summary* ini dapat dilihat pada Gambar 4.19.



Gambar 4.19 *Form Summary*

4.3.20 Form review project

Form review project ini digunakan oleh *project leader* dan *project team* untuk mengetahui *project* mana yang akan dikerjakan. Untuk masuk ke dalam *taskboard*, *user* perlu memilih *project*. Jika status *user* pada *project* tersebut adalah *project leader*, maka *user* akan masuk ke dalam *project leader taskboard*, jika *tester*, maka akan masuk ke dalam *project tester taskboard*, jika *trainer* maka

akan masuk ke dalam *project trainer taskboard*, dan jika bukan ketiganya maka akan masuk ke *project team taskboard*. Untuk lebih jelasnya, *form review project* ini dapat dilihat pada Gambar 4.20.

Selamat Datang, Tan Amelia

Project dimana Anda terlibat :

Drag a column header here to group by that column

Nama Project	Status
> Sistem Informasi Perguruan Tinggi Maju Terus	Project Leader

Gambar 4.20 *Form Review Project*

4.3.21 *Form project leader taskboard*

Form project leader taskboard ini digunakan oleh *project leader* untuk membuat *task*, memasuki *form resource level*, *master sub task*, *approve sub task*, dan *view and app task*. Untuk membuat *task*, *project leader* dapat memasukkan data *task* ke dalam *form* dan menekan tombol *save*. Jika ingin mengubah *task*, *project leader* dapat memilih *task* dan menekan tombol *edit*. Untuk memasuki *form resource level*, *project leader* dapat menekan tombol *resource level*. Untuk memasuki *form sub task*, *project leader* dapat menekan link “Buat Sub Task”. Untuk memasuki *form approve sub task*, *project leader* dapat menekan tombol

Approve Sub Task. Untuk memasuki *form view and app task*, *project leader* dapat menekan tombol *View Summary and Cost*. Untuk lebih jelasnya, *form project leader taskboard* ini dapat dilihat pada Gambar 4.21.

Nama Task	Tgl Mulai	Tgl Selesai	Sub Task
> KRS Online	4/15/2010	5/19/2010	Buat Sub Task
Sicyca Kaprodi	5/6/2010	7/18/2010	Buat Sub Task
Sicyca Mahasiswa	5/11/2010	6/21/2010	Buat Sub Task
Sicyca Dosen da...	5/20/2010	7/12/2010	Buat Sub Task
AAK	5/28/2010	7/20/2010	Buat Sub Task
Sicyca Pimpinan	6/10/2010	8/3/2010	Buat Sub Task
Keuangan	7/9/2010	8/22/2010	Buat Sub Task
AAK Khusus	7/12/2010	8/16/2010	Buat Sub Task
HRD	7/16/2010	9/3/2010	Buat Sub Task
HRD Khusus	8/4/2010	9/1/2010	Buat Sub Task

Gambar 4.21 *Form Project Leader Taskboard*

4.3.22 *Form master sub task*

Form master sub task ini digunakan oleh *project leader* untuk membuat *sub task*, mengubah *sub task*, menambah predesesor, dan menghapus predesesor dari sebuah *sub task*. Untuk menambah *sub task*, *project leader* dapat memasukkan data *sub task*, memilih *user*, memilih *tester*, memilih *trainer*, memilih predesesor dan menekan tombol *save*. Untuk mengubah *sub task*, *project leader* dapat memilih *sub task* dan menekan tombol *edit*. Untuk menambahkan predesesor, *project leader* dapat menekan tombol *Add Predesesor*, dan untuk menghapus predesesor, *project leader* dapat menekan link “delete”. Untuk lebih jelasnya, *form master sub task* ini dapat dilihat pada Gambar 4.22.

Gambar 4.22 *Form Master Sub Task*

4.3.23 *Form pencarian predesesor sub task*

Form pencarian predesesor *sub task* ini digunakan oleh *project leader* untuk mencari kemungkinan *sub task* yang bisa menjadi predesesor dari *sub task* yang lain. Untuk lebih jelasnya, *form* pencarian predesesor *sub task* ini dapat dilihat pada Gambar 4.23.

Nama Sub Task	User	Tester	Trainer
> Membuat Master	Pantjawati Suda...	Pantjawati Suda...	Pantjawati Sudar...

Gambar 4.23 *Form Pencarian Predesesor Sub Task*

4.3.24 Form detail sub task

Form detail sub task ini digunakan oleh *project leader* untuk mendeskripsikan *sub task* menjadi lebih detail. Untuk lebih jelasnya, *form detail sub task* ini dapat dilihat pada Gambar 4.24.

Gambar 4.24 Form Detail Sub Task

4.3.25 Form project team taskboard

Form project team taskboard ini digunakan oleh *project team* untuk melihat pekerjaan yang harus dikerjakan dan memasuki *form history work*. Untuk memasuki *form history work*, *project team* dapat menekan *progress bar* yang ada. Untuk lebih jelasnya, *form project team taskboard* ini dapat dilihat pada Gambar 4.25.

Project Team Taskboard

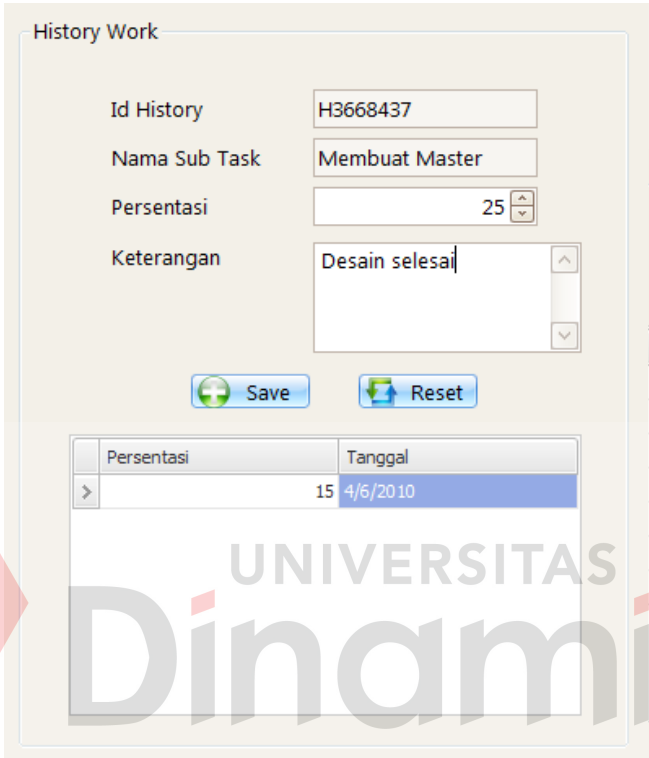
Id Project:

Nama Sub Task	Tgl Mulai	Tgl Selesai	Progress	Progress (diharap...	Finished	Predesesor
> Membuat Master	4/15/2010	4/29/2010	0%	0%	<input type="checkbox"/>	Lihat Predesesor
Membuat Transaksi	4/30/2010	5/9/2010	0%	0%	<input type="checkbox"/>	Lihat Predesesor
Transaksi khusus	5/10/2010	5/19/2010	0%	0%	<input type="checkbox"/>	Lihat Predesesor
Komunitas	5/11/2010	5/17/2010	0%	0%	<input type="checkbox"/>	Lihat Predesesor
Personal	5/18/2010	6/6/2010	0%	0%	<input type="checkbox"/>	Lihat Predesesor
Pendidikan	5/18/2010	5/31/2010	0%	0%	<input type="checkbox"/>	Lihat Predesesor
Perpustakaan	6/7/2010	6/21/2010	0%	0%	<input type="checkbox"/>	Lihat Predesesor

Gambar 4.25 Form Project Team Taskboard

4.3.26 Form history work

Form history work ini digunakan oleh *project team* untuk memasukkan inputan kerja. Untuk lebih jelasnya, *form history work* ini dapat dilihat pada Gambar 4.26.



Persentasi	Tanggal
15	4/6/2010

Gambar 4.26 Form History Work

4.3.27 Form project test taskboard

Form project test taskboard ini digunakan oleh *project test* untuk melihat pekerjaan yang harus dikerjakan dan memasuki *form history test*. Untuk memasuki *form history test*, *project test* dapat menekan *progress bar* yang ada. Untuk lebih jelasnya, *form project test taskboard* ini dapat dilihat pada Gambar 4.27.

Project Tester Taskboard

Id Project

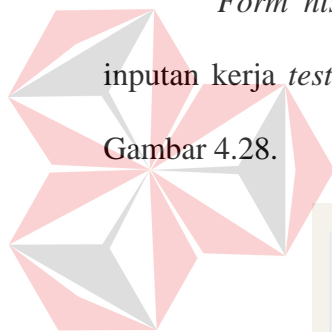
	Nama Sub Task	Tgl Mulai	Tgl Selesai	Approve Test	Progress	Progress (dihara...
>	Membuat Master	4/15/2010	4/29/2010	<input type="checkbox"/>	0%	0%
	Membuat Transaksi	4/30/2010	5/9/2010	<input type="checkbox"/>	0%	0%
	Komunitas	5/6/2010	5/19/2010	<input type="checkbox"/>	0%	0%
	Transaksi khusus	5/10/2010	5/19/2010	<input type="checkbox"/>	0%	0%
	Komunitas	5/11/2010	5/17/2010	<input type="checkbox"/>	0%	0%
	Pendidikan	5/18/2010	5/31/2010	<input type="checkbox"/>	0%	0%
	Personal	5/18/2010	6/6/2010	<input type="checkbox"/>	0%	0%
	Komunitas	5/20/2010	6/2/2010	<input type="checkbox"/>	0%	0%
	Dosen	5/20/2010	5/26/2010	<input type="checkbox"/>	0%	0%
	Mahasiswa	5/20/2010	5/29/2010	<input type="checkbox"/>	0%	0%

Gambar 4.27 Form Project Test Taskboard

4.3.28 Form history test

Form history test ini digunakan oleh *project test* untuk memasukkan inputan kerja *test*. Untuk lebih jelasnya, *form history test* ini dapat dilihat pada

Gambar 4.28.



History Test

Id History

Nama Sub Task

Persentasi

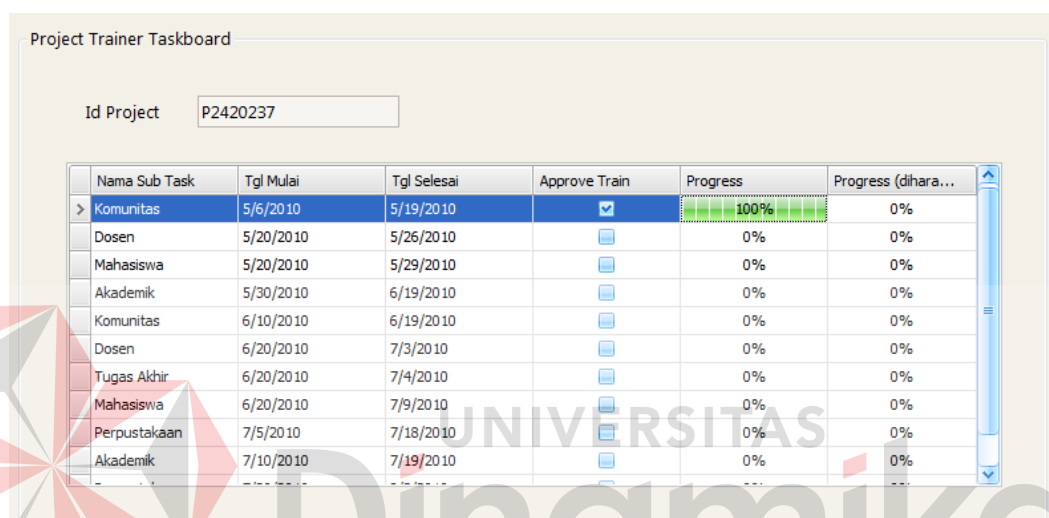
Keterangan

Persentasi	Tgl Testing

Gambar 4.28 Form History Test

4.3.29 Form project train taskboard

Form project train taskboard ini digunakan oleh *project train* untuk melihat pekerjaan yang harus dikerjakan dan memasuki *form history train*. Untuk memasuki *form history test*, *project test* dapat menekan *progress bar* yang ada. Untuk lebih jelasnya, *form project train taskboard* ini dapat dilihat pada Gambar 4.29.



Project Trainer Taskboard

Id Project: P2420237

Nama Sub Task	Tgl Mulai	Tgl Selesai	Approve Train	Progress	Progress (dihara...
> Komunitas	5/6/2010	5/19/2010	<input checked="" type="checkbox"/>	100%	0%
Dosen	5/20/2010	5/26/2010	<input type="checkbox"/>	0%	0%
Mahasiswa	5/20/2010	5/29/2010	<input type="checkbox"/>	0%	0%
Akademik	5/30/2010	6/19/2010	<input type="checkbox"/>	0%	0%
Komunitas	6/10/2010	6/19/2010	<input type="checkbox"/>	0%	0%
Dosen	6/20/2010	7/3/2010	<input type="checkbox"/>	0%	0%
Tugas Akhir	6/20/2010	7/4/2010	<input type="checkbox"/>	0%	0%
Mahasiswa	6/20/2010	7/9/2010	<input type="checkbox"/>	0%	0%
Perpustakaan	7/5/2010	7/18/2010	<input type="checkbox"/>	0%	0%
Akademik	7/10/2010	7/19/2010	<input type="checkbox"/>	0%	0%

Gambar 4.29 Form Project Train Taskboard

4.3.30 Form history train

Form history test ini digunakan oleh *project train* untuk memasukkan inputan kerja *train*. Untuk lebih jelasnya, *form history train* ini dapat dilihat pada Gambar 4.30.

Gambar 4.30 Form History Train

4.3.31 Form approve sub task

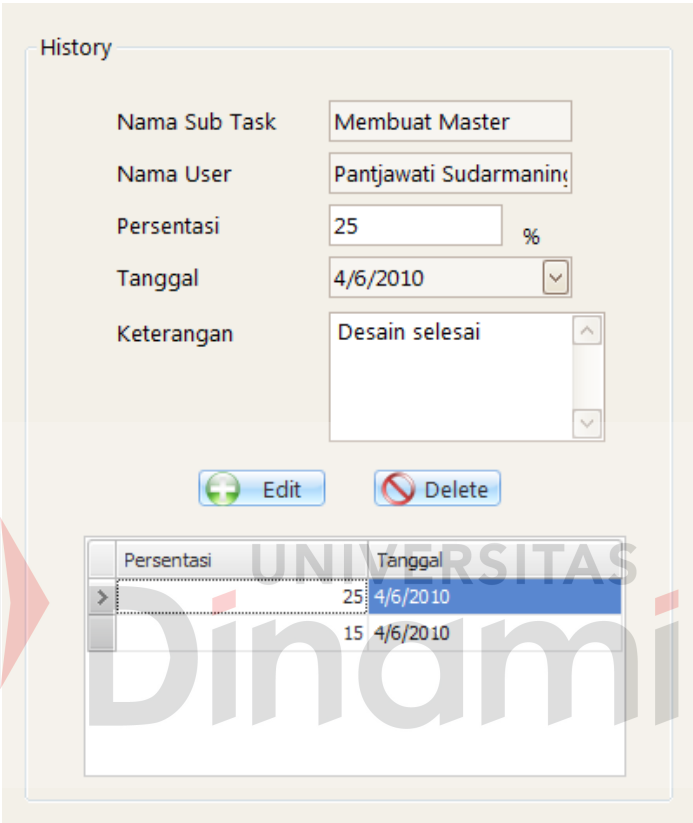
Form approve sub task ini digunakan oleh project leader untuk melihat pekerjaan yang harus dikerjakan dan memasuki form view history dan menyetujui pekerjaan masing-masing anggota timnya. Untuk memasuki form view history, project leader dapat menekan progress bar. Untuk menyetujui pekerjaan masing-masing anggota timnya, project leader dapat mencentang bagian App (App Work, App Test, atau App Train). Untuk lebih jelasnya, form approve sub task ini dapat dilihat pada Gambar 4.31.

Nama Sub Task	Tgl Sub Task	Progress W...	App ...	Progress Test	App ...	Progress Tr...	App ...	Progress (diharapkan)
Membuat Master	4/15/2010 - 4/29/2010	25%	<input type="checkbox"/>	0%	<input type="checkbox"/>	0%	<input type="checkbox"/>	0%
Membuat Transaksi	4/30/2010 - 5/9/2010	0%	<input type="checkbox"/>	0%	<input type="checkbox"/>	0%	<input type="checkbox"/>	0%
Komunitas	5/6/2010 - 5/19/2010	100%	<input type="checkbox"/>	0%	<input type="checkbox"/>	0%	<input type="checkbox"/>	0%
Transaksi khusus	5/10/2010 - 5/19/2010	0%	<input type="checkbox"/>	0%	<input type="checkbox"/>	0%	<input type="checkbox"/>	0%
Komunitas	5/11/2010 - 5/17/2010	0%	<input type="checkbox"/>	0%	<input type="checkbox"/>	0%	<input type="checkbox"/>	0%
Pendidikan	5/18/2010 - 5/31/2010	0%	<input type="checkbox"/>	0%	<input type="checkbox"/>	0%	<input type="checkbox"/>	0%
Personal	5/18/2010 - 6/6/2010	0%	<input type="checkbox"/>	0%	<input type="checkbox"/>	0%	<input type="checkbox"/>	0%
Komunitas	5/20/2010 - 6/2/2010	0%	<input type="checkbox"/>	0%	<input type="checkbox"/>	0%	<input type="checkbox"/>	0%
Dosen	5/20/2010 - 5/26/2010	0%	<input type="checkbox"/>	0%	<input type="checkbox"/>	0%	<input type="checkbox"/>	0%
Mahasiswa	5/20/2010 - 5/29/2010	0%	<input type="checkbox"/>	0%	<input type="checkbox"/>	0%	<input type="checkbox"/>	0%
Fungsi umum	5/28/2010 - 6/11/2010	0%	<input type="checkbox"/>	0%	<input type="checkbox"/>	0%	<input type="checkbox"/>	0%

Gambar 4.31 Form Project Approve Sub Task

4.3.32 *Form view history*

Form view history ini digunakan oleh *project leader* untuk melihat kerja anggota timnya. Untuk lebih jelasnya, *form view history* ini dapat dilihat pada Gambar 4.32.



Persentasi	Tanggal
25	4/6/2010
15	4/6/2010

Gambar 4.32 *Form View History*

4.3.33 *Form hasil project*

Form hasil project ini digunakan oleh *marketing* dan *manajer* untuk menyetujui atau membatalkan *project*. Untuk lebih jelasnya, *form hasil project* ini dapat dilihat pada Gambar 4.33.

Finishing/Canceling Project

Id Hasil Project: H4646118

Id Project: P2420237

Status Project: Completed

Tanggal Status: 9/25/2010

Keterangan: Konsumen puas dan berharap mengerjakan sistem yang lain

Save

Gambar 4.33 Form Hasil Project

4.3.34 Laporan penutupan *project*

Laporan penutupan dapat diakses ketika sebuah *project* dibatalkan oleh *marketing* atas permintaan *customer* dan manajer atau ditutup oleh manajer setelah *project* itu selesai dikerjakan. Laporan ini berisi data tentang *project*, status, tanggal, dan keterangan yang mendukung. Untuk lebih jelasnya, laporan penutupan *project* ini dapat dilihat pada Gambar 4.34.

Laporan Penutupan

Main Report

ssi Solusi Sistem Informasi
solusi sistem informasi

Solusi Sistem Informasi
Laporan Penutupan Project

Nama Project : Sistem Informasi Perguruan Tinggi Maju Terus

Nama Customer : Dhani

Tanggal Project : 4/2/2010 - 9/15/2010

Status : COMPLETED

Tanggal : 9/25/2010

Keterangan : Konsumen puas dan berharap mengerjakan sistem yang lain

Task yang dibuat :

- * Sicyca Mahasiswa
- * KRS Online
- * Sicyca Dosen dan Karyawan
- * Keuangan
- * ...

Gambar 4.34 Laporan Penutupan Project

4.3.35 Laporan *project cost*

Laporan *project cost* berisi data parameter biaya yang totalnya menjadi harga *project*. Karena sistem ini berdasarkan pada metode *Cost of Quality*, biaya tersebut digolongkan dalam 4 kategori biaya yaitu biaya pencegahan, biaya penilaian, biaya kegagalan internal dan biaya kegagalan eksternal. Untuk lebih jelasnya, laporan *project cost* ini dapat dilihat pada Gambar 4.35.

The screenshot shows a software interface for generating a 'Project Cost' report. The report title is 'Solusi Sistem Informasi Report Project Cost'. The project name is 'Sistem Informasi Perguruan Tinggi Maju Terus'. The report is categorized by 'Project Cost' and dated '2010'. The main report area contains a table with the following data:

Kategori Biaya	Nama Parameter	Man Hour	Biaya MH	Biaya
Biaya Kegagalan Eksternal				
Biaya Kegagalan Eksternal	Technical Support Calls	2.400	Rp 11.000	Rp 26.400.000
Biaya Kegagalan Eksternal	Warranty Cost	9.109	Rp 6.600	Rp 60.120.000
Biaya Kegagalan Eksternal		11.509	Rp	86.520.000
Biaya Kegagalan Internal				
Biaya Kegagalan Internal	Rework	280	Rp 11.000	Rp 3.080.000
Biaya Kegagalan Internal	Bug Fixes	40	Rp 11.000	Rp 440.000
Biaya Kegagalan Internal		320	Rp	3.520.000

Gambar 4.35 Laporan *Project Cost*

4.3.36 Laporan *project overview*

Laporan *project overview* berisi keterangan tentang *project* tersebut diantaranya nama *sub task*, tanggal *sub task*, tanggal aktual *sub task*, biaya *plan* dan biaya aktual digolongkan dalam nama *task*. Untuk lebih jelasnya, laporan *project overview* ini dapat dilihat pada Gambar 4.36.

REPORT

Jenis Report: Project Overview 2010

Nama Project: Sistem Informasi Perguruan Tin... OK

ssii Solusi Sistem Informasi
Laporan Project Overview

Nama Project : Sistem Informasi Perguruan Tinggi Maju Terus
Nama Project Leader : Amelia

Nama Sub Task	Tgl Mulai	Tgl Selesai	Actual Mulai	Actual Selesai	Biaya Plan	Biaya Actual
AAK						
Fungsi umum	05/28/2010	06/11/2010			Rp 4.396.335	Rp 35.463.769
Penjadwalan otomatis	06/12/2010	06/25/2010			Rp 4.103.246	Rp 31.067.434
Komplain nilai	06/26/2010	07/05/2010			Rp 2.930.890	Rp 26.964.188
Kerja Praktek	07/06/2010	07/20/2010			Rp 4.396.335	Rp 24.033.298
					Rp 15.826.806	Rp 17.528.689
AAK Khusus						
Fasilitas	07/12/2010	07/23/2010			Rp 3.517.068	Rp 22.274.764
Index Kinerja	07/24/2010	08/02/2010			Rp 2.930.890	Rp 18.757.696
Yudisium	08/03/2010	08/16/2010			Rp 4.103.246	Rp 15.826.806

Gambar 4.36 Laporan Project Overview

4.3.37 Laporan project

Laporan *project* berisi keterangan tentang biaya yang dihabiskan oleh masing-masing anggota tim yang bekerja dalam *project* tersebut. Untuk lebih jelasnya, laporan *project* ini dapat dilihat pada Gambar 4.37.

REPORT

Jenis Report: Project Resource Status Plan 2010

Nama Project: Sistem Informasi Perguruan Tin... OK

ssii Solusi Sistem Informasi
Laporan Project Plan

Nama Project : Sistem Informasi Perguruan Tinggi Maju Terus

Nama Resource	Job	Biaya
Pantjawati	Programmer, Trainer, Design Database, Tester	Rp 55.613.637,75
Sudarmaningtvas	Programmer, Trainer	Rp 21.981.675,00
Nia Chayanisan	Programmer, Trainer	Rp 10.258.115,00
Buyung	Programmer, Trainer	Rp 14.507.905,50
Bobby	Trainer, Programmer	Rp 17.878.429,00
Amelia	Project Leader	Rp 40.079.920,75

Total Biaya : Rp 160.319.683,00
Surabaya, 25 September 2010
Mengetahui,

Gambar 4.37 Laporan Project

4.3.38 Laporan *resource*

Laporan *resource* berisi keterangan tentang biaya yang dihabiskan oleh *user* tersebut pada semua *project* yang ada. Untuk lebih jelasnya, laporan *resource* ini dapat dilihat pada Gambar 4.38.

REPORT

Jenis Report: Resource Project Status: Plan 2010

Nama Resource: Pantjwati Sudarmaningtyas OK

Main Report

SSI Solusi Sistem Informasi

Solusi Sistem Informasi
Laporan Resource Plan

Nama User : Pantjwati Sudarmani

Nama Project	Job	Biaya
Sistem Informasi Perguruan Tinggi Maju Terus	Programmer, Trainer, Design Database, Tester	Rp 55.613.747,71

Total Biaya : Rp 55.613.747,71

Surabaya, 25 September 2010
Mengetahui,

Tan Amelia
Ketua Bagian SSI

Gambar 4.38 Laporan *Project*

4.3.39 Laporan *project resource*

Laporan *project resource* berisi keterangan tentang biaya yang dihabiskan oleh *user* tersebut pada sebuah *project* tertentu. Laporan *project resource* akan memunculkan nama *sub task* dan status *user* di *project* tersebut. Untuk lebih jelasnya, laporan *project resource* ini dapat dilihat pada Gambar 4.39.

Nama Sub Task	Status User	Biaya
Komunitas	User ,Trainer	Rp 2.051.623,00
Dosen	User ,Trainer	Rp 1.025.811,50
Mahasiswa	User ,Trainer	Rp 1.465.445,00
Akademik	User ,Trainer	Rp 3.077.434,50
Komunitas	User ,Trainer	Rp 1.465.445,00
Dosen	User ,Trainer	Rp 2.051.623,00
Mahasiswa	User ,Trainer	Rp 2.930.890,00
Tugas Akhir	User ,Trainer	Rp 2.198.167,50
Perpustakaan	User ,Trainer	Rp 2.051.623,00
Akademik	User ,Trainer	Rp 1.465.445,00
Perpustakaan	User ,Trainer	Rp 2.198.167,50
Total Biaya :		Rp 21.981.675,00

Gambar 4.39 Laporan *Project*

4.4 Evaluasi Sistem

Setelah melakukan implementasi sistem, tahap selanjutnya adalah melakukan uji coba dan evaluasi terhadap sistem. Tujuan evaluasi ini adalah untuk mengetahui apakah aplikasi yang telah dibuat ini sudah berjalan dengan baik dan sesuai dengan tujuan dan *output* yang diharapkan. Evaluasi ini dilakukan dengan bantuan sebuah studi kasus.

Pada tanggal 2 April 2010, SSI mendapat pesanan sebuah proyek dari perguruan tinggi “Maju Terus” untuk membuat sistem informasi perguruan tinggi. SSI kemudian melakukan observasi dan menemukan beberapa hal yang harus dikerjakan pada proyek sistem informasi ini. Hasil dari observasi akan digunakan oleh manajer untuk menentukan estimasi biaya. Setelah biaya didapat, maka *market* dapat membuat surat kontrak yang diberikan kepada konsumen.

Selanjutnya, manajer menentukan pemimpin proyek dan pemimpin proyek akan membentuk tim yang berisikan anggota-anggota untuk menyelesaikan proyek tersebut. Pemimpin proyek juga membuat susunan

aktivitas yang nantinya dikerjakan oleh anggota tim. Pada proyek ini, pemimpin proyeknya adalah Amelia dan anggota timnya adalah Pantja, Nia, Hendra, Buyung, dan Bobby. Amelia lalu membuat sebuah jadwal kerja yang terdiri atas 2 level yaitu *task* dan *sub task*. Berikut adalah task dan sub task pada sistem informasi perguruan tinggi “Maju Terus”.

Tabel 4.1 *Task* AAK

<i>Task</i> AAK
Fungsi Umum AAK (Pindah fakultas, menghapus mhs yang keluar, dll)
Penjadwalan Otomatis
Komplain Nilai
Fasilitas
Index Kinerja
Kerja Praktek
Yudisium

Tabel 4.2 *Task* Keuangan

<i>Task</i> Keuangan (Butuh <i>Task</i> AAK Selesai)
Fungsi Umum Keuangan(syarat KRS, Transfer, Pembayaran)
Transaksi khusus Keuangan(Syarat Ujian, Multi payment, kondisi SPP saat mhs keluar)

Tabel 4.3 *Task* HRD

<i>Task</i> HRD (Butuh <i>Task</i> AAK Selesai)
Honor HRD (Revisi, Fingerprint)
Transaksi HRD I (Honor)
Transaksi HRD II (Mutasi, Resign, Cuti)
Transaksi HRD III (Training, Tanggungan, Tunjangan)
Recruitment
Maintenance HRD

Tabel 4.4 *Task* KRS Online

<i>Task</i> KRS Online (Butuh <i>Task</i> AAK dan Keuangan Selesai)
Master KRS (Mata kuliah yang diambil, transkrip, sisa MK, dan jadwal kuliah)
Transaksi KRS (Tambah/Ganti MK, Cetak KRS)
Transaksi khusus KRS (Syarat ambil, Perencanaan studi, Penyimpanan data perubahan)

Tabel 4.5 *Task Sicyca*

<i>Task Sicyca (Butuh Task AAK dan Keuangan Selesai)</i>
Komunitas
Pendidikan
Personal
Perpustakaan
HRD (Untuk Dosen)
Dosen dan Mahasiswa (Untuk Kaprodi dan Pimpinan)
Akademik (Untuk Kaprodi Pimpinan)

Setelah aktivitas selesai dibuat, maka manajer akan menyetujui urutan kerja. Aktivitas yang sudah disetujui akan dikerjakan oleh masing-masing user dan hasil dari kerja itu akan diinputkan user ke dalam sistem. Hasil kerja itu akan dilihat oleh pimpinan proyek, dan jika tidak setuju, project leader dapat merubahnya sendiri. Pada proyek ini, inputan diberikan kepada *sub task* membuat *master* pada *task KRS Online*. User bernama Pantja akan memasukkan persentasi 25 persen yang kemudian dicoba diubah menjadi 15 persen. Sementara itu Pimpinan proyek yaitu Amelia yang melihat akan mengubah persentasi menjadi 20 persen dikarenakan desain yang ada masih belum sempurna.

Setelah semua proyek selesai dikerjakan, maka manajer akan menutup proyek dan sistem akan menghasilkan laporan penutupan proyek. Adapun uji coba yang akan dilakukan adalah sebagai berikut:

4.4.1 Hasi uji coba membuat proyek baru

A. Hasil uji coba memasuki sistem

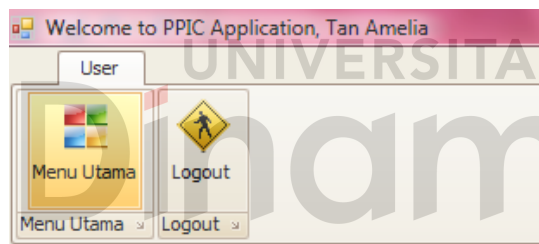
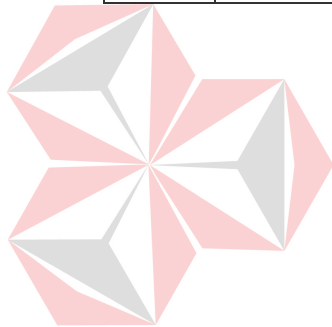
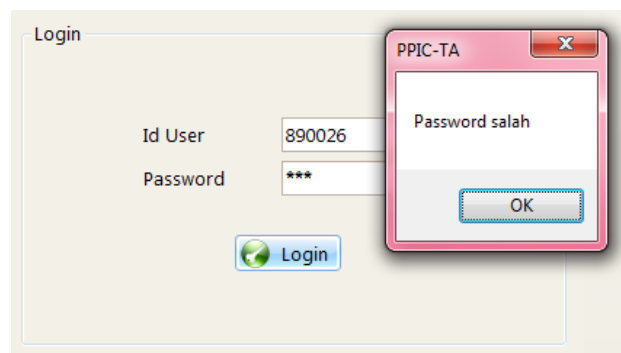
Seorang user dapat masuk ke dalam ketika memasuki *form login*. Di *form login*, seorang *user* harus menginputkan id dan *password* pada. Hasil uji coba yang dilakukan pada *form login* dapat dilihat pada Tabel 4.7.

Tabel 4.6 Data User

Nama Field	Data 1	Data 2
Id User	020393	890026
Password	123	987

Tabel 4.7 Hasil Uji Coba *Form Login*

Test Case	Tujuan	Input	Output Yang Diharapkan	Hasil
1	Mengenali <i>user</i> memasukkan id <i>user</i> dan <i>password</i> yang sesuai	Id <i>user</i> dan <i>password</i> yang valid	<i>User</i> masuk ke dalam sistem	<i>User</i> berhasil masuk ke sistem
2	Mengenali <i>user</i> memasukkan id <i>user</i> dan <i>password</i> yang tidak sesuai	Id <i>user</i> dan <i>password</i> yang tidak valid	Muncul pesan peringatan	Sukses, pesan peringatan muncul

Gambar 4.40 Hasil *Test Case* 1Gambar 4.41 Hasil *Test Case* 2

B. Hasil uji coba membuat proyek baru

Setelah masuk ke dalam sistem, seorang market dapat membuat proyek baru pada form *master project*. Hasil uji coba yang dilakukan pada *form master project* dapat dilihat pada Tabel 4.9.

Tabel 4.8 Data Proyek

Nama Field	Data
Nama Project	Sistem Informasi Perguruan Tinggi Maju Terus
Nama Customer	Dhani
Tgl Mulai Project	02 April 2010
Tgl Selesai Project	15 September 2010
Keterangan Project	Berharap sistem seperti sicyca, sistem untuk AAK, Keuangan dan HRD

Tabel 4.9 Hasil Uji Coba *Form Master Project*

Test Case	Tujuan	Input	Output Yang Diharapkan	Hasil
3	Sistem tidak dapat dikelabui saat memasukkan nilai tanggal selesai yang lebih dulu dari nilai tanggal awal	Memasukkan nilai tanggal selesai yang lebih dulu dari nilai tanggal awal	Nilai tanggal selesai berubah menjadi tanggal awal	Sukses, nilai tanggal selesai mengikuti nilai tanggal awal
4	Menyimpan data <i>project</i>	Data <i>project</i> , tombol <i>save</i>	Project baru terbuat	Sukses, project terbuat dan muncul pada menu utama market
5	Mengubah data <i>project</i>	Data <i>project</i> , tombol edit	Project berubah sesuai dengan perubahan yang dilakukan	Sukses, project berubah sesuai perubahan
6	Mengosongkan semua inputan	Tombol reset	Semua inputan menjadi kosong	Sukses, semua inputan menjadi kosong

Selamat Datang, Nia

Master Project In Progress

Nama Project	Manajer ...	No. SK	Status	Cost	Cancel	Keterangan
> SI Perpustakaan Universitas 45 Surabaya	<input checked="" type="checkbox"/>	SK-2404113	In Progress	Lihat Cost	Cancel Project	Keterangan
Sistem Informasi Perguruan Tinggi Maju Terus	<input type="checkbox"/>	Tambahkan SK	In Progress	Lihat Cost	Cancel Project	Keterangan
Sistem Informasi Rumah Sakit Mitra Keluarga	<input checked="" type="checkbox"/>	Tambahkan SK	In Progress	Lihat Cost	Cancel Project	Keterangan
Sistem Informasi Perguruan Tinggi STIE Wi...	<input checked="" type="checkbox"/>	SK-2205112	In Progress	Lihat Cost	Cancel Project	Keterangan
Sistem Informasi Perpustakaan STIKOM	<input checked="" type="checkbox"/>	Tambahkan SK	In Progress	Lihat Cost	Cancel Project	Keterangan

Gambar 4.42 Hasil Test Case 4

C. Hasil uji coba membuat biaya proyek

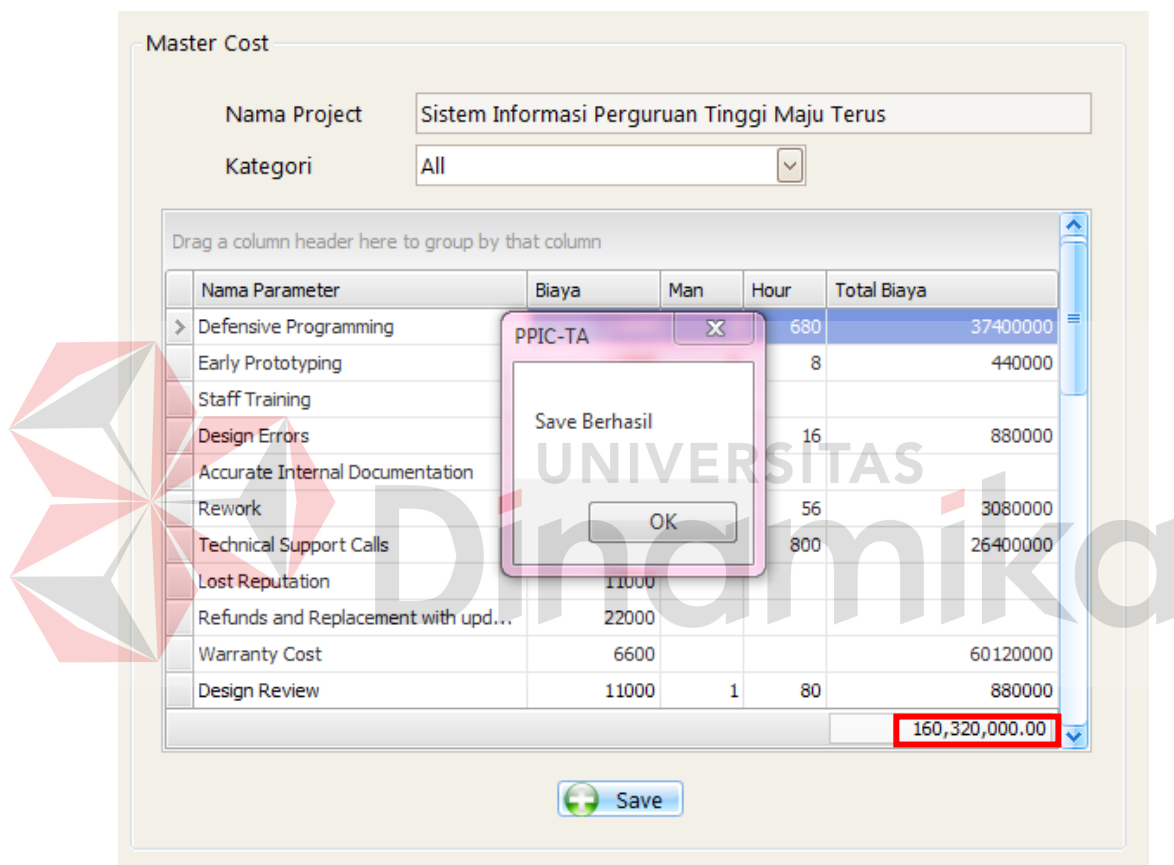
Setelah proyek dibuat, manajer lalu menghitung estimasi biaya dengan metode Cost of Quality yang berdasarkan pada beberapa parameter dalam 4 kategori. Biaya garansi tidak dihitung namun ditetapkan sebesar Rp 60,120,000. Hasil uji coba yang dilakukan pada *form master cost* dapat dilihat pada Tabel 4.11.

Tabel 4.10 Data Biaya

Kategori Biaya	Nama Parameter	Man	Hour	Biaya Man Hour
Biaya Pencegahan	Requirements Analysis	2	60	11,000
Biaya Pencegahan	Clear Specifications	4	10	11,000
Biaya Pencegahan	Early Prototyping	5	8	11,000
Biaya Pencegahan	Defensive Programming	5	680	11,000
Biaya Pencegahan	Coding Errors	5	16	11,000
Biaya Pencegahan	Design Errors	5	16	11,000
Biaya Pencegahan	Operational	5	840	5,000
Biaya Pencegahan	Customer Training	3	200	11,000
Biaya Penilaian	Design Review	1	80	11,000
Biaya Penilaian	Black box Testing	1	40	11,000
Biaya Kegagalan Internal	Bug fixes	5	8	11,000
Biaya Kegagalan Internal	Rework	5	56	11,000
Biaya Kegagalan External	Techinal Support Call	3	800	11,000

Tabel 4.11 Hasil Uji Coba *Form Master Cost*

Test Case	Tujuan	Input	Output Yang Diharapkan	Hasil
7	Mendapatkan estimasi biaya	Memilih parameter, menginputkan data biaya dan menekan tombol save	Estimasi biaya diketahui	Sukses, diketahui estimasi biaya total Rp 160,320,000

Gambar 4.43 Hasil *Test Case 7*

Estimasi biaya dari program lalu akan dibandingkan dengan estimasi biaya yang didapat dari sistem lama SSI yaitu estimasi biaya tanpa adanya metode *Cost of Quality*. Estimasi biaya yang didapat dari sistem lama SSI dapat dilihat pada gambar 4.41.

Estimasi Perkiraan Harga				
Labour Hours				
	Jml org	Bulan	Per bulan	SubTotal
Programmer	4	3	6,500,000	78,000,000
Programmer	2	2	6,500,000	26,000,000
Programmer	2	2	6,500,000	26,000,000
System Analyst	1	2	8,500,000	17,000,000
Project Manager	1	1	17,000,000	17,000,000
		Sub Total		164,000,000

Gambar 4.44 Estimasi Biaya Dengan Sistem Lama SSI

Estimasi biaya yang didapat pada sistem lama SSI adalah Rp 164,000,000. Angka ini didapat dari perhitungan estimasi kerja masing-masing orang pada proyek sistem informasi perguruan tinggi Maju Terus dengan parameter jumlah orang dan jumlah bulan. Biaya bulanan dari masing-masing orang mengikuti referensi dari *Kelly Salary Guide*. Perbedaan dari estimasi biaya dengan metode dan tanpa metode akan dapat dilihat pada Tabel 4.12.

Tabel 4.12 Hasil Uji Coba Perbandingan Dengan Dan Tanpa Metode

Estimasi Biaya dengan CoQ	Estimasi Biaya tanpa CoQ	Selisih
Rp 160,320,000	Rp 164,000,000	Rp 3,680,000

Terdapat selisih 3,680,000 Rupiah atau 2,24 persen antara sistem yang menggunakan metode dengan sistem yang tanpa metode. Selain lebih murah, estimasi biaya dengan metode Cost of Quality juga lebih jelas menerangkan apa saja yang akan dikerjakan pada proyek sistem informasi perguruan tinggi Maju Terus.

D. Hasil uji coba menentukan pemimpin proyek

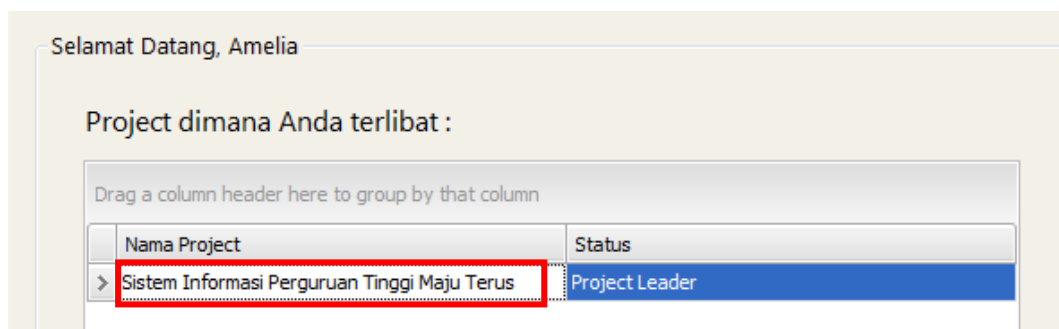
Setelah sebuah proyek mendapatkan persetujuan dari pelanggan dengan keluarnya surat kontrak, maka manajer akan menentukan project leader (pimpinan proyek) pada setiap proyek yang telah disetujui. Langkah ini dilakukan pada *form resource level*. Hasil uji coba yang dilakukan pada *form resource level* dapat dilihat pada Tabel 4.14.

Tabel 4.13 Data *Resource Level*

Nama Field	Data
Nama Proyek	Sistem Informasi Perguruan Tinggi Maju Terus
Nama Pemimpin Proyek	Amelia

Tabel 4.14 Hasil Uji Coba *Form Resource Level*

Test Case	Tujuan	Input	Output Yang Diharapkan	Hasil
8	Menentukan <i>project leader</i>	Nama proyek dan nama pemimpin proyek	<i>Project leader</i> berhasil ditetapkan dan pada <i>project overview</i> dari <i>project leader</i> akan muncul <i>project</i> tersebut	Sukses, dan pada <i>project overview user</i> Amelia, muncul <i>project</i> Sistem Informasi Perguruan Tinggi Maju Terus



Gambar 4.45 Hasil *Test Case* 8

4.4.2 Hasil uji coba merancang proyek

A. Hasil uji coba menentukan anggota proyek

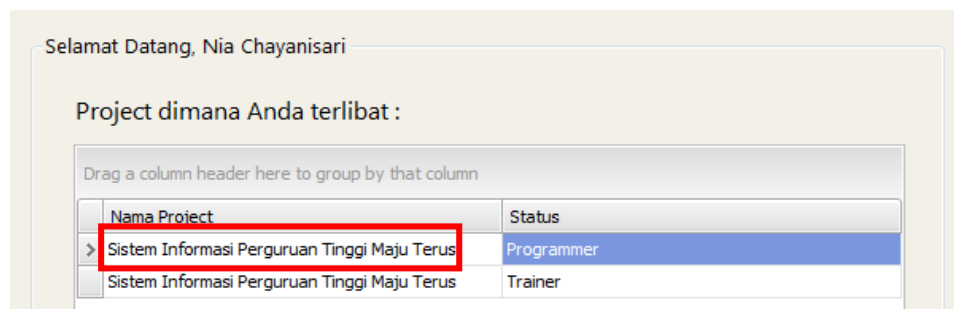
Setelah ditentukan oleh manajer, seorang project leader dapat memilih anggota timnya yang akan membantunya mengerjakan sebuah proyek. Langkah ini dilakukan pada *form resource level* dimana seorang project leader tidak dapat memilih dirinya sendiri. Hasil uji coba yang dilakukan pada *form resource level* dapat dilihat pada Tabel 4.16.

Tabel 4.15 Data *Resource Level*

Nama Field	Data 1	Data 2
Nama Proyek	Sistem Informasi Perguruan Tinggi Maju Terus	Sistem Informasi Perguruan Tinggi Maju Terus
Nama <i>Resource</i>	Nia Chayanisari	Buyung
Nama <i>Level</i>	Programmer	Trainer

Tabel 4.16 Hasil Uji Coba *Form Resource Level*

Test Case	Tujuan	Input	Output Yang Diharapkan	Hasil
1	Menentukan <i>resource project</i>	Nama proyek, nama <i>resource</i> dan nama <i>level</i>	<i>Resource project</i> berhasil ditambahkan dan pada <i>project overview</i> dari <i>user</i> akan muncul <i>project</i> tersebut	Sukses, dan pada <i>project overview user</i> Nia Chayanisari dan Bobby, muncul project Sistem Informasi Perguruan Tinggi Maju Terus

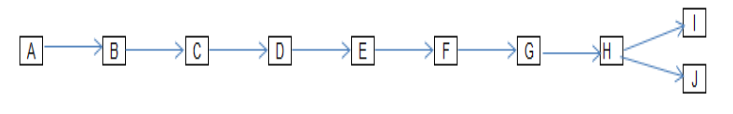
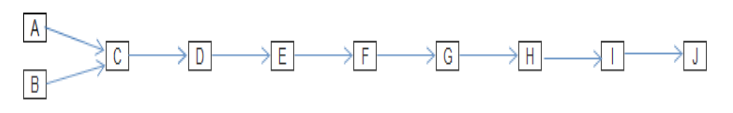



Gambar 4.46 Hasil *Test Case* 9

B. Hasil uji coba *master sub task*

Dalam mengurutkan aktivitas (*sub task*), ada beberapa hal dapat diperhatikan, salah satunya adalah setiap sub task (kecuali awal dan terakhir), harus memiliki minimal satu predesesor dan satu suksesor. Hasil uji coba yang dilakukan pada *form master sub task* dapat dilihat pada Tabel 4.18.

Tabel 4.17 Data Jalur

Contoh Jalur	Gambar	Keterangan
Jalur 1		Memiliki 2 akhir
Jalur 2		Memiliki 2 awal
Jalur 3		Tidak tersambung

Keterangan:

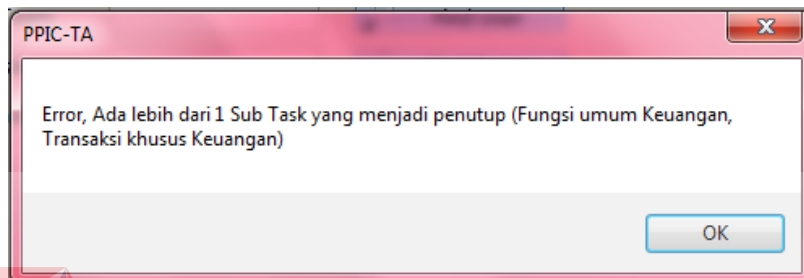
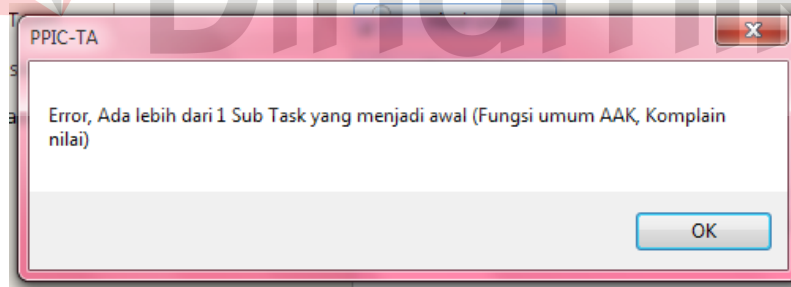
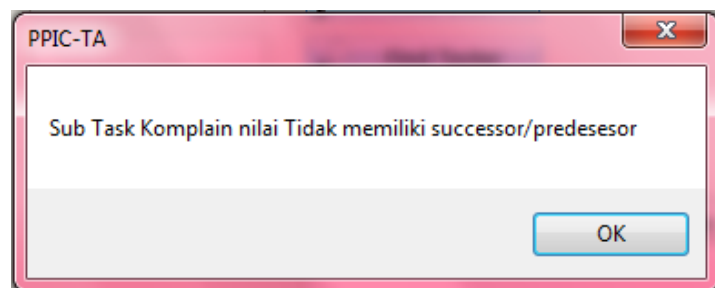
Kode	Sub Task	Predesesor	Task
A	Fungsi Umum AAK	-	AAK
B	Komplain Nilai	Fungsi Umum AAK	AAK
C	Penjadwalan Otomatis	Fungsi Umum AAK	AAK
D	Fasilitas	Penjadwalan Otomatis, Komplain Nilai	AAK
E	Index Kerja	Fasilitas	AAK
F	Perpustakaan	Index Kerja	AAK
G	Kerja Praktek	Perpustakaan	AAK
H	Yudisium	Kerja Praktek	AAK
I	Fungsi Umum Keuangan	Yudisium	Keuangan
J	Transaksi Khusus	Fungsi Umum Keuangan	Keuangan

Tabel 4.18 Hasil Uji Coba *Form Master Sub Task*

Test Case	Tujuan	Input	Output Yang Diharapkan	Hasil
2	Mencegah aktivitas yang memiliki 2 akhir	Jalur 1	Memunculkan error dengan keterangan aktivitas memiliki 2 akhir	Sukses. Error muncul

Tabel 4.18 Hasil Uji Coba *Form Master Sub Task* (Lanjutan)

Test Case	Tujuan	Input	Output Yang Diharapkan	Hasil
3	Mencegah aktivitas yang memiliki 2 awal	Jalur 2	Memunculkan error dengan keterangan aktivitas memiliki 2 awal	Sukses. Error muncul
4	Mencegah aktivitas yang tidak memiliki awal maupun akhir	Jalur 3	Memunculkan error dengan keterangan aktivitas yang tidak jelas	Sukses. Error muncul

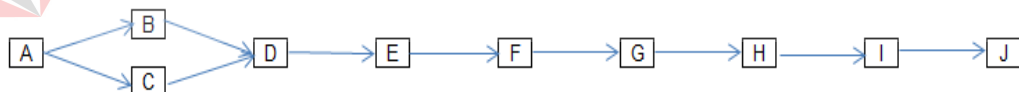
Gambar 4.47 Hasil *Test Case 2*Gambar 4.48 Hasil *Test Case 3*Gambar 4.49 Hasil *Test Case 4*

C. Hasil uji coba menampilkan *critical path*

Setelah *sub task* selesai disusun, maka seorang *project leader* atau manajer dapat melihat *critical path* pada *form summary*. Dalam *task* AAK dan keuangan terdapat 9 *Sub Task*. Untuk jelasnya, dapat dilihat pada tabel 4.19.

Tabel 4.19 Kegiatan Pada *Task* AAK dan Keuangan

Kode	Sub Task	Durasi	Predecessor	Task
A	Fungsi Umum AAK	27 Hari	-	AAK
B	Komplain Nilai	5 Hari	Fungsi Umum AAK	AAK
C	Penjadwalan Otomatis	14 Hari	Fungsi Umum AAK	AAK
D	Fasilitas	14 Hari	Penjadwalan Otomatis, Komplain Nilai	AAK
E	Index Kerja	7 Hari	Fasilitas	AAK
F	Perpustakaan	7 Hari	Index Kerja	AAK
G	Kerja Praktek	7 Hari	Perpustakaan	AAK
H	Yudisium	7 Hari	Kerja Praktek	AAK
I	Fungsi Umum Keuangan	7 Hari	Yudisium	Keuangan
J	Transaksi Khusus	7 Hari	Fungsi Umum Keuangan	Keuangan

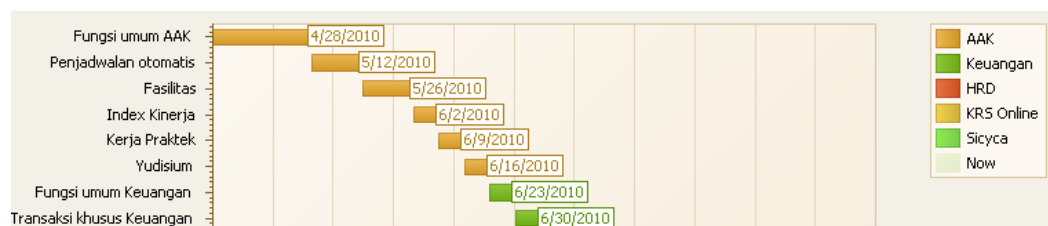


Gambar 4.50 Diagram Dari Kegiatan Pada *Task* AAK dan Keuangan

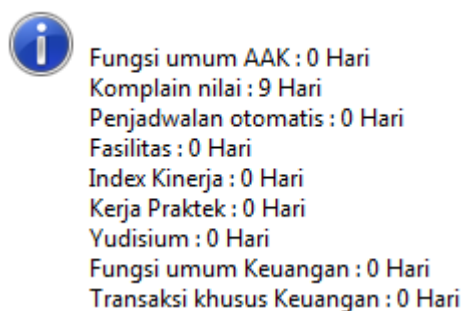
Hasil uji coba yang dilakukan pada *form summary* dapat dilihat pada Tabel 4.20.

Tabel 4.20 Hasil Uji Coba *Form Summary*

Test Case	Tujuan	Input	Output Yang Diharapkan	Hasil
5	Menampilkan gantt chart	Jalur 1	A → C → D → E → F	Sukses, gantt chart sesuai



Gambar 4.51 Gantt Chart *Critical Path* Aktivitas Pada *Task* AAK Dan Keuangan



Gambar 4.52 *Slack* Pada *Task* AAK Dan Keuangan

Dari hasil uji coba yang dilakukan pada *task* AAK dan keuangan dapat disimpulkan bahwa *sub task* fungsi umum AAK, penjadwalan otomatis, fasilitas, index kinerja, kerja praktek, yudisium, fungsi umum keuangan, dan transaksi khusus keuangan menjadi jalur kritis pada *task* ini dimana *sub task* fungsi umum AAK menjadi awal dan *sub task* transaksi khusus keuangan menjadi akhir. Dalam pengerjaannya, *sub task* komplain nilai memiliki slack 9 hari, sehingga batas toleransi keterlambatan *sub task* komplain nilai adalah 9 hari.

4.4.3 Hasil uji coba mengimplementasikan proyek

A. Hasil uji coba memasukkan persentasi

Project yang telah disetujui oleh manajer kemudian dikerjakan oleh *user*. Seorang *user* dapat menginputkan persentasi pekerjaannya dalam sistem dan akan disetujui oleh *project leader*. Langkah ini dilakukan pada form *history work, test*,

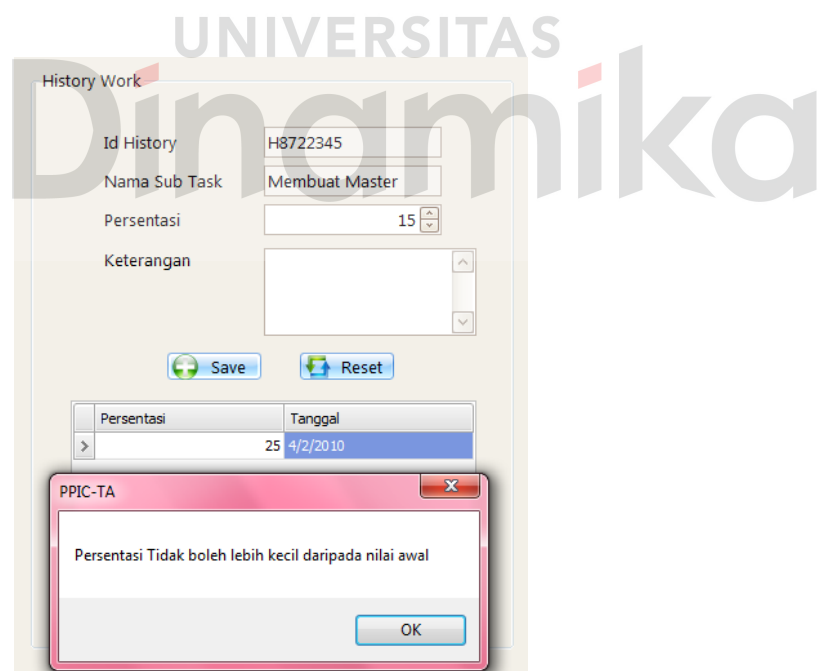
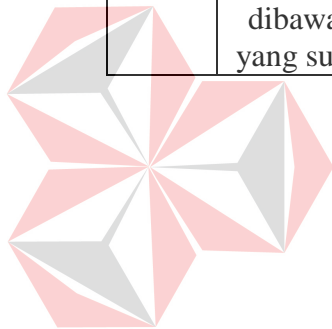
atau *train*. Hasil uji coba yang dilakukan pada *form history work* dapat dilihat pada Tabel 4.22.

Tabel 4.21 Data *History*

Nama Field	Data 1	Data 2
Nama Sub Task	Membuat master barang	Membuat master barang
Tanggal	02-04-2010	05-04-2010
Persentasi	25	15
Keterangan	Desain selesai	Pencarian selesai

Tabel 4.22 Hasil Uji Coba *Form History Work*

Test Case	Tujuan	Input	Output Yang Diharapkan	Hasil
1	Mencegah <i>user</i> memasukkan persentasi dibawah nilai yang sudah ada	<i>Data history</i>	Muncul <i>error</i> akibat persentasi yang diinputkan tidak cocok	Sukses. Muncul error

Gambar 4.53 Hasil *Test Case* 1

B. Desain uji coba mengontrol proyek

Pekerjaan user yang selesai dapat dikontrol oleh *project leader* salah satunya adalah memeriksa apakah pekerjaan user tersebut betul-betul sudah selesai. Langkah ini dilakukan pada *form history*. Uji coba yang akan dilakukan pada *form history* dapat dilihat pada Tabel 4.24.

Tabel 4.23 Data *History*

Nama Field	Data 1
Nama Sub Task	Membuat master barang
Tanggal	06-04-2010
Persentasi	20
Keterangan	Desain belum sempurna

Tabel 4.24 Hasil Uji Coba *Form History*

Test Case	Tujuan	Input	Output Yang Diharapkan	Hasil
2	Mengubah data <i>user</i>	Data history	Data berubah dan pada project team, data tersebut dapat terlihat	Sukses. Data berubah menjadi 20%
3	Menghapus data <i>user</i>	Tombol Delete	Data terhapus dalam sistem, namun anggota proyek tidak dapat melihatnya.	Sukses, pada form approved sub task, membuat master menjadi 0%

History

Nama Sub Task: Membuat Master

Nama User: Pantjawati Sudarmaning

Persentasi: 20 %

Tanggal: 4/6/2010

Keterangan: Desain selesai
PL Note: Desain belum sempurna

Edit Delete

Persentasi	Tanggal
20	4/6/2010

Gambar 4.54 Hasil *Test Case* 2

Approved Sub Task

Id Project: P2420237

Nama Sub Task	Tgl Sub Task	Progress W...	App ...	Progres
Membuat Master	4/15/2010 - 4/29/2010	0%		0
Membuat Transaksi	4/30/2010 - 5/9/2010	0%		0
Komunitas	5/6/2010 - 5/19/2010	0%		0
Transaksi khusus	5/10/2010 - 5/19/2010	0%		0
Komunitas	5/11/2010 - 5/17/2010	0%		0

Gambar 4.55 Hasil *Test Case* 3

4.4.4 Hasil uji coba menutup proyek

A. Hasil uji coba menutup proyek

Setelah semua *sub task* selesai ditraining kepada.konsumen dan disetujui oleh project leader, langkah terakhir adalah menutup proyek. Langkah ini hanya dapat dilakukan oleh manajer dan dilakukan pada *form* hasil proyek. Hasil uji coba yang dilakukan pada *form* hasil proyek dapat dilihat pada Tabel 4.26.

Tabel 4.25 Data Hasil Proyek

Nama Field	Data
Nama Proyek	Sistem Informasi Perguruan Tinggi Maju Terus
Tanggal	9/25/2010
Status	Completed
Keterangan	Konsumen puas dan berharap mengerjakan sistem yang lain

Tabel 4.26 Hasil Uji Coba *Form* Hasil Proyek

Test Case	Tujuan	Input	Output Yang Diharapkan	Hasil
4	Menutup proyek	Data hasil proyek	Proyek berhasil ditutup dan memunculkan laporan penutupan	Sukses. Proyek berhasil ditutup dan muncul laporan penutupan

Gambar 4.56 Hasil *Test Case* 4

BAB V

PENUTUP

5.1 Kesimpulan

Setelah dilakukan uji coba dan evaluasi terhadap aplikasi untuk perencanaan biaya dan waktu proyek sistem informasi ini, maka dapat ditarik kesimpulan sebagai berikut:

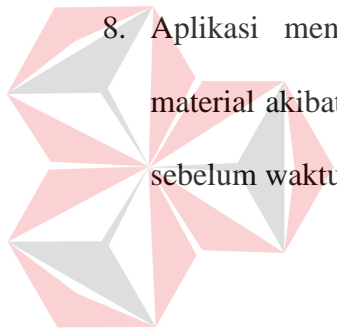
1. Sistem ini dapat menghasilkan estimasi biaya yang dibutuhkan dengan metode *Cost of Quality* tetapi biaya estimasinya masih jauh dibawah biaya aktual, sehingga dibutuhkan pengembangan lebih lanjut.
2. Sistem ini dapat menghasilkan jalur kritis dengan metode *Critical Path* sehingga pemimpin proyek dapat mengelola pembuatan proyek sistem informasi.
3. Sistem ini dapat mengelola perencanaan biaya dan waktu dengan menghasilkan estimasi biaya dan jalur kritis.

5.2 Saran

Adapun beberapa saran yang dapat diberikan kepada peneliti berikutnya apabila ingin mengembangkan sistem yang telah dibuat ini agar menjadi lebih baik adalah sebagai berikut:

1. Tampilan aplikasi ini masih sederhana sehingga dapat ditingkatkan lagi kualitasnya.
2. Aplikasi mendatang sebaiknya dapat menggunakan enkripsi data atau teknologi lainnya untuk keamanan data.

3. Aplikasi mendatang sebaiknya dapat mengurutkan waktu dengan cara selain *finish to start*.
4. Aplikasi mendatang sebaiknya dapat menampilkan hubungan antar *sub task* dalam *gantt chart*.
5. Aplikasi mendatang sebaiknya dapat menampilkan jalur kritis pada level selain *sub task*.
6. Aplikasi mendatang sebaiknya dapat menghubungkan sumber daya manusia dengan metode *critical path*.
7. Aplikasi mendatang sebaiknya dapat menampilkan waktu aktual dengan metode *critical path*.
8. Aplikasi mendatang sebaiknya mampu menghasilkan laporan kerugian material akibat keterlambatan ataupun keuntungan akibat proyek yang selesai sebelum waktunya.



DAFTAR PUSTAKA

- Gray, Clifford F. dan Larson, Eric W., 2007, Manajemen Proyek Proses Manajerial, Andi, Yogyakarta.
- Hansen, Don R. dan Mowen, Maryanne M., 2000, Manajemen Biaya Akuntansi dan Pengendalian, Salemba Empat, Jakarta.
- Herlambang, Soendoro dan Tanuwijaya, Haryanto, 2005, Sistem Informasi: Konsep, Teknologi & Manajemen, Graha Ilmu, Yogyakarta.
- Hillier, Frederick S. dan Lieberman, Gerald J., 1994, Pengantar Riset Operasi, Erlangga, Jakarta.
- Kaner, Cem, 1996, Quality Cost Analysis: Benefits and Risks. URL : http://www.kaner.com/pdfs/Quality_Cost_Analysis.pdf.
- Marlinda, Linda, 2004, Sistem Basis Data, Andi, Yogyakarta.
- Mustakini, Jogyanto H., Analisis & Disain Sistem Informasi: Pendekatan Terstruktur teori dan praktek aplikasi bisnis, Andi, Yogyakarta, 2001
- PMI, 2008, A Guide To The Project Management Body of Knowledge (PMBOK Guide) Fourth Edition, Project Management Institute, Pennsylvania.
- Romeo, 2003, Testing dan Implementasi Sistem, STIKOM, Surabaya.
- Schiffauerova, A. dan Thomson, V., 2006, A Review Of Research On Cost Of Quality Models And Best Practices, International Journal of Quality and Reliability Management, 23 (4), 10-13.
- Taha, Hamdy A, 1997, Riset Operasional Suatu Pengantar, Binarupa Aksara, Jakarta.