

**RANCANG BANGUN SISTEM INFORMASI MANAJEMEN
PROYEK MENGGUNAKAN *CRITICAL PATH METHOD* DAN
WHAT IF ANALYSIS BERBASIS WEB
(STUDI KASUS: PT. TRIPRIMA KARYA)**



**PROGRAM STUDI SISTEM INFORMASI
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI SYARIF HIDAYATULLAH JAKARTA
2022 M / 1443 H**

LEMBAR PERNYATAAN

DENGAN INI SAYA MENYATAKAN BAHWA SKRIPSIINI BENAR-BENAR HASIL KARYA SENDIRI YANG BELUM PERNAH DIAJUKAN SEBAGAI SKRIPSI ATAU KARYA ILMIAH PADA PERGURUAN TINGGI MANAPUN.

Jakarta, 11 Maret 2022



Farah Dhia Yasmin
11160930000084



ABSTRAK

FARAH DHIA YASMIN – 11160930000084 Rancang Bangun Sistem Informasi Manajemen Proyek Menggunakan *Critical Path Method* dan *What If Analysis* Berbasis Web (Studi Kasus: PT. Triprima Karya). Di bawah bimbingan **Zulfiandri dan Rinda Hesti Kusumaningtyas**.

Manajemen proyek merupakan proses perencanaan, pelaksanaan, pemantauan dan penutupan proyek agar proyek berjalan sesuai dengan target. Salah satu perusahaan yang memerlukan manajemen proyek adalah PT. Triprima Karya yang menawarkan jasa konsultasi proyek konstruksi. Banyaknya proyek yang dilakukan dalam waktu bersamaan, membuat direksi sulit untuk memantau *progress* sehingga sulit untuk mengantisipasi keterlambatan. Selain itu, perusahaan juga tidak mempunyai pengingat untuk kegiatan yang sudah mendekati waktu tenggat. Dan pelaporan tidak dilakukan secara *realtime* serta pembuatan laporan yang rentan terjadi *human error*. Tujuan dari penelitian ini adalah membangun sistem informasi manajemen proyek yang dapat memantau proyek, mengantisipasi keterlambatan dan memudahkan pelaporan *progress* serta pembuatan laporan proyek. Metode yang digunakan untuk analisis keterlambatan adalah *Critical Path Method* dan *What If Analysis*. Metode pengembangan sistem yang digunakan adalah *Rapid Application Development* (RAD). Perancangan sistem menggunakan *Unified Modelling Language* (UML). Bahasa pemrograman sistem yang digunakan adalah ReactJS, PHP framework Laravel dan database MySQL. Hasil pengujian sistem menggunakan *black box testing* menunjukkan bahwa *output* yang ditampilkan sudah sesuai yaitu sebuah sistem informasi manajemen proyek berbasis web yang dapat memantau *progress* proyek, menganalisis keterlambatan, melakukan pelaporan secara *realtime* dan pembuatan laporan secara otomatis serta terdapat fitur validasi untuk mencegah karyawan mengikuti lebih dari tiga proyek dalam waktu bersamaan.

Kata Kunci: Manajemen Proyek, *Critical Path Method*, *What If Analysis*, *Rapid Application Development* (RAD)

V Bab + xix Halaman + 277 Halaman + 121 Gambar + 66 Tabel + Lampiran
Pustaka Acuan (50, 2011 – 2021)

KATA PENGANTAR

Assalamualaikum Wr. Wb.

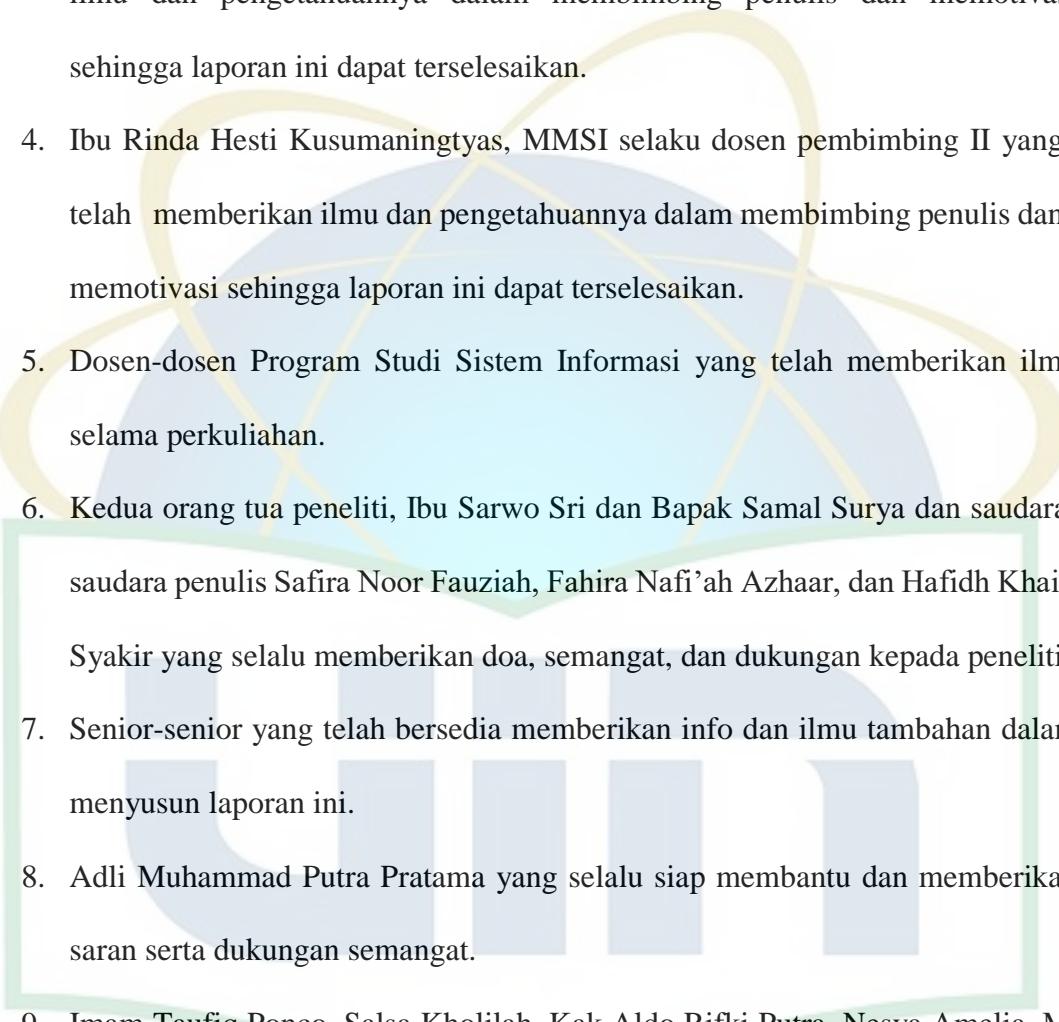
Puji dan syukur atas kehadiran Allah SWT yang telah memberikan rahmat serta hidayah-Nya sehingga penulis dapat menyelesaikan skripsi ini. Semoga shalawat serta salam selalu tercurahkan kepada junjungan kita Rasulullah Muhammad SAW yang telah membawa kita keluar dari zaman kegelapan.

Dalam skripsi ini, penulis menyadari bahwa masih banyak terdapat kekurangan. Namun demikian penulis skripsi ini dapat memenuhi prasyarat dalam memperoleh gelar sarjana (S-1) dalam jurusan Sistem Informasi di Fakultas Sains dan Teknologi UIN Syarif Hidayatullah Jakarta.

Skripsi yang berjudul “Rancang Bangun Sistem Informasi Manajemen Proyek Menggunakan *Critical Path Method* dan *What If Analysis* Berbasis Web (Studi Kasus: PT. Triprima Karya)” akhirnya dapat diselesaikan sesuai dengan harapan penulis. Selama penyusunan skripsi ini tentunya penulis menghadapi banyak kesulitan dan tantangan. Semua kesulitan dapat penulis lalui tak lepas dari kebesaran hati dari berbagai pihak dalam membantu penulis.

Pada kesempatan ini penulis juga hendak mengucapkan terima kasih kepada pihak-pihak yang telah membantu memberikan dukungan baik berupa materi maupun imateri, sehingga penulis dapat menyelesaikan skripsi ini. Ucapan terima kasih secara khusus penulis berikan kepada:

1. Bapak Nashrul Hakiem, S.Si., M.T., Ph.D. selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Syarif Hidayatullah Jakarta.

- 
2. Bapak Aang Subiyakto, Ph.D., selaku Ketua Program Studi Sistem Informasi dan Nuryasin, M.Kom., selaku Sekretaris Program studi Sistem Informasi.
 3. Bapak Zulfiandri, MMSI selaku dosen pembimbing I yang telah memberikan ilmu dan pengetahuannya dalam membimbing penulis dan memotivasi sehingga laporan ini dapat terselesaikan.
 4. Ibu Rinda Hesti Kusumaningtyas, MMSI selaku dosen pembimbing II yang telah memberikan ilmu dan pengetahuannya dalam membimbing penulis dan memotivasi sehingga laporan ini dapat terselesaikan.
 5. Dosen-dosen Program Studi Sistem Informasi yang telah memberikan ilmu selama perkuliahan.
 6. Kedua orang tua peneliti, Ibu Sarwo Sri dan Bapak Samal Surya dan saudara-saudara penulis Safira Noor Fauziah, Fahira Nafi'ah Azhaar, dan Hafidh Khairi Syakir yang selalu memberikan doa, semangat, dan dukungan kepada peneliti.
 7. Senior-senior yang telah bersedia memberikan info dan ilmu tambahan dalam menyusun laporan ini.
 8. Adli Muhammad Putra Pratama yang selalu siap membantu dan memberikan saran serta dukungan semangat.
 9. Imam Taufiq Ponco, Salsa Kholilah, Kak Aldo Rifki Putra, Nesya Amelia, M. Khairuzzaman yang selalu membantu memberikan arahan, dan berbagi ilmu yang sangat membantu pengerjaan skripsi.
 10. Jihan Azita Maharani dan Khonsa Zofiro yang selalu menjadi pendengar yang baik dan memberikan semangat kepada peneliti.

11. Teman-teman Telescoope, Kak Muhammad Taqwa Aziz, Kak Willy Andika, Kak Raka, Kak Rangga, Kak Hamdi, Widya Azzahra, Ricky dan lainnya yang selalu memotivasi peneliti untuk menyelesaikan skripsi.
12. Teman kelas C serta angkatan 2016 yang berjuang bersama untuk kelulusan dan berbagi semangat.
13. Semua pihak yang tidak dapat disebutkan satu per satu yang telah membantu hingga laporan ini terselesaikan.

Penulis menyadari bahwa dalam penyusunan skripsi ini masih jauh dari kata sempurna, serta masih banyak kekurangan baik dalam penulisan materi maupun dalam susunan bahasanya. Untuk itu kiranya, pembaca dapat memaklumi atas kekurangan dalam laporan ini. Akhir kata penulis berharap semoga laporan ini dapat bermanfaat bagi penulis khususnya dan bagi para pembaca pada umumnya.

Wassalamualaikum Wr. Wb

Jakarta, 4 Februari 2022



Farah Dhia Yasmin

11180930000084

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PENGESAHAN	ii
LEMBAR PENGESAHAN UJIAN	iii
LEMBAR PERNYATAAN	iv
ABSTRAK	v
KATA PENGANTAR.....	vi
DAFTAR ISI.....	ix
DAFTAR GAMBAR.....	xiv
DAFTAR TABEL	xx
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Identifikasi Masalah	5
1.3 Rumusan Masalah	6
1.4 Batasan Masalah.....	6
1.5 Tujuan Penelitian.....	7
1.6 Manfaat Penelitian.....	8
1.6.1. Bagi Peneliti.....	8
1.6.2. Bagi Instansi.....	8
1.6.3. Bagi Universitas	9
1.7 Metodologi Penelitian	9
1.7.1 Metode Pengumpulan Data	9

1.7.2	Metode Analisis Sistem Berjalan	10
1.7.3	Metode Pengembangan Sistem Usulan.....	10
1.8	Sistematika Penulisan.....	11
BAB 2 TINJAUAN PUSTAKA		13
2.1	Pengertian Rancang Bangun	13
2.2	Konsep Dasar Sistem Informasi	13
2.2.1	Pengertian Sistem.....	13
2.2.2	Pengertian Informasi	14
2.2.3	Pengertian Sistem Informasi	14
2.2.4	Komponen Sistem Informasi	15
2.2.5	Klasifikasi Sistem Informasi	16
2.2.6	Tipe-Tipe Sistem Informasi	17
2.3	Sistem Informasi Manajemen.....	18
2.4	Pengertian Proyek.....	18
2.5	Manajemen Proyek.....	19
2.5.1	<i>Process Group</i>	24
2.5.2	<i>Knowledge Area</i>	25
2.5.3	Penjadwalan Proyek	26
2.6	<i>Critical Path Method</i>	29
2.7	<i>What If Analysis</i>	31
2.7.1	Pengertian <i>What If Analysis</i>	31
2.7.2	Tahapan Analisis <i>What If</i> pada <i>Critical Path Method</i>	31
2.8	Metode Analisis Sistem SWOT	34

2.9	Metode Pengembangan Sistem	36
2.9.1	Pengertian RAD	36
2.9.2	Tahapan RAD.....	37
2.9.3	Keunggulan dan Kelemahan RAD.....	38
2.9.4	Pengujian Sistem.....	39
2.10	<i>Unified Modeling Language</i>	40
2.10.1	<i>Use Case Diagram</i>	41
2.10.2	<i>Activity Diagram</i>	43
2.10.3	<i>Class Diagram</i>	45
2.10.4	<i>Sequence Diagram</i>	46
2.10.5	<i>Component Diagram</i>	47
2.10.6	<i>Deployment Diagram</i>	49
2.11	Konsep Dasar Basis Data	50
2.11.1	Basis Data	50
2.11.2	<i>Database Management System (DBMS)</i>	52
2.11.3	Normalisasi Basis Data	53
2.12	ReactJS	55
2.13	PHP.....	56
2.14	MySQL.....	57
2.15	<i>Tools</i>	58
2.15.1	draw.io.....	58
2.15.2	Figma	58
2.16	Studi Penelitian Sejenis	58

BAB 3 METODOLOGI PENELITIAN	67
3.1 Metode Pengumpulan Data	67
3.1.1 Studi Pustaka.....	67
3.1.2 Studi Lapangan.....	67
3.2 Metode Analisis Sistem Berjalan	68
3.3 Metode Pengembangan Sistem	69
3.3.1 <i>Requirement Planning</i>	69
3.3.2 RAD <i>Design Workshop</i>	70
3.3.3 Implementasi	72
3.4. Tahapan Penelitian	72
BAB 4 PEMBAHASAN	74
4.1 <i>Requirements Planning</i>	74
4.1.1 Gambaran Umum Perusahaan.....	74
4.1.1.1 Gambaran Umum PT. Triprima Karya.....	74
4.1.1.2 Struktur Organisasi.....	74
4.1.1.3 Tugas dan Tanggung Jawab	75
4.1.2 Analisis Permasalahan	81
4.1.2.1 Identifikasi Sistem Berjalan	81
4.1.2.2 Analisis Sistem dengan SWOT	83
4.1.3 Analisis Sistem Usulan	85
4.2 <i>Design Workshop</i>	87
4.2.1 Perancangan Analisis Keterlambatan.....	87
4.2.1.1 Perhitungan <i>Critical Path Method</i>	87

4.2.1.2	Perhitungan <i>What If Analysis</i>	91
4.2.2	Perancangan UML	94
4.2.2.1	<i>Use Case Diagram</i>	94
4.2.2.2	<i>Activity Diagram</i>	132
4.2.2.3	<i>Class Diagram</i>	149
4.2.2.4	<i>Sequence Diagram</i>	151
4.2.2.5	<i>Component Diagram</i>	173
4.2.2.6	<i>Deployment Diagram</i>	174
4.2.3	Perancangan <i>Database</i>	175
4.2.3.1	Normalisasi <i>Database</i>	175
4.2.3.2	Skema <i>Database</i>	189
4.2.3.3	Spesifikasi <i>Database</i>	190
4.2.4	<i>Design Interface</i>	213
4.2.4.1	Perancangan Struktur Menu	213
4.2.4.2	Perancangan <i>Interface</i>	215
4.3	Implementasi	248
4.3.1	Pemrograman	248
4.3.2	Pengujian.....	248
BAB 5 PENUTUPAN	270
5.1	Kesimpulan.....	270
5.2	Saran.....	271
DAFTAR PUSTAKA	272
LAMPIRAN	xx

DAFTAR GAMBAR

Gambar 1.1 <i>Mapping</i> Manajemen Proyek (Institute, 2017)	2
Gambar 2.1 Komponen Manajemen Proyek (Sugiyanto, 2020).....	20
Gambar 2.2 Siklus Hidup Manajemen Proyek.....	24
Gambar 2.3 Bagian <i>Critical Path Method</i> (Utami & Asnawati, 2015)	30
Gambar 2.4 Matriks Metode SWOT (Hamali, 2016)	35
Gambar 2.5 Tahapan Metode RAD (Kosasi, 2015).....	38
Gambar 2.6 Contoh <i>Use Case Diagram</i> (Unhelkar, 2018).....	43
Gambar 2.7 Contoh <i>Activity Diagram</i> (Unhelkar, 2018).....	44
Gambar 2.8 Contoh <i>Class Diagram</i> (Unhelkar, 2018)	46
Gambar 2.9 Contoh <i>Sequence Diagram</i> (Unhelkar, 2018).....	47
Gambar 2.10 Contoh <i>Component Diagram</i> (Visual Paradigm, 2020a)	48
Gambar 2.11 Contoh <i>Deployment Diagram</i> (Visual Paradigm, 2020b).....	50
Gambar 2.12 Bentuk Normalisasi (Indrajani, 2015).....	53
Gambar 3.1 Tahapan Penelitian	73
Gambar 4.1 Struktur Organisasi PT. Triprima Karya	75
Gambar 4.2 <i>Rich Picture</i> Sistem Berjalan	81
Gambar 4.3 <i>Rich Picture</i> Sistem Usulan	86
Gambar 4.4 Grafik Hasil <i>Critical Path Method</i>	90
Gambar 4.5 <i>Use Case Diagram</i> Sistem Informasi Manajemen Proyek.....	97
Gambar 4.6 <i>Activity Diagram</i> Kelola Data User	132
Gambar 4.7 <i>Activity Diagram</i> Login.....	133

Gambar 4.8 <i>Activity Diagram</i> Kelola Data Proyek.....	134
Gambar 4.9 <i>Activity Diagram</i> Kelola Data Karyawan	135
Gambar 4.10 <i>Activity Diagram</i> Kelola Data Klien	136
Gambar 4.11 <i>Activity Diagram</i> Kelola Detail Proyek	138
Gambar 4.12 <i>Activity Diagram</i> Verifikasi	140
Gambar 4.13 <i>Activity Diagram</i> Kelola Tugas.....	141
Gambar 4.14 <i>Activity Diagram</i> Update Progress Tugas	142
Gambar 4.15 <i>Activity Diagram</i> Tutup Tugas.....	143
Gambar 4.16 <i>Activity Diagram</i> Analisis Keterlambatan Proyek	144
Gambar 4.17 <i>Activity Diagram</i> Kelola Laporan	145
Gambar 4.18 <i>Activity Diagram</i> Tutup Proyek	146
Gambar 4.19 <i>Activity Diagram</i> Ubah Profil	147
Gambar 4.20 <i>Activity Diagram</i> Kelola Pesan	148
Gambar 4.21 <i>Activity Diagram</i> Logout.....	148
Gambar 4.22 <i>Class Diagram</i> Sistem Informasi Manajemen Proyek	150
Gambar 4.23 <i>Sequence Diagram</i> Kelola Data User	151
Gambar 4.24 <i>Sequence Diagram</i> Login	152
Gambar 4.25 <i>Sequence Diagram</i> Kelola Data Proyek.....	153
Gambar 4.26 <i>Sequence Diagram</i> Kelola Data Karyawan.....	155
Gambar 4.27 <i>Sequence Diagram</i> Kelola Data Klien	157
Gambar 4.28 <i>Sequence Diagram</i> Kelola Detail Proyek.....	160
Gambar 4.29 <i>Sequence Diagram</i> Verifikasi	163
Gambar 4.30 <i>Sequence Diagram</i> Kelola Tugas.....	164

Gambar 4.31 <i>Sequence Diagram</i> Update Progress Tugas	165
Gambar 4.32 <i>Sequence Diagram</i> Tutup Tugas	166
Gambar 4.33 <i>Sequence Diagram</i> Analisis Keterlambatan Proyek	167
Gambar 4.34 <i>Sequence Diagram</i> Kelola Laporan	169
Gambar 4.35 <i>Sequence Diagram</i> Tutup Proyek	170
Gambar 4.36 <i>Sequence Diagram</i> Ubah Profil.....	171
Gambar 4.37 <i>Sequence Diagram</i> Kelola Pesan	172
Gambar 4.38 <i>Sequence Diagram</i> Logout	173
Gambar 4.39 <i>Component Diagram</i> Sistem Informasi Manajemen Proyek	174
Gambar 4.40 <i>Deployment Diagram</i> Sistem Informasi Manajemen Proyek	175
Gambar 4.41 <i>Second Normal Form</i>	188
Gambar 4.42 <i>Third Normal Form</i>	189
Gambar 4.43 Skema Database.....	190
Gambar 4.44 Struktur Menu Aplikasi Admin	213
Gambar 4.45 Struktur Menu Aplikasi Direktur Teknis	214
Gambar 4.46 Struktur Menu Aplikasi Kepala Divisi Proyek	214
Gambar 4.47 Struktur Menu Aplikasi <i>Team Leader</i> dan Tenaga Ahli	215
Gambar 4.48 Struktur Menu Aplikasi Administrasi Proyek	215
Gambar 4.49 <i>Login Interface</i>	216
Gambar 4.50 <i>Dashboard Admin Interface</i>	216
Gambar 4.51 <i>User Table Interface</i>	217
Gambar 4.52 <i>Form Tambah User</i>	217
Gambar 4.53 <i>Form Ubah User</i>	218

Gambar 4.54 Tabel Karyawan <i>Interface</i>	218
Gambar 4.55 <i>Form</i> Tambah Karyawan	219
Gambar 4.56 <i>Form</i> Ubah Karyawan.....	219
Gambar 4.57 Tabel Klien <i>Interface</i>	220
Gambar 4.58 <i>Form</i> Tambah Klien.....	220
Gambar 4.59 <i>Form</i> Ubah Klien	221
Gambar 4.60 Daftar Proyek Admin	221
Gambar 4.61 <i>Form</i> Tambah Proyek Admin	222
Gambar 4.62 <i>Form</i> Ubah Proyek Admin.....	223
Gambar 4.63 Daftar Pesan Admin	223
Gambar 4.64 Detail Pesan Admin.....	224
Gambar 4.65 Ubah Profil Admin	224
Gambar 4.66 <i>Dashboard</i> Direktur Teknis	225
Gambar 4.67 Verifikasi Jadwal.....	226
Gambar 4.68 Verifikasi Tim	226
Gambar 4.69 Verifikasi Analisis.....	227
Gambar 4.70 Verifikasi Laporan.....	227
Gambar 4.71 Daftar Proyek Direktur Teknis	228
Gambar 4.72 <i>Form</i> Detail Proyek Direktur Teknis	229
Gambar 4.73 <i>Form</i> Jadwal Proyek Direktur Teknis	229
Gambar 4.74 <i>Form</i> Tim Proyek Direktur Teknis.....	230
Gambar 4.75 <i>Form</i> Analisis Proyek Direktur Teknis	230
Gambar 4.76 Daftar Pesan Direktur Teknis.....	231

Gambar 4.77 Detail Pesan Direktur Teknis	231
Gambar 4.78 Ubah Profil Direktur Teknis.....	232
Gambar 4.79 <i>Dashboard</i> Kepala Divisi Proyek	233
Gambar 4.80 Daftar Proyek Kepala Divisi Proyek	234
Gambar 4.81 <i>Form</i> Detail Proyek Kepala Divisi Proyek	235
Gambar 4.82 <i>Form</i> Jadwal Proyek Kepala Divisi Proyek	235
Gambar 4.83 <i>Form</i> Tim Proyek Kepala Divisi Proyek.....	236
Gambar 4.84 Daftar Pesan Kepala Divisi Proyek	236
Gambar 4.85 Detail Pesan Kepala Divisi Proyek	237
Gambar 4.86 Ubah Profil Kepala Divisi Proyek.....	237
Gambar 4.87 <i>Dashboard Team Leader</i>	238
Gambar 4.88 Daftar Tugas <i>Team Leader</i>	239
Gambar 4.89 <i>Form</i> Detail Tugas	239
Gambar 4.90 Tutup Tugas <i>Interface</i>	240
Gambar 4.91 Daftar Pesan <i>Team Leader</i>	241
Gambar 4.92 Detail Pesan <i>Team Leader</i>	241
Gambar 4.93 Ubah Profil <i>Team Leader</i>	242
Gambar 4.94 <i>Dashboard</i> Tenaga Ahli	242
Gambar 4.95 Daftar Tugas Tenaga Ahli	243
Gambar 4.96 <i>Form</i> Detail Tugas Tenaga Ahli.....	243
Gambar 4.97 Daftar Pesan Tenaga Ahli	244
Gambar 4.98 Detail Pesan Tenaga Ahli	244
Gambar 4.99 Ubah Profil Tenaga Ahli	245

Gambar 4.100 <i>Dashboard</i> Administrasi Proyek	245
Gambar 4.101 Daftar Laporan	246
Gambar 4.102 <i>Form</i> Detail Laporan	246
Gambar 4.103 Daftar Pesan Administrasi Proyek	247
Gambar 4.104 Detail Pesan Administrasi Proyek	247
Gambar 4.105 Ubah Profil Administrasi Proyek	248



DAFTAR TABEL

Tabel 2.1 <i>Multiplicity Class Diagram</i>	45
Tabel 2.2 Penelitian Sejenis	60
Tabel 4.1 Matriks SWOT Analisis Sistem Berjalan	84
Tabel 4.2 Identifikasi Aktivitas.....	88
Tabel 4.3 Hasil Perhitungan Maju dan Mundur.....	89
Tabel 4.4 Hasil Perhitungan <i>Critical Path Method</i>	90
Tabel 4.5 Hasil Perhitungan <i>What If Analysis</i>	93
Tabel 4.6 Identifikasi Aktor	94
Tabel 4.7 Identifikasi <i>Use Case</i>	95
Tabel 4.8 Narasi <i>Use Case</i> Kelola Data User.....	98
Tabel 4.9 Narasi <i>Use Case</i> Login.....	100
Tabel 4.10 Narasi <i>Use Case</i> Kelola Data Proyek	101
Tabel 4.11 Narasi <i>Use Case</i> Kelola Data Karyawan	103
Tabel 4.12 Narasi <i>Use Case</i> Kelola Data Klien.....	106
Tabel 4.13 Narasi <i>Use Case</i> Kelola Detail Proyek	108
Tabel 4.14 Narasi <i>Use Case</i> Verifikasi.....	112
Tabel 4.15 Narasi <i>Use Case</i> Kelola Tugas.....	117
Tabel 4.16 Narasi <i>Use Case</i> <i>Update progress tugas</i>	119
Tabel 4.17 Narasi <i>Use Case</i> Tutup Tugas.....	120
Tabel 4.18 Narasi <i>Use Case</i> Analisis Keterlambatan Proyek	122
Tabel 4.19 Narasi <i>Use Case</i> Kelola Laporan	125

Tabel 4.20 Narasi <i>Use Case</i> Tutup Proyek	127
Tabel 4.21 Narasi <i>Use Case</i> Ubah Profil	128
Tabel 4.22 Narasi <i>Use Case</i> Kelola Pesan	130
Tabel 4.23 Narasi <i>Use Case</i> Logout.....	131
Tabel 4.24 <i>Unnnormalized Form</i>	176
Tabel 4.25 <i>First Normal Form</i>	181
Tabel 4.26 Tabel <i>User</i>	191
Tabel 4.27 Tabel <i>Role</i>	192
Tabel 4.28 Tabel Karyawan	192
Tabel 4.29 Tabel Posisi	193
Tabel 4.30 Tabel Karyawan Detail	194
Tabel 4.31 Tabel Pesan	195
Tabel 4.32 Tabel Klien.....	196
Tabel 4.33 Tabel Proyek	197
Tabel 4.34 Tabel Cuaca.....	198
Tabel 4.35 Tabel Jadwal	199
Tabel 4.36 Tabel Aktivitas.....	200
Tabel 4.37 Tabel <i>Successor</i>	201
Tabel 4.38 Tabel CPM	202
Tabel 4.39 Tabel <i>What If</i>	203
Tabel 4.40 Tabel <i>What If</i> Detail.....	205
Tabel 4.41 Tabel Tim	206
Tabel 4.42 Tabel Tim Detail	207

Tabel 4.43 Tabel Tugas.....	208
Tabel 4.44 Tabel Hasil Tugas	210
Tabel 4.45 Tabel Laporan	211
Tabel 4.46 Tabel Notifikasi.....	212
Tabel 4.47 Hasil Pengujian <i>Login</i>	249
Tabel 4.48 Hasil Pengujian <i>Dashboard</i>	249
Tabel 4.49 Hasil Pengujian Menu <i>User</i>	249
Tabel 4.50 Hasil Pengujian Menu Karyawan	251
Tabel 4.51 Hasil Pengujian Menu Klien	252
Tabel 4.52 Hasil Pengujian Proyek	254
Tabel 4.53 Hasil Pengujian Menu Jadwal	255
Tabel 4.54 Hasil Pengujian Menu Tim	257
Tabel 4.55 Hasil Pengujian Menu Analisis.....	259
Tabel 4.56 Hasil Pengujian Menu Verifikasi Jadwal.....	260
Tabel 4.57 Hasil Pengujian Menu Verifikasi Tim	261
Tabel 4.58 Hasil Pengujian Menu Verifikasi Laporan.....	262
Tabel 4.59 Hasil Pengujian Menu Verifikasi Analisis.....	263
Tabel 4.60 Hasil Pengujian Menu Tugas	265
Tabel 4.61 Hasil Pengujian Menu Laporan.....	267
Tabel 4.62 Hasil Pengujian Menu Pesan.....	268
Tabel 4.63 Hasil Pengujian Menu Profil.....	269
Tabel 4.64 Hasil Pengujian <i>Logout</i>	269

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Melansir dari hasil riset yang tertera pada Global Digital Report 2018 terdapat 4.021 miliar pengguna internet yang rata-rata menggunakannya selama enam jam sehari (Kempp, 2018). Banyaknya pengguna yang menggunakan internet karena kemudahan memenuhi kebutuhan yang diperoleh melalui penggunaan internet tersebut. Salah satu kebutuhan yang dipermudah adalah penggunaan *website* untuk kegiatan manajemen perusahaan.

Setiap perusahaan tentu melakukan berbagai kegiatan manajemen dalam perusahaannya agar perusahaan terus berjalan untuk mencapai tujuan perusahaan. Seperti pada perusahaan yang menawarkan jasa konsultasi yang menerima banyak proyek dari berbagai klien perlu untuk melakukan manajemen proyek. Dengan adanya manajemen proyek, perusahaan dapat memastikan proyek dari klien selesai tepat waktu dan mencapai target yang sesuai.

Manajemen proyek merupakan penggabungan dari perencanaan, penjadwalan, dan pengendalian kegiatan proyek yang berorientasi pada tujuan dalam konteks lingkungan organisasi yang spesifik dan unik (Sugiyanto, 2020). Pada manajemen proyek terdapat lima proses yang saling terhubung yaitu: inisiasi, perencanaan, pelaksanaan, pengontrolan dan penutupan yang terhubung dengan sembilan *knowledge areas* untuk menyukseskan suatu proyek. Sembilan *knowledge areas* terdiri atas *integration, scope, schedule, cost, quality, resource,*

communication, risk, procurement, stakeholder. Gambar 1.1 menunjukkan hubungan antara proses dan *knowledge areas*.

Knowledge Areas	Project Management Process Groups				
	Initiating Process Group	Planning Process Group	Executing Process Group	Monitoring and 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 Work 4.4 Manage Project Knowledge	4.5 Monitor and Control Project Work 4.6 Perform Integrated Change Control	4.7 Close Project or Phase
5. Project Scope Management		5.1 Plan Scope Management 5.2 Collect Requirements 5.3 Define Scope 5.4 Create WBS		5.5 Validate Scope 5.6 Control Scope	
6. Project Schedule Management		6.1 Plan Schedule Management 6.2 Define Activities 6.3 Sequence Activities 6.4 Estimate Activity Durations 6.5 Develop Schedule		6.6 Control Schedule	
7. Project Cost Management		7.1 Plan Cost Management 7.2 Estimate Costs 7.3 Determine Budget		7.4 Control Costs	
8. Project Quality Management		8.1 Plan Quality Management	8.2 Manage Quality	8.3 Control Quality	
9. Project Resource Management		9.1 Plan Resource Management 9.2 Estimate Activity Resources	9.3 Acquire Resources 9.4 Develop Team 9.5 Manage Team	9.6 Control Resources	
10. Project Communications Management		10.1 Plan Communications Management	10.2 Manage Communications	10.3 Monitor Communications	
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 Implement Risk Responses	11.7 Monitor Risks	
12. Project Procurement Management		12.1 Plan Procurement Management	12.2 Conduct Procurements	12.3 Control Procurements	
13. Project Stakeholder Management	13.1 Identify Stakeholders	13.2 Plan Stakeholder Engagement	13.3 Manage Stakeholder Engagement	13.4 Monitor Stakeholder Engagement	

Gambar 1.1 *Mapping* Manajemen Proyek (Institute, 2017)

Penelitian terkait sistem informasi manajemen proyek dan atau keterlambatan proyek pernah dilakukan oleh Abdurrasyid *et al.* (2019) mengenai implementasi metode PERT dan CPM pada proyek pembangunan kapal dalam bentuk sistem informasi berbasis web. Penelitian tersebut mendapatkan metode CPM dan PERT dapat membantu *project manager* dalam menentukan kegiatan kritis dan mengetahui estimasi waktu penyelesaian proyek dari hasil jalur kritis dan waktu optimis penyelesaian proyek agar proyek tidak mengalami keterlambatan. Penelitian lain juga dilakukan mengenai penerapan metode CPM pada proyek konstruksi oleh Iwawo, Tjakra, & Pratasis (2016). Penelitian menerapkan metode CPM untuk penjadwalan pada proyek pembangunan gedung baru Kompleks Eben Haezar yang menghasilkan bahwa metode CPM dapat mengetahui kegiatan yang kritis dan lintasan proyek sehingga mempermudah untuk dilakukan percepatan. Dan untuk perbandingan hasil durasi proyek dengan penjadwalan tanpa CPM, metode CPM mempunyai durasi proyek yang lebih cepat daripada durasi proyek yang tidak menerapkan CPM. Penelitian sejenis juga pernah dilakukan mengenai antipasti keterlambatan menggunakan metode *what if* oleh Unas, Hasyim, & Negara (2014). Melalui penelitian ini diketahui bahwa dengan metode *what if* keterlambatan proyek dapat dikejar sehingga kembali ke waktu normal tanpa adanya penambahan biaya. Penelitian sejenis lainnya pernah dilakukan oleh Fatimah (2019) mengenai antisipasi keterlambatan proyek menggunakan metode *what if analysis* dan *crash programming*. Penelitian tersebut mendapatkan metode CPM dapat mengetahui jalur kritis dan metode *what if analysis* lebih valid untuk mempercepat durasi penggeraan proyek daripada kurva S dengan hasil perhitungan

percepatan durasi kegiatan, total jam-orang, penambahan jumlah pekerja, dan nilai *float*. Sehingga proyek dapat diselesaikan secara tepat waktu. Selain itu terdapat penelitian lain mengenai analisis perencanaan proyek kapal cepat rudal dengan metode CPM dan *what if analysis* oleh Firmansyah (2017). Hasil dari penelitian ini didapatkan alternatif untuk mengantisipasi keterlambatan proyek dengan menambah tenaga kerja kerja dan waktu dari metode *what if analysis*.

Salah satu perusahaan yang juga mempunyai permasalahan pada manajemen proyek adalah PT. Triprima Karya. PT. Triprima Karya adalah perusahaan konsultan bangunan yang telah berdiri sejak 2003. Selama tujuh belas tahun, perusahaan tersebut telah mengerjakan setidaknya 180 proyek. Proyek dimulai dengan pengajuan kepada *owner* proyek. Setelah itu, perusahaan dan *owner* proyek akan melakukan persetujuan dan membuat kontrak untuk memulai proyek. Setelah itu, proyek akan dikerjakan oleh perusahaan. Selama pengerjaan proyek, direksi akan memantau dengan cara berkoordinasi langsung dengan team leader masing-masing proyek untuk melaporkan *progress* proyek.

Berdasarkan hasil wawancara dengan Direktur Utama PT. Triprima Karya Ir. Sarwo Sri pada 2 Maret 2020, perusahaan sulit untuk memantau pelaksanaan proyek saat perusahaan mengerjakan banyak proyek dalam waktu bersamaan, bagian direksi sulit melakukan proses pengontrolan *progress* pengerjaan proyek. Kesulitan dalam memantau proyek ini berdampak pada kesulitan untuk mengantisipasi keterlambatan penyelesaian proyek. Keterlambatan proyek yang pernah terjadi sekitar dua sampai empat minggu. Jika proyek terlambat, pembayaran proyek pun juga akan terhambat dan biaya yang dikeluarkan semakin

besar yang tentu akan merugikan perusahaan. Keterlambatan juga terjadi karena tidak adanya pengingat waktu tenggat dari kegiatan-kegiatan proyek. Masalah lain juga ditemukan pada pembuatan laporan sebagai syarat dokumentasi proyek untuk penutupan proyek, dimana pada pembuatan laporan harian, mingguan dan bulanan tidak dilakukan secara *real time*. Tim proyek perlu untuk mengarsip hasil kerjanya tiap minggu lalu melaporkannya kepada staf administrasi proyek yang selanjutnya akan diolah oleh staf administrasi proyek menggunakan *Microsoft office*. Hal tersebut cukup memakan waktu pembuatan laporan rata-rata selama dua minggu dan rentan terjadi *human error*. Berdasarkan latar belakang permasalahan, peneliti tertarik untuk melakukan penelitian pada PT. Triprima Karya dengan judul **“Rancang Bangun Sistem Informasi Manajemen Proyek Menggunakan Critical Path Method dan What If Analysis Berbasis Web (Studi Kasus: PT. Triprima Karya)”**.

1.2 Identifikasi Masalah

Berdasarkan latar belakang permasalahan dapat diidentifikasi sebagai berikut:

- a. Direksi yang meliputi direktur teknis dan kepala divisi proyek sulit untuk memantau *progress* penggerjaan proyek terutama saat perusahaan mengerjakan banyak proyek dalam waktu bersamaan.
- b. Masih terdapat proyek yang tidak selesai tepat waktu saat pelaksanaan proyek yang berakibat pada tertundanya pembayaran dari *owner* proyek dan biaya yang dikeluarkan semakin besar.

- c. Tidak adanya pengingat untuk pengontrolan jika kegiatan proyek sudah mendekati waktu tenggat.
- d. Pembuatan laporan untuk penutupan proyek tidak dilakukan secara *real time* dan penginputan data tidak otomatis sehingga rentan terjadi *human error*.

1.3 Rumusan Masalah

Berdasarkan latar belakang dan identifikasi masalah yang telah diuraikan, maka didapatkan rumusan masalah:

- a. Bagaimana merancang dan membangun sistem informasi manajemen proyek menggunakan *critical path method* dan *what if analysis* berbasis web?
- b. Bagaimana mengantisipasi keterlambatan proyek menggunakan metode *critical path method* dan *what if analysis*?

1.4 Batasan Masalah

Berdasarkan rumusan masalah yang telah diuraikan, batasan masalah penelitian ini adalah sebagai berikut:

- a. Penelitian ini dilakukan di PT. Triprima Karya yang berfokus pada manajemen proyek terutama pada *knowledge areas* manajemen jadwal, manajemen sumber daya manusia, dan manajemen komunikasi.
- b. Perancangan sistem hanya dilakukan pada proses penginputan data proyek, pengubahan data proyek, pembuatan jadwal proyek, perhitungan *critical path method* dan *what if analysis* dan pembuatan laporan.

- c. Analisis sistem berjalan menggunakan metode *Strength, Weakness, Opportunity, Threat* (SWOT).
- d. Metode pengembangan sistem menggunakan metode *Rapid Application Development* (RAD) pada tahap *Requirement Planning, Design System* dan *Implementation*.
- e. Penyajian data menggunakan *Unified Modeling Language* (UML) meliputi *Use Case Diagram, Activity Diagram, Sequence Diagram, Class Diagram, Component Diagram, Deployment Diagram* dengan *tools* draw.io versi 12.6.1.
- f. Desain *database* terbatas pada normalisasi dan *schema database*.
- g. Menggunakan PHP dan ReactJS sebagai alat pengkodean.
- h. Menggunakan MySQL sebagai *database management system* dan PHPMyAdmin untuk *tools* pengelolaan *database*.
- i. Menggunakan Figma untuk perancangan *user interface*.
- j. Tahap *testing* menggunakan *Black Box Testing*.

1.5 Tujuan Penelitian

Berdasarkan pada latar belakang dan rumusan masalah, maka tujuan dari penelitian ini adalah merancang dan membangun sistem informasi manajemen proyek berbasis *web* pada PT. Triprima Karya. Sedangkan tujuan khusus dari penelitian ini adalah:

- a. Membuat *dashboard* yang menampilkan *progress* masing-masing proyek yang sedang dikerjakan.

- b. Mengantisipasi keterlambatan proyek dengan pembuatan jadwal menggunakan *Critical Path Method* dan *what if analysis*.
- c. Membuat notifikasi jika kegiatan proyek sudah mendekati waktu tenggat kepada direktur teknis, *team leader* dan staf yang bersangkutan.
- d. Membuat sistem berbasis *web* dengan fitur tugas yang dapat di-update langsung oleh tim teknis dan *team leader* kapan saja dan dimana saja serta pembuatan laporan secara otomatis.

1.6 Manfaat Penelitian

Adapun manfaat dari penelitian bagi peneliti, instansi dan universitas sebagai berikut.

1.6.1. Bagi Peneliti

Manfaat yang peneliti dapatkan dalam rancang bangun sistem informasi dalam penelitian ini adalah:

- a. Dapat menerapkan ilmu pengetahuan dan kemampuan yang didapatkan selama perkuliahan khususnya rekayasa perangkat lunak, manajemen proyek.
- b. Menambah wawasan pengetahuan untuk peneliti mengenai rancang bangun sistem informasi manajemen proyek menggunakan *critical path method* dan *what if analysis* berbasis web pada PT. Triprima Karya.

1.6.2. Bagi Instansi

Manfaat bagi instansi dalam rancang bangun sistem informasi dalam penelitian ini sebagai referensi PT. Triprima Karya dalam manajemen proyek untuk perencanaan, *monitoring* dan penutupan proyek.

1.6.3. Bagi Universitas

Manfaat bagi universitas dalam rancang bangun sistem informasi dalam penelitian ini adalah:

- a. Mengetahui kemampuan mahasiswa dalam menerapkan ilmu yang didapat selama perkuliahan.
- b. Sebagai bahan evaluasi dan pertimbangan meningkatkan kinerja dan mutu pendidikan pada Universitas Islam Negeri Syarif Hidayatullah Jakarta, khususnya Program Studi Sistem Informasi Fakultas Sains dan Teknologi.

1.7 Metodologi Penelitian

Beberapa metode yang digunakan dalam penelitian ini adalah:

1.7.1 Metode Pengumpulan Data

Metodologi pengumpulan data terdiri atas:

- a. Studi Pustaka

Studi Pustaka dilakukan dengan cara membaca dan mempelajari sumber-sumber berupa buku, jurnal dan skripsi yang terkait dengan sistem informasi manajemen proyek, *critical path* method dan *what if analysis* serta sumber-sumber lain yang dapat mendukung penelitian.

- b. Studi Lapangan

- 1) Observasi

Metode pengumpulan data ini dilakukan untuk mengumpulkan data dan informasi dengan cara meninjau dan mengamati secara langsung bagaimana sistem yang sedang berjalan pada:

Tempat : PT. Triprima Karya
Jl. Arjuna VI DF 3 No. 6, Villa Pamulang, Pondok Benda,
Pamulang, Tangerang Selatan, Banten

Waktu : 07 Februari 2020 sampai dengan 6 Maret 2020

2) Wawancara

Metode ini dilaksanakan untuk mengetahui masalah apa yang sedang dihadapi dan apa saja yang dibutuhkan dalam Sistem Informasi ini dengan direktur utama dan karyawan pada:

Tempat : PT. Triprima Karya
Jl. Arjuna VI DF 3 No. 6, Villa Pamulang, Pondok Benda,
Pamulang, Tangerang Selatan, Banten

Waktu : 2 Maret 2020

Narasumber : Ir. Sarwo Sri

1.7.2 Metode Analisis Sistem Berjalan

Metode analisis sistem berjalan yang digunakan adalah metode analisis SWOT agar mengetahui kebutuhan sistem. Metode ini menjabarkan *strength* (kekuatan), *weakness* (kelemahan), *opportunity* (peluang), *threat* (ancaman). Melalui metode ini dapat memaksimalkan kekuatan dan peluang sistem serta meminimalkan kelemahan dan ancaman

1.7.3 Metode Pengembangan Sistem Usulan

Metode pengembangan sistem yang digunakan dalam melakukan rancang bangun sistem manajemen proyek adalah *Rapid Application Development* (RAD)

dengan alat bantu draw.io untuk memperlihatkan aliran proses dan data pada sistem yang dirancang.

Tahap-tahap pengembangan sistem dari tiap-tiap fase pengembangan sistem adalah: (Aswati & Siagian, 2016)

- *Requirements Planning* (Perencanaan Syarat-Syarat)

Dalam fase ini, pengguna dan penganalisis bertemu untuk mengidentifikasi tujuan-tujuan aplikasi atau sistem serta untuk megidentifikasi syarat-syarat informasi yang ditimbulkan dari tujuan-tujuan tersebut.

- *RAD Design Workshop* (Workshop Desain RAD)

Fase ini adalah fase untuk merancang dan memperbaiki yang bisa digambarkan sebagai *workshop*. Penganalisis dan pemrogram dapat bekerja membangun dan menunjukkan representasi visual desain dan pola kerja kepada pengguna.

- *Implementation* (Implementasi)

Setelah aspek-aspek bisnis dan non-teknis perusahaan disetujui serta sistem yang dibangun selesai baik sebagian maupun secara keseluruhan, kemudian dilakukan uji coba sistem dan selanjutnya diperkenalkan dan diaplikasikan kepada organisasi.

1.8 Sistematika Penulisan

Dalam penyusunan laporan ini pembahasan terbagi menjadi beberapa bab yang secara singkat diuraikan sebagai berikut:

BAB 1 PENDAHULUAN

Bab ini menjelaskan tentang latar belakang, rumusan masalah, tujuan dan sasaran penelitian, ruang lingkup dan batasan penelitian, metodologi penelitian, manfaat penelitian dan sistematika penulisan laporan hasil penelitian.

BAB 2 TINJAUAN PUSTAKA

Bab ini menjelaskan tentang teori yang mendukung penelitian perancangan sistem manajemen proyek dan penelitian terdahulu yang relevan.

BAB 3 METODOLOGI PENELITIAN

Bab ini memaparkan metode yang digunakan saat proses pelaksanaan penelitian, mencakup pengumpulan data dan analisis sistem, serta pengembangan sistem.

BAB 4 PEMBAHASAN

Bab ini berisi tentang hasil-hasil yang diperoleh dari hasil analisis dan perancangan sistem informasi manajemen proyek yang meliputi hasil tahapan-tahapan dalam merancang dan membangun sistem.

BAB 5 PENUTUP

Bab ini merupakan akhir penulisan laporan, dimana berdasarkan uraian-uraian yang telah dipaparkan sebelumnya dijabarkan ke dalam suatu kesimpulan dan saran.

Pada bab ini juga berisi jawaban dari rumusan masalah serta saran penelitian lebih lanjut.

BAB 2

TINJAUAN PUSTAKA

2.1 Pengertian Rancang Bangun

Rancang adalah kegiatan yang memiliki tujuan untuk mendesain sistem baru yang dapat menyelesaikan masalah-masalah yang dihadapi perusahaan yang diperoleh dari pemilihan alternatif sistem yang terbaik (Sutabri, 2012). Sedangkan menurut Muarie pada tahun 2015, rancang bangun adalah menciptakan dan membuat suatu aplikasi maupun sistem yang belum ada pada suatu instansi atau objek tersebut. Dan menurut Usep Teisnajaya pada tahun 2015, rancang bangun adalah program yang menentukan aktivitas pemrosesan informasi yang dibutuhkan untuk penyelesaian tugas-tugas khusus dari pemakai atau pengguna komputer (Maulani, Septiani, & Sahara, 2018). Mengacu pada pengertian sebelumnya dapat disimpulkan bahwa rancang bangun adalah kegiatan membuat sistem baru atau mengganti sistem yang telah ada berdasarkan hasil analisis ke dalam bahasa pemrograman untuk menyelesaikan permasalahan suatu instansi menjadi suatu aplikasi atau program.

2.2 Konsep Dasar Sistem Informasi

2.2.1 Pengertian Sistem

Secara sederhana sistem adalah suatu kumpulan atau himpunan dari unsur, komponen atau variabel terorganisir, saling berinteraksi, saling tergantung satu sama lain dan terpadu (Anggraeni & Irviani, 2017). Secara umum: (Sutabri, 2012)

- a. Sistem terdiri atas unsur-unsur;
- b. Unsur-unsur tersebut merupakan bagian terpadu sistem yang bersangkutan;
- c. Unsur sistem tersebut bekerjasama untuk mencapai tujuan sistem;
- d. Suatu sistem merupakan bagian dari sistem yang lebih besar.

Inti dari sistem adalah mempunyai elemen, mempunyai interaksi, terintegrasi dan mempunyai tujuan (Nafiudin, 2019). Jadi dapat disimpulkan, sistem merupakan kumpulan unsur atau elemen yang saling bersangkutan satu sama lain untuk mencapai tujuan tertentu.

2.2.2 Pengertian Informasi

Informasi merupakan hasil pengolahan atau pengklasifikasian data sehingga menjadi lebih berguna dan berarti bagi penerimanya. Informasi memberikan gambaran dari kejadian-kejadian nyata untuk proses pengambilan keputusan (Anggraeni & Irviani, 2017).

2.2.3 Pengertian Sistem Informasi

Sistem informasi adalah suatu sistem dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan informasi yang diperlukan untuk pengambilan keputusan (Anggraeni & Irviani, 2017).

2.2.4 Komponen Sistem Informasi

Dalam sistem informasi terdapat enam komponen yang disebut blok bangunan yang saling berinteraksi satu dengan yang lain, di antaranya: (Hutahaenan, 2015; Lestari & Amri, 2020; Sutabri, 2012)

a. *Input Block*

Blok ini adalah data yang masuk lalu diproses menjadi informasi ke dalam sistem informasi, termasuk metode dan media yang digunakan untuk menangkap data yang akan dimasukkan berupa dokumen-dokumen dasar.

b. *Model Block*

Blok ini adalah tempat pemrosesan input yang melibatkan kombinasi prosedur, logika dan model matematik yang akan memanipulasi data *input* dan data yang tersimpan di basis data dengan cara tertentu untuk menghasilkan keluaran yang diinginkan

c. *Output Block*

Blok ini berupa produk yang dihasilkan sistem informasi berupa informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

d. *Technology Block*

Teknologi dalam sistem informasi menjadi alat atau *tool box* dan mempunyai semua alur sistem informasi yaitu input, menjalankan fungsi model, menyimpan dan mengakses data yang masuk dan menghasilkan output yang mempunyai kualitas tinggi serta membantu pengendalian sistem secara keseluruhan. Teknologi terdiri dari tiga bagian utama yaitu *brainware*, *software*, dan *hardware*.

e. *Database Block*

Database atau basis data menjadi tempat tersimpannya kumpulan data yang masuk ke sistem dan saling berkaitan serta menggunakan perangkat lunak untuk memanipulasinya.

f. *Control Block*

Control dibutuhkan sistem informasi untuk mengendalikan hal-hal yang dapat merusak sistem dapat dicegah dan kesalahan-kesalahan dapat diatasi dengan cepat. Sumber lain mengatakan komponen sistem informasi yang berbeda, yaitu perangkat keras, perangkat lunak, *database*, telekomunikasi, manusia dan prosedur (Muslihudin & Oktafiano, 2016).

2.2.5 Klasifikasi Sistem Informasi

Sistem informasi dapat diklasifikasikan sebagai berikut: (Sutabri, 2012)

a. Sistem informasi berdasarkan level organisasi

Sistem informasi berdasarkan level organisasi dikelompokkan menjadi tiga level. Pertama, level operasional adalah mendukung manajer operasional untuk kegiatan rutin dan aliran transaksi organisasi. Kedua, level fungsional adalah mendukung *knowledge* dan *data workers* untuk melayani pengamatan, pengendalian, pembuatan keputusan dan aktivitas administratif level manajer tengah. Ketiga, level manajerial merupakan membantu perencanaan jangka panjang oleh manajer senior untuk perencanaan strategi dan pemecahan masalah.

b. Sistem informasi berdasarkan aktivitas manajemen

Sistem informasi berdasarkan aktivitas manajemen dikelompokkan menjadi perbankan, akademik, kesehatan, asuransi dan perhotelan.

c. Sistem informasi berdasarkan fungsionalitas bisnis

Sistem informasi berdasarkan fungsionalitas bisnis disebut sistem informasi manajemen yang dikelompokkan menjadi akuntansi, keuangan, manufaktur, pemasaran dan sumber daya manusia.

2.2.6 Tipe-Tipe Sistem Informasi

Sistem informasi berbasis komputer dibedakan menjadi beberapa tipe aplikasi, yaitu: (Muslihudin & Oktafiano, 2016)

a. *Transaction Processing System (TPS)*

Sistem informasi terkomputerisasi yang dikembangkan untuk memproses sejumlah besar data untuk transaksi bisnis rutin.

b. *Management Information System (MIS)*

Sistem informasi pada level manajemen yang berfungsi untuk membantu perencanaan, pengendalian, dan pengambilan keputusan dengan menyediakan resume rutin dan laporan-laporan tertentu. MIS mengambil data mentah dari TPS dan mengubahnya menjadi data-data yang lebih berarti.

c. *Decision Support System (DSS)*

Sistem informasi pada level manajemen yang mengkombinasikan data dan model analisis canggih atau peralatan data analisis untuk mendukung pengambilan keputusan yang semiterstruktur dan tidak terstruktur.

d. *Expert System and Artificial Intelligence (ES & AL)*

Sistem informasi yang merepresentasikan pengetahuan dari gambaran cara seorang ahli dalam mendeteksi masalah.

2.3 Sistem Informasi Manajemen

Sistem informasi manajemen terdiri atas tiga unsur yaitu sistem, informasi dan manajemen dimana bagaimana manajemen informasi berbasis komputer dalam sebuah sistem. Jadi sistem informasi manajemen adalah sistem informasi berbasis komputer yang menghasilkan informasi untuk semua tingkatan manajerial sehingga dapat membantu dalam pengambilan keputusan dan pengawasan (Mulyani, 2016; Nafiudin, 2019).

2.4 Pengertian Proyek

Menurut Soeharto yang dimaksud proyek adalah suatu kegiatan sementara yang berlangsung dalam jangka waktu dan alokasi sumber daya tertentu yang dimaksudkan untuk menghasilkan tugas yang kualitas mutunya telah digariskan dengan jelas. Lingkup dari tugas yang dimaksud dapat berupa pembangunan bangunan, pembuatan produk baru atau pelaksanaan dan pengembangan penelitian (Husodo, 2017). Lebih lanjut Husodo menyatakan bahwa setiap proyek memiliki tujuan khusus dan dalam setiap pencapaian tujuan tersebut terdapat hal-hal yang perlu diperhatikan seperti anggaran yang akan dialokasikan, waktu pelaksanaan, serta mutu dari hasil proyek yang dilaksanakan. Ketiga hal ini juga sering diungkapkan sebagai tiga kendala (*triple constraint*) (Husodo, 2017). Terdapat pendapat yang dikemukakan oleh Jason Westland, bahwa proyek berbeda dengan

operasi bisnis biasa, karena proyek memiliki: a) keunikan, di mana setiap proyek satu beda dengan proyek lainnya, b) batasan waktu telah ditentukan untuk memulai dan menyelesaikan proyek dengan jelas, c) anggaran alokasi sudah ditentukan sebelumnya untuk memenuhi kebutuhan pelaksanaan proyek, d) sumber yang dimiliki terbatas, dan e) meliputi elemen resiko karena ketidakpastiannya (Sitanggang, Simamarta, & Luthan, 2019).

2.5 Manajemen Proyek

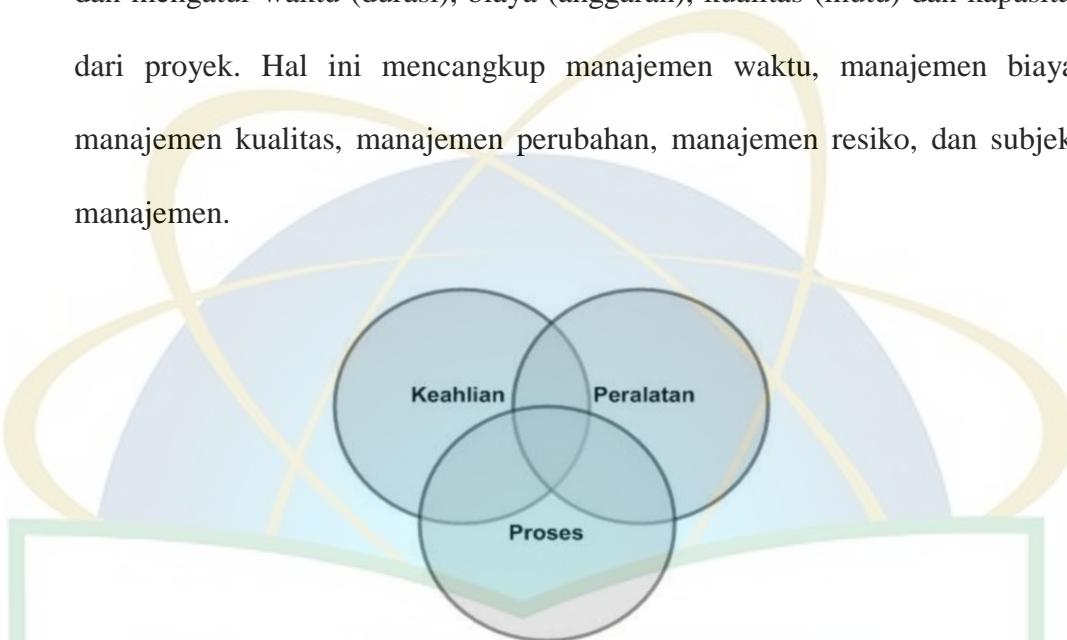
Manajemen proyek merupakan penggabungan dari perencanaan, penjadwalan, dan pengendalian kegiatan proyek yang berorientasi pada tujuan dalam konteks lingkungan organisasi yang spesifik dan unik. Tujuan utama yang harus dipenuhi meliputi kinerja, biaya, dan sasaran waktu, sementara pada saat yang sama mengendalikan atau mempertahankan ruang lingkup proyek pada tingkat yang benar (Sugiyanto, 2020).

Dalam manajemen proyek dibutuhkan tiga hal penting agar menghasilkan suatu proyek yang sukses, ketiga hal tersebut adalah: (Sugiyanto, 2020)

- a. Keahlian, untuk dapat mengurangi resiko dalam pelaksanaan sebuah proyek dan meningkatkan keberhasilannya dibutuhkan pengetahuan, keahlian serta pengalaman khusus dalam pengerjaan proyek tersebut.
- b. Peralatan, berbagai macam peralatan dibutuhkan untuk digunakan oleh manajer proyek untuk meningkatkan kesempatan mereka untuk sukses. Contoh dari peralatan yang dibutuhkan antara lain, model dokumen, model perangkat lunak,

perangkat lunak yang direncanakan, pengecekan audit, serta formulir yang direka ulang.

- c. Proses, terdapat berbagai proses dan teknik yang diperlukan untuk menguasai dan mengatur waktu (durasi), biaya (anggaran), kualitas (mutu) dan kapasitas dari proyek. Hal ini mencangkup manajemen waktu, manajemen biaya, manajemen kualitas, manajemen perubahan, manajemen resiko, dan subjek-manajemen.



Gambar 2.1 Komponen Manajemen Proyek (Sugiyanto, 2020)

Pada umumnya proyek memiliki empat siklus utama: (Sitanggang *et al.*, 2019)

- a. Permulaan Proyek

Terdapat beberapa langkah siklus yang dihadapi pada tahap permulaan sebuah proyek, yaitu:

- Melakukan studi kelayakan. Langkah pertama yang harus dilakukan adalah dengan melakukan studi kelayakan sehingga masalah utama dari proyek yang akan dikerjakan dapat diidentifikasi dan diberikan solusi untuk memecahkan masalah tersebut;

- Mengidentifikasi ruang lingkup. Langkah ini dilakukan agar mengetahui kedalaman serta luasnya proyek yang akan dikerjakan;
- Mengidentifikasi kiriman. Langkah ini dilakukan agar mengetahui produk atau layanan yang harus disediakan;
- Mengidentifikasi pemangku kepentingan proyek. Langkah ini dilakukan untuk mengetahui siapa saja yang mempengaruhi proyek dan kebutuhan apa saja yang mereka butuhkan;
- Mengembangkan kasus bisnis. Dari kriteria yang sudah diidentifikasi sebelumnya, langkah selanjutnya yang dilakukan adalah untuk membandingkan biaya dan manfaat potensial bagi proyek untuk ditentukan apakah akan bergerak maju.

b. Perencanaan Proyek

Setelah area dari proyek ditetapkan pada tahap sebelumnya, maka proses perencanaan proyek adalah tahap selanjutnya. Dalam tahap ini terdapat beberapa langkah, yaitu:

- Merencanakan kegiatan-kegiatan, tugas-tugas tertentu, ketergantungan, dan waktu perencanaan dari proyek tersebut;
- Merencanakan sumber tenaga kerja, peralatan serta materi yang diperlukan;
- Merencanakan aspek finansial seperti biaya tenaga kerja, peralatan serta materi yang telah ditentukan sebelumnya;
- Merencanakan kualitas yang didalamnya termasuk kualitas sasaran, jaminan dan pengukuran kontrol;

- Merencanakan risiko yang menekankan tentang adanya potensi risiko dan aksi apa yang akan diambil untuk mengurangi risiko tersebut;
- Merencanakan persetujuan yang menyebutkan tentang kriteria yang harus dipenuhi;
- Merencanakan komunikasi yang dapat memberikan informasi yang diperlukan untuk memberitahukan pemegang saham;
- Merencanakan perantaraan dengan mengidentifikasi sumber dari proyek yang disediakan oleh pemasok luar.

c. Pelaksanaan Proyek

Langkah-langkah yang terdapat pada tahapan pelaksanaan proyek mencangkap hal-hal berikut ini:

- Penetapan tugas dan mengatur alur kerja. Dengan disusunnya segala aspek secara terperinci, maka dapat dipastikan setiap pekerjaan dikerjakan oleh tim yang sesuai dengan jumlah pekerjaan yang tepat;
- Briefing tugas dengan anggota tim. Menjelaskan segala tugas/kegiatan yang harus dilakukan oleh anggota tim dengan memberikan panduan yang diperlukan seperti bagaimana tugas tersebut diselesaikan, serta mengatur pelatihan yang terkait dengan proses jika diperlukan;
- Komunikasi dengan anggota tim, klien, dan manajemen tingkat atas. Dengan melakukan komunikasi yang baik, maka semua informasi akan selalu diperbarui kepada setiap pemangku kepentingan proyek di semua tingkatan;

- Mengawasi kualitas pekerjaan. Pengawasan dilakukan untuk memastikan bahwa tiap anggota tim bekerja tepat waktu dan sasaran kualitas yang telah ditetapkan dapat terpenuhi;
 - Mengelola anggaran. Setiap pengeluaran harus dimonitor serta menjaga proyek yang dilaksanakan agar tetap sesuai dengan jalurnya, terutama pada hal aset dan sumber daya.
- d. Penutupan Proyek

Dalam penutupan proyek, langkah-langkah yang dilakukan adalah:

- Menganalisis kinerja proyek. Pada langkah ini, dilakukan penentuan apakah tujuan proyek terpenuhi dan apakah masalah yang telah diidentifikasi di awal telah terselesaikan;
- Menganalisis kinerja tim. Di langkah ini, anggota tim akan dievaluasi bagaimana pekerjaan yang mereka lakukan, apakah memenuhi tujuan yang telah ditetapkan;
- Mendokumentasikan penutupan proyek. Setelah proyek telah diselesaikan, harus dipastikan bahwa semua aspek proyek telah diselesaikan tanpa ada yang tersisa dan menyediakan laporan kepada pemangku kepentingan;
- Melakukan tinjauan pasca implementasi. Menganalisis proyek dengan mempertimbangkan pelajaran apa yang dapat diambil untuk proyek serupa di masa yang akan datang;
- Akuntansi untuk anggaran yang digunakan dan tidak terpakai. Ketika terdapat sisa anggaran, maka dapat dialokasikan untuk proyek di masa yang akan datang.



2.5.1 *Process Group*

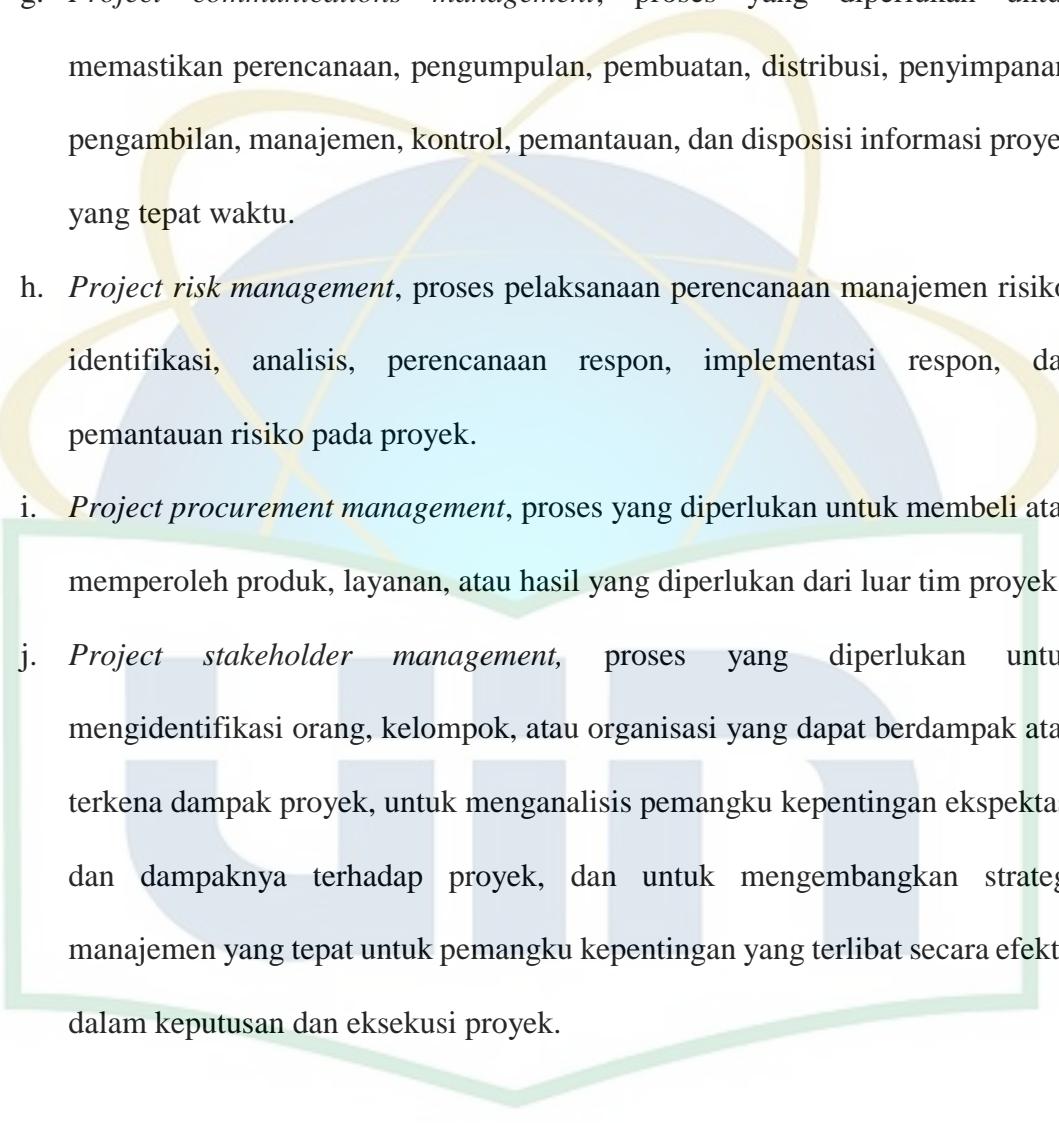
Proses dalam manajemen dikelompokkan menjadi 5 bagian: (Institute, 2017)

- Initiating*, proses untuk menentukan proyek baru atau fase baru dari proyek yang ada, dengan mendapatkan otorisasi untuk memulai proyek.
- Planning*, proses untuk menentukan ruang lingkup proyek, menentukan tujuan, dan menentukan tindakan yang diperlukan untuk mencapai tujuan dari proyek.
- Executing*, proses untuk menyelesaikan pekerjaan dari rencana manajemen proyek untuk memenuhi kebutuhan proyek.
- Monitoring and Controlling*, proses untuk melacak, me-review, dan mengatur progress dan performa proyek.
- Closing*, proses untuk menutup atau mengakhiri proyek secara resmi.

2.5.2 *Knowledge Area*

Knowledge Area adalah area yang diidentifikasi dari manajemen proyek yang ditentukan oleh persyaratan pengetahuannya dan dijelaskan dalam istilah proses, praktik, input, *output*, alat, dan teknik komponennya. Berdasarkan PMBOK, area pengetahuan yang banyak digunakan pada proyek-proyek adalah: (Institute, 2017)

- a. *Project integration management*, proses-proses untuk mengidentifikasi, mendefinisikan, mengkombinasikan, menyatukan dan mengkoordinasi berbagai proses.
- b. *Project scope management*, proses-proses untuk memastikan proyek mencakup semua pekerjaan yang diperlukan dan hanya pekerjaan yang diperlukan, untuk menyelesaikan proyek.
- c. *Project schedule management*, proses yang diperlukan untuk mengelola penyelesaian proyek.
- d. *Project cost management*, proses yang terlibat dalam perencanaan, memperkirakan biaya penganggaran, pembiayaan, pendanaan, pengelolaan, dan pengendalian sehingga proyek dapat diselesaikan sesuai anggaran yang disetujui.
- e. *Project quality management*, proses untuk menggabungkan kebijakan mutu mengenai perencanaan, pengelolaan, dan pengendalian proyek dan persyaratan kualitas produk, dalam rangka memenuhi ekspektasi pemangku kepentingan.

- 
- f. *Project resource management*, proses untuk mengidentifikasi, memperoleh, dan mengelola sumber daya yang diperlukan untuk keberhasilan penyelesaian proyek.
 - g. *Project communications management*, proses yang diperlukan untuk memastikan perencanaan, pengumpulan, pembuatan, distribusi, penyimpanan, pengambilan, manajemen, kontrol, pemantauan, dan disposisi informasi proyek yang tepat waktu.
 - h. *Project risk management*, proses pelaksanaan perencanaan manajemen risiko, identifikasi, analisis, perencanaan respon, implementasi respon, dan pemantauan risiko pada proyek.
 - i. *Project procurement management*, proses yang diperlukan untuk membeli atau memperoleh produk, layanan, atau hasil yang diperlukan dari luar tim proyek.
 - j. *Project stakeholder management*, proses yang diperlukan untuk mengidentifikasi orang, kelompok, atau organisasi yang dapat berdampak atau terkena dampak proyek, untuk menganalisis pemangku kepentingan ekspektasi dan dampaknya terhadap proyek, dan untuk mengembangkan strategi manajemen yang tepat untuk pemangku kepentingan yang terlibat secara efektif dalam keputusan dan eksekusi proyek.

2.5.3 Penjadwalan Proyek

Seperti yang telah diungkapkan sebelumnya bahwa waktu pelaksanaan merupakan aspek penting dalam pelaksanaan sebuah proyek. Maka diperlukan perencanaan akan waktu pelaksanaan sehingga dapat mencapai penyelesaian

proyek yang tepat pada waktunya. Penjadwalan proyek merupakan pengurutan serta pembagian waktu atas seluruh kegiatan proyek. Hal-hal yang termasuk dalam pembuatan jadwal ini seperti penghitungan waktu setiap tahapan produksi pada proyek tersebut dilakukan, bagaimana hubungan antara satu kegiatan dengan kegiatan lainnya, berapa sumber daya yang diperlukan pada setiap kegiatan tersebut (Husodo, 2017).

Dikutip dari Prabowo, bahwa terdapat beberapa tahap pada penjadwalan proyek guna proyek tersebut memiliki tujuan terutama dalam hal waktu. Berikut adalah tahapan yang terdapat dalam penjadwalan: (Sugiyanto, 2020)

a. *Planning*

Tahap pertama yang dilakukan dalam menyusun penjadwalan proyek adalah dengan *planning*/perencanaan. Perencanaan dilakukan untuk dapat menentukan aktivitas apa saja yang nantinya akan dilakukan dalam proyek tersebut. Dalam melakukan perencanaan terdapat beberapa langkah, sebagai berikut:

1) *Work Breakdown Structure (WBS)*

Agar proyek dapat dikontrol dengan lebih mudah, maka perlu dilakukannya *Work Breakdown Structure (WBS)* yaitu pemecahan suatu proyek yang bersifat kompleks menjadi susunan aktivitas-aktivitas pekerjaan.

2) Penyusunan rencana kerja proyek

Langkah ini dilakukan untuk menentukan urutan aktivitas pekerjaan yang dilakukan dalam sebuah proyek. Hal ini dilakukan untuk menggambarkan hubungan antar berbagai aktivitas pekerjaan yang terjadi dalam pelaksanaan sebuah proyek.

3) Estimasi durasi aktivitas

Yang dimaksud dengan estimasi durasi aktivitas adalah memperkirakan banyaknya waktu yang dibutuhkan untuk menyelesaikan suatu aktivitas. Angka yang didapat untuk estimasi ini berdasarkan hasil perhitungan volume pekerjaan dan produktivitas tenaga kerja.

4) Menentukan metode penjadwalan

Dalam metode penjadwalan proyek terdapat beberapa metode yang dapat digunakan dalam mengelola waktu dan sumber daya proyek. Setiap metode memiliki kelebihan dan kekurangannya sendiri, sehingga pertimbangan dalam menggunakan metode mana didasarkan atas kebutuhan serta hasil yang diinginkan terhadap kinerja penjadwalan.

b. *Monitoring*

Tahap selanjutnya dalam menyusun penjadwalan proyek adalah *Monitoring* yaitu kegiatan pengamatan secara langsung akan jalannya aktivitas di lapangan. Pengamatan yang dilakukan ini bertujuan untuk memastikan bahwa durasi aktivitas sesuai dengan rencana awal. Dalam melakukan *monitoring* hal penting yang harus dilakukan adalah membuat catatan/dokumen terhadap *actual work* dan durasi. Hasil dari catatan/dokumen tersebut nantinya akan dibandingkan dengan *as plan schedule* yang merupakan langkah *controlling*.

c. *Controlling*

Ketika terjadi penyimpangan dalam sebuah proyek, maka dibutuhkan standar atau tolak ukur sebagai pembanding, alat ukur kinerja, serta tindakan koreksi yang tepat untuk menangani penyimpangan tersebut. Dalam proses pengendalian

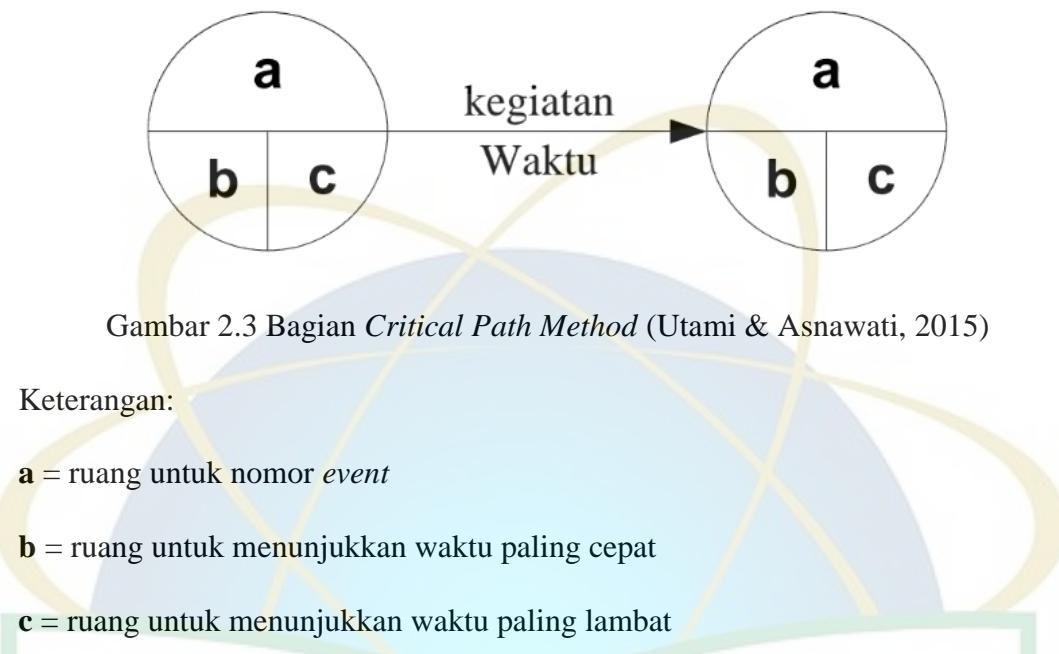
terdapat beberapa kegiatan yang dilakukan, seperti pengawasan, pemeriksaan serta tindakan koreksi, yang dilakukan selama proses implementasi.

2.6 Critical Path Method

Critical Path Method atau metode jalur kritis merupakan suatu bentuk penjadwalan yang dapat memperkirakan waktu yang dibutuhkan untuk melaksanakan setiap kegiatan dan dapat menentukan prioritas kegiatan yang harus diperhatikan oleh pelaksana proyek agar kegiatan tersebut dapat diselesaikan sesuai rencana. Dengan metode ini dapat diketahui aktivitas atau kegiatan yang mempunyai tingkat kepekaan tinggi terhadap keterlambatan proyek agar dapat dikontrol (Sugiyanto, 2020). Terdapat beberapa istilah: (Utami & Asnawati, 2015)

- a) E (*earliest event occurrence time*), saat tercepat terjadinya suatu peristiwa.
- b) L (*latest event occurrence time*), saat terlambat yang masih diperbolehkan terjadinya suatu peristiwa.
- c) ES (*earliest activity start time*), waktu mulai paling awal suatu kegiatan.
- d) EF (*earliest activity finish time*), waktu selesai paling awal suatu kegiatan dan akan menjadi ES untuk kegiatan selanjutnya.
- e) LS (*latest activity start time*), waktu mulai paling lambat kegiatan boleh dimulai tanpa memperlambat proyek secara keseluruhan.
- f) LF (*latest activity finish time*), waktu paling lambat kegiatan diselesaikan tanpa memperlambat proyek secara keseluruhan.
- g) t (*activity duration time*), durasi atau waktu yang dibutuhkan untuk suatu kegiatan. Dapat berupa hari, minggu, bulan dan lainnya.

- h) *Float* atau *slack*, kelonggaran waktu yang didapat dari selisih waktu mulai dan selesai.



Tahapan dalam *critical path method* adalah sebagai berikut: (Utami & Asnawati, 2015)

1. Menentukan kegiatan dan durasinya.
2. Menghitung EF masing-masing *event* dari awal sampai akhir dengan rumus:

$$EF(i - j) = ES(i - j) + t(i - j) \quad (2.1)$$

Jika terdapat banyak kegiatan sebelumnya, maka ES yang digunakan adalah yang paling besar.

3. Menghitung LS masing-masing *event* dari *event* terakhir sampai pertama dengan rumus:

$$LS(i - j) = LF(i - j) - t \quad (2.2)$$

Jika terdapat banyak kegiatan terpecah, maka LS berikutnya yang digunakan adalah yang paling kecil.

4. Menghitung Total *Slack* atau *Float* dengan rumus:

$$TS = LS - ES \text{ atau } LF - EF \quad (2.3)$$

Kegiatan yang mempunyai *total slack* 0 merupakan kegiatan kritis.

5. Menentukan jalur kritis dengan mengurutkan kegiatan yang mempunyai *total slack* 0.

2.7 *What If Analysis*

2.7.1 Pengertian *What If Analysis*

Analisis *What If* adalah salah satu metode antisipasi keterlambatan proyek, merupakan studi yang bertujuan menyelesaikan aktivitas proyek dalam kondisi terlambat dan mengejar aktivitas yang seharusnya sudah selesai dikerjakan agar selesai tepat pada waktunya (Widaningsih, Maukar, & Shinta, 2017). Analisis *What If* dapat digunakan sebagai acuan bagi manajer proyek untuk dapat segera mengambil keputusan yang tepat dan efektif bila terjadi ketidaksesuaian jadwal aktual dengan jadwal rencana (Firmansyah, 2017).

2.7.2 Tahapan Analisis *What If* pada *Critical Path Method*

Analisis *What If* pada model CPM menanyakan “Bagaimana bila terjadi keterlambatan pada salah satu aktivitas?”. Jika salah aktivitas terlambat, maka diperlakukan percepatan aktivitas yang dapat dilakukan dengan cara menambah

jumlah pekerja atau waktu kerja pada proyek (Maulana, 2018). Tahapan untuk menganalisis *what if* pada *critical path method* yaitu: (Firmansyah, 2017)

1. Membuat jadwal menggunakan *Critical Path Method* yang berisi:

- a. kegiatan,
- b. durasi (d),
- c. *float*,
- d. jumlah pekerja (n),
- e. jam kerja normal (H),
- f. *man hour* ($\sum mh$), yang didapat dari $n \times H \times d$
- g. kegiatan pengikut (*successor*)

2. Membuat skenario keterlambatan pada masing-masing aktivitas. Seperti terlambat 10% atau 20% atau 30% durasi semula dengan persamaan:

$$delay = \%keterlambatan \times durasi\ proyek \quad (2.4)$$

3. Memeriksa apakah keterlambatan yang terjadi pada aktivitas tersebut akan menyebabkan keterlambatan proyek secara keseluruhan. Dengan menghitung *delay* proyek:

$$delay'p = delay + float \quad (2.5)$$

4. Menghitung durasi percepatan masing-masing aktivitas pengikut.

$$d's = ds + float - delay'p \quad (2.6)$$

dan memeriksa kemungkinan aktivitas pengikut dapat dilakukan percepatan, dengan batasan:

- a. Durasi percepatan lebih besar dari pada nilai *float* aktivitas pengikut.

$$1/2d's \geq ds \quad (2.7)$$

- b. Durasi percepatan aktivitas pengikut tidak lebih durasi rencananya.

$$d's < ds \quad (2.8)$$

Jika aktivitas pengikut memenuhi batasan, maka aktivitas tersebut dapat dilakukan percepatan.

5. Melakukan perhitungan penambahan jumlah pekerja dengan persamaan:

$$\Delta n = \frac{\sum mh}{d's \times H} - n \quad (2.9)$$

Keterangan:

Δn = jumlah pekerja tambahan

$\sum mh$ = jumlah jam-orang untuk menyelesaikan aktivitas

$d's$ = durasi percepatan

H = jam kerja normal

n = jumlah pekerja yang direncanakan

6. Melakukan perhitungan penambahan waktu dengan rumus:

$$\Delta H = \frac{\sum mh}{d's \times n} - H \quad (2.10)$$

Keterangan:

ΔH = jumlah waktu tambahan

$\sum mh$ = jumlah jam-orang untuk menyelesaikan aktivitas

$d's$ = durasi percepatan

H = jam kerja normal

n = jumlah pekerja yang direncanakan

7. Menghitung jumlah tambahan biaya pada aktivitas yang terlambat dengan cara:

$$\text{Biaya tambahan per aktivitas} = \Delta n \times \Delta H \times \text{upah per jam} \quad (2.11)$$

2.8 Metode Analisis Sistem SWOT

Metode *Strength* (kekuatan), *Weakness* (kelemahan), *Opportunity* (peluang), *Threat* (ancaman) atau SWOT merupakan identifikasi yang bersifat sistematis dari faktor kekuatan dan kelemahan organisasi serta peluang dan ancaman lingkungan luar strategi yang menyajikan kombinasi terbaik di antara keempat faktor. Dengan metode ini perusahaan dapat memanfaatkan kekuatan yang dimilikinya untuk mengambil keuntungan dari peluang-peluang yang ada, sekaligus memperkecil atau mengatasi kelemahan yang dimilikinya untuk menghindari ancaman yang ada (Hamali, 2016). Dalam metode SWOT mempunyai empat unsur utama yaitu: (Wibowo, 2019)

1. *Strength* atau Kekuatan

Analisis terhadap unsur kekuatan yang dimiliki oleh perusahaan. Seperti analisis kelebihan yang dimiliki perusahaan seperti segi teknologi, kualitas hasil produksi, lokasi strategis atau unsur kekuatan lainnya.

2. *Weakness* atau Kelemahan

Analisis terhadap unsur kelemahan yang dimiliki oleh perusahaan. Untuk mengetahui kelemahan yang dimiliki, perusahaan dapat melakukan perbandingan dengan pesaing dan atau melakukan testimoni kepada konsumen perusahaan.

3. *Opportunity* atau Peluang/Kesempatan

Analisis terhadap unsur-unsur apa saja yang memungkinkan bisnis mampu bertahan dan diterima di masyarakat, baik dalam jangka pendek maupun panjang.

4. *Threat* atau ancaman

Analisis terhadap unsur-unsur yang menentukan apakah perusahaan dapat bertahan atau tidak di masa depan untuk jangka pendek maupun jangka panjang. Misalnya banyaknya pesaing, ketersediaan sumber daya, jangka waktu minat konsumen dan lain sebagainya.

Matriks SWOT dapat digunakan untuk menggambarkan secara jelas peluang dan ancaman eksternal yang dihadapi perusahaan, dan disesuaikan dengan kekuatan dan kelemahan yang dimilikinya. Matriks ini dapat menghasilkan empat set kemungkinan alternatif strategik seperti pada Gambar 2.6 (Hamali, 2016).

SWOT	STRENGTHS	TREATHS (T)
	Tentukan 5-10 faktor kekuatan internal	Tentukan 5-10 faktor kelemahan internal
STRENGTHS	STRATEGI SO	STRATEGI WO
Tentukan 5-10 faktor peluang eksternal	Ciptakan strategi yang menggunakan kekuatan untuk memanfaatkan peluang	Ciptakan strategi yang meminimalkan kelemahan untuk memanfaatkan peluang
TREATHS (T)	STRATEGI ST	STRATEGI WT
Tentukan 5-10 faktor ancaman eksternal	Ciptakan strategi yang menggunakan kekuatan untuk mengatasi ancaman	Ciptakan strategi yang meminimalkan kelemahan dan menghindari ancaman

Gambar 2.4 Matriks Metode SWOT (Hamali, 2016)

Penjelasan dari matriks SWOT, sebagai berikut:

1. Strategi SO (*Strength-Opportunity*): Strategi ini dibuat berdasarkan jalan pikiran perusahaan, yaitu dengan memanfaatkan seluruh kekuatan untuk merebut dan memanfaatkan peluang sebesar-besarnya.
2. Strategi ST (*Strength-Threat*): Strategi ini menggunakan kekuatan yang dimiliki perusahaan untuk mengatasi ancaman.
3. Strategi WO (*Weakness-Opportunity*): Strategi ini diterapkan berdasarkan pemanfaatan peluang yang ada dengan cara meminimalkan kelemahan yang ada.
4. Strategi WT (*Weakness-Threat*): Strategi ini didasarkan pada kegiatan yang bersifat *defensive* dan berusaha meminimalkan kelemahan yang ada serta menghindari ancaman.

2.9 Metode Pengembangan Sistem

2.9.1 Pengertian RAD

Dari Pengembangan Aplikasi oleh James Martin Rapid. *Rapid Application Development* (RAD) adalah sebuah pengembangan siklus hidup dirancang untuk memberikan pengembangan lebih cepat dan kualitas yang lebih tinggi daripada siklus hidup tradisional. Hal ini dirancang untuk mengambil keuntungan dari pengembangan perangkat lunak yang kuat. RAD digunakan untuk membangun aplikasi Sistem Informasi besar dari jenis yang terjadi dalam setiap bisnis yang besar (Aswati & Siagian, 2016).

2.9.2 Tahapan RAD

Metode RAD memiliki 3 tahapan sebagai berikut:

- a. Rencana Kebutuhan (*Requirement Planning*): *User* dan *analyst* melakukan pertemuan untuk mengidentifikasi tujuan dari sistem dan kebutuhan informasi untuk mencapai tujuan. Pada tahap ini merupakan hal terpenting dengan adanya keterlibatan dari kedua belah pihak.
- b. Proses Desain Sistem (*Design System*): Pada tahap ini keaktifan *user* yang terlibat menentukan untuk mencapai tujuan karena pada proses ini melakukan proses desain dan melakukan perbaikan-perbaikan apabila masih terdapat ketidaksesuaian desain antara *user* dan *analyst*. Seorang *user* dapat langsung memberikan komentar apabila terdapat ketidaksesuaian pada desain, merancang sistem dengan mengacu pada dokumentasi kebutuhan *user* yang dibuat pada tahap sebelumnya. Keluaran dari tahapan ini adalah spesifikasi *software* yang meliputi organisasi sistem secara umum, struktur data dan yang lain.
3. Implementasi (*Implementation*): Tahapan ini adalah tahapan programmer yang mengembangkan desain suatu program yang telah disetujui oleh *user* dan *analyst*. Sebelum diaplikasikan pada suatu organisasi terlebih dahulu dilakukan proses pengujian terhadap program tersebut apakah ada kesalahan atau tidak. Pada tahap ini *user* biasa memberikan tanggapan akan sistem yang sudah dibuat serta mendapat persetujuan mengenai sistem tersebut.



Gambar 2.5 Tahapan Metode RAD (Kosasi, 2015)

2.9.3 Keunggulan dan Kelemahan RAD

Keunggulan Model RAD:

- Setiap fungsi mayor dapat dimodulkan dalam waktu tertentu kurang dari 3 bulan dan dapat dibicarakan oleh tim RAD yang terpisah dan kemudian diintegrasikan sehingga waktunya lebih efisien.
- RAD mengikuti tahapan pengembangan sistem seperti umumnya, tetapi mempunyai kemampuan untuk menggunakan kembali komponen yang ada (reusable object) sehingga pengembang tidak perlu membuat dari awal lagi dan waktu lebih singkat.

Kelemahan Model RAD:

- Proyek yang besar dan berskala, RAD memerlukan sumber daya manusia yang memadai untuk menciptakan jumlah tim yang baik.
- RAD menuntut pengembang dan pelanggan memiliki komitmen dalam aktivitas *rapid fire* yang diperlukan untuk melengkapi sebuah sistem dalam

waktu yang singkat. Jika komitmen tersebut tidak ada maka proyek RAD akan gagal.

Alasan penggunaan RAD:

- a. Apabila menggunakan RAD untuk mendapatkan suatu desain yang dapat diterima oleh konsumen dan dapat dikembangkan dengan mudah.
- b. Apabila menggunakan RAD untuk memberikan batasan-batasan pada suatu sistem supaya tidak mengalami perubahan.
- c. Apabila menggunakan RAD untuk menghemat waktu, dan kalau memungkinkan bisa menghemat biaya serta menghasilkan produk yang berkualitas.

2.9.4 Pengujian Sistem

Black box testing adalah tipe testing yang memperlakukan perangkat lunak yang tidak diketahui kinerja internalnya. Sehingga para tester memandang perangkat lunak layaknya sebuah “kotak hitam” yang tidak penting dilihat isinya, tapi cukup dikenai proses testing bagian luar untuk memastikan kebutuhan pengguna (Wicaksono, 2017). Keuntungan yang diperoleh dari *black box testing* adalah: (Wicaksono, 2017)

- a. Anggota tim *tester* tidak harus dari seseorang yang memiliki kemampuan teknis di bidang pemrograman.
- b. Kesalahan dari perangkat lunak atau *bug* seringkali ditemukan oleh komponen tester yang berasal dari pengguna.

- c. Hasil dari *black box testing* dapat memperjelas kontradiksi ataupun kerancuan yang mungkin timbul dari eksekusi sebuah perangkat lunak.
- d. Proses *testing* dapat lebih cepat daripada *white box testing*.

Beberapa teknik *testing* yang tergolong dalam *black box testing*: (Wicaksono, 2017)

- a. *Equivalence Partitioning*, tiap inputan data dikelompokkan ke dalam grup tertentu, kemudian dibandingkan *output*-nya.
- b. *Boundary Value Analysis*, melakukan penginputan data melebihi dari batasan sebuah data.
- c. *Cause Effect Graph*, menghubungkan sebab dari sebuah inputan dan akibatnya pada output yang dihasilkan.
- d. *Random Data Selection*, melakukan penginputan data dengan menggunakan nilai acak.
- e. *Feature Test*, melakukan pengujian terhadap spesifikasi atau fitur dari perangkat lunak yang telah selesai dikerjakan.

2.10 Unified Modeling Language

Unified Modeling Language (UML) adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan sebuah standar penulisan atau semacam blueprint dimana didalamnya termasuk sebuah bisnis proses, penulisan kelas-kelas dalam sebuah bahasa yang spesifik (Hendini, 2016; Prihandoyo, 2018). Menurut Abdurohman pada perancangan UML, sistem didefinisikan sebagai sekumpulan objek yang

memiliki atribut dan metode. Atribut adalah variabel-variabel yang melekat pada objek. Sedangkan metode adalah fungsi-fungsi yang dapat dilakukan oleh objek (Nazli, 2018).

2.10.1 *Use Case Diagram*

Use Case Diagram merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat yang memberikan gambaran fungsi yang diharapkan ada dalam sebuah sistem, dan menggambarkan hubungan antara aktor dan sistem. Di dalam *use case* terdapat aktor yang merupakan sebuah gambaran entitas dari manusia atau sebuah sistem yang melakukan pekerjaan di sistem (Hendini, 2016; Prihandoyo, 2018). Diagram *use case* memberikan tinjauan umum fungsionalitas sistem atau proses bisnis dari perspektif pengguna. Cara di mana pengguna "menggunakan" sistem adalah titik awal untuk membuat diagram *use case* (Unhelkar, 2018). *Use case diagram* mempunyai 3 komponen yaitu: (Hutauruk, 2019)

a. **Sistem**

Menyatakan batasan sistem dalam relasi dengan aktor-aktor yang menggunakannya (di luar sistem) dan fitur-fitur yang harus disediakan (dalam sistem).

b. **Aktor**

Aktor adalah segala hal diluar sistem yang akan menggunakan sistem tersebut untuk melakukan sesuatu. Bisa merupakan manusia, sistem, atau *device* yang memiliki peranan dalam keberhasilan operasi dari sistem.

c. *Use Case*

Use Case sendiri adalah gambaran fungsional dari sebuah sistem. Dengan demikian, antara konsumen dan juga pengguna pada sistem tersebut, akan mengerti atau paham mengenai fungsi sistem yang tengah dibangun.

Use case diagram juga mempunyai beberapa relasi, yaitu: (Hutauruk, 2019)

a. *Association*

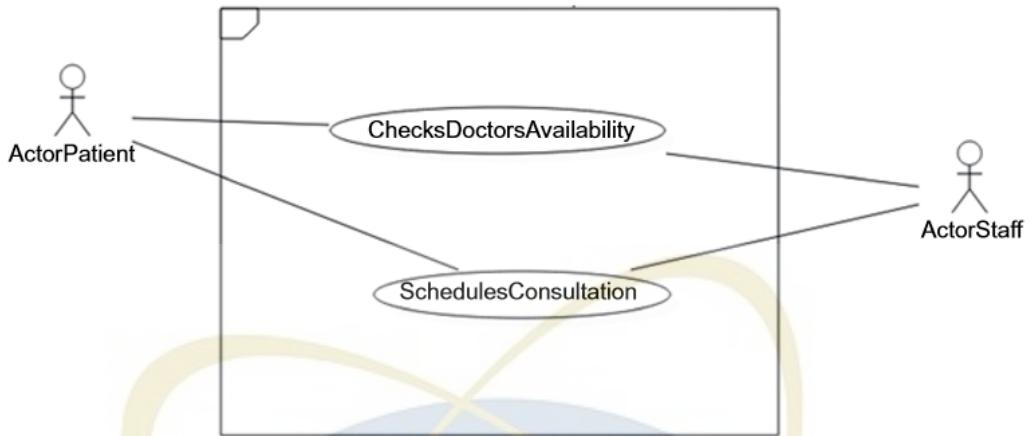
Teknik mengidentifikasi interaksi yang dilakukan oleh *actor* tertentu dengan *use case* tertentu. Hal ini digambarkan dengan garis antara aktor terhadap *use case* tersebut.

b. *Generalization*

Mendefinisikan relasi antara dua aktor atau dua *use case* yang mana salah satunya meng-*inherit* dan menambahkan atau *override* sifat dari yang lainnya.

c. *Dependency*

Dependency ini terbagi menjadi 2 macam, yaitu *include* dan juga *extend*. *Include*, berfungsi untuk mengidentifikasi hubungan antara 2 *use case*, dimana *use case* yang satu akan memanggil *use case* yang lainnya. *Extend*, apabila pemanggilan, memerlukan kondisi tertentu maka akan berlaku dependensi.



Gambar 2.6 Contoh *Use Case Diagram* (Unhelkar, 2018)

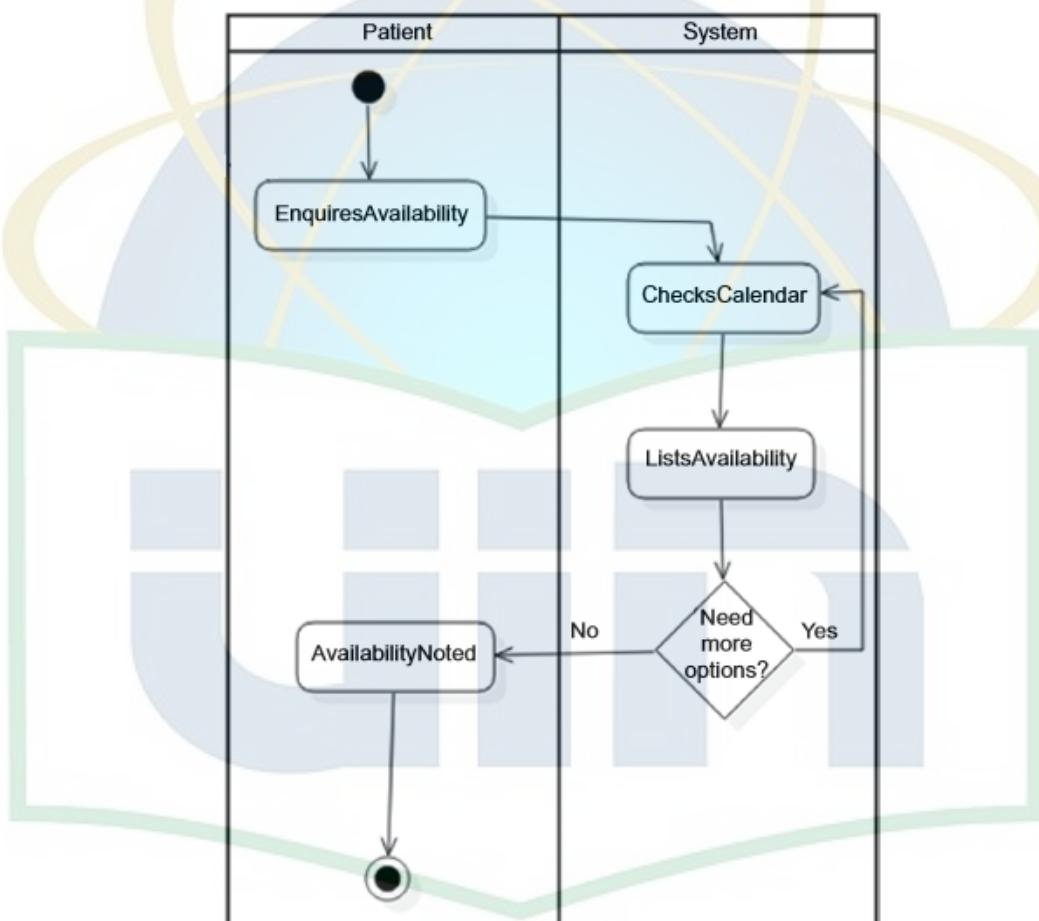
2.10.2 *Activity Diagram*

Diagram *activity diagram* menggambarkan model aliran dari aktivitas dalam sistem (Hendini, 2016; Prihandoyo, 2018). Detailnya, aliran dalam *use case* yang menggambarkan interaksi pengguna normal dan alternatif dan pengecualian dimodelkan dengan sangat baik oleh diagram aktivitas ini (Unhelkar, 2018).

Simbol-simbol dalam *activity diagram* yaitu: (Hendini, 2016)

- Start Point*, adalah simbol yang menggambarkan awal dimulainya suatu aliran kerja.
- End Point*, adalah simbol yang menggambarkan akhir dari suatu aliran kerja.
- Activities*, adalah simbol yang menggambarkan aktivitas atau pekerjaan yang dilakukan dalam aliran kerja.
- Fork* atau percabangan, adalah simbol yang menunjukkan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu.

- e. *Join* atau penggabungan, adalah simbol yang menunjukkan adanya dekomposisi.
- f. *Decision Points*, adalah simbol yang menggambarkan pilihan untuk pengambilan keputusan, *true* atau *false*.
- g. *Swimlane*, adalah simbol yang menggambarkan pembagian *activity diagram* untuk menunjukkan siapa melakukan apa.



Gambar 2.7 Contoh *Activity Diagram* (Unhelkar, 2018)

2.10.3 Class Diagram

Class Diagram merupakan kelas, definisi, dan hubungan mereka. Kelas dan entitas dari ruang masalah juga entitas teknis rinci dalam ruang solusi. Atribut dan operasi yang mendefinisikan kelas termasuk dalam diagram kelas ini. Hubungan dalam diagram kelas menggambarkan bagaimana kelas berinteraksi, berkolaborasi, dan mewarisi dari kelas lain. Kelas juga bisa mewakili tabel relasional, antarmuka pengguna, dan pengontrol (Unhelkar, 2018).

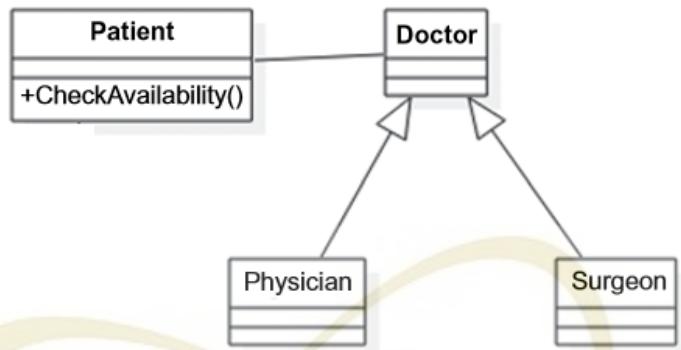
Class Diagram secara khas meliputi: Kelas (*Class*), Relasi *Associations*, *Generalization* dan *Aggregation*, attribut (*Attributes*), operasi (*operation/method*) dan *visibility*, tingkat akses objek eksternal kepada suatu operasi atau atribut. Hubungan antar kelas mempunyai keterangan yang disebut dengan *Multiplicity* atau *Cardinality* (Hendini, 2016).

Tabel 2.1 *Multiplicity Class Diagram*

Multiplicity	Penjelasan
1	Satu dan hanya satu
0..*	Boleh tidak ada atau 1 atau lebih
1..*	1 atau lebih
0..1	Boleh tidak ada, maksimal 1
n..n	Batasan antara

(Hendini, 2016)

Berikut contoh dari *class diagram*.



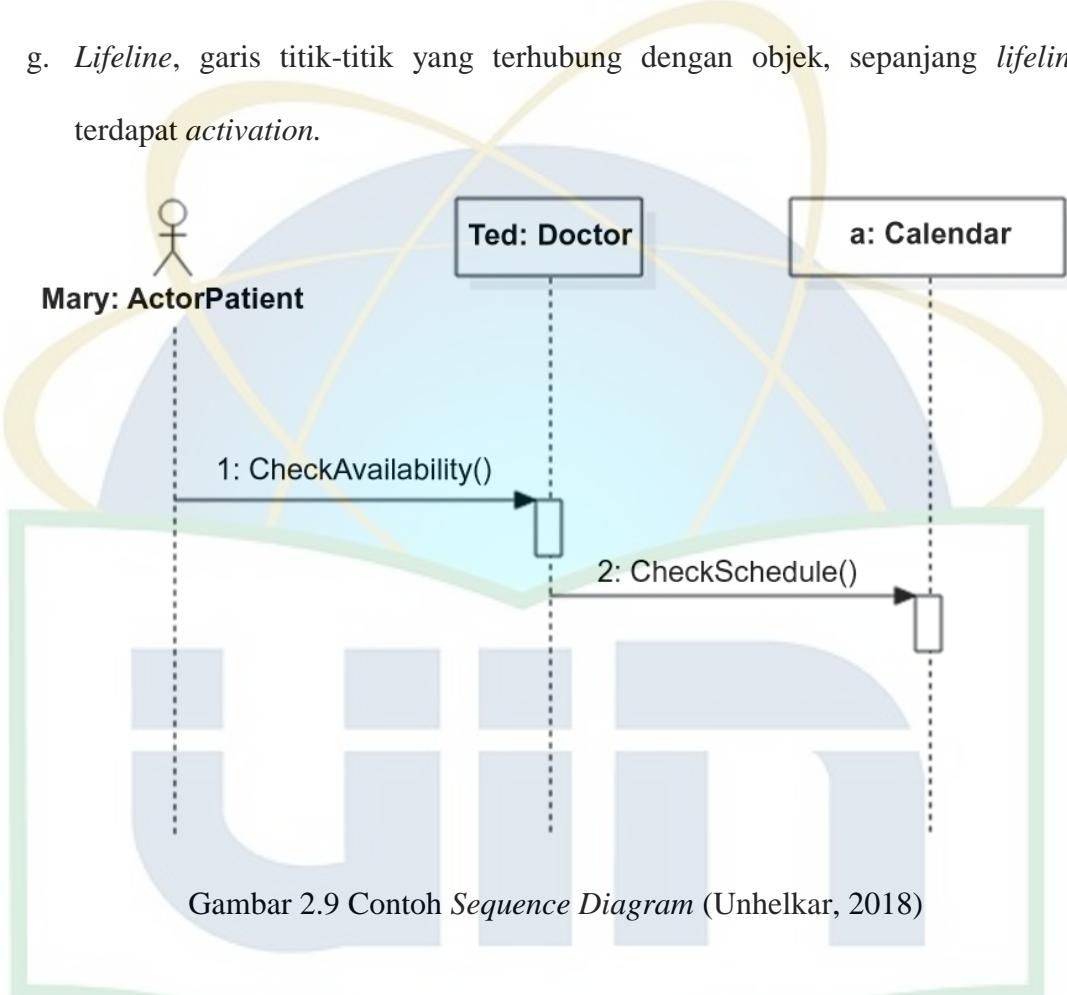
Gambar 2.8 Contoh *Class Diagram* (Unhelkar, 2018)

2.10.4 Sequence Diagram

Sequence Diagram menggambarkan model interaksi antara objek berdasarkan *timeline* mereka. Objek dapat secara spesifik ditampilkan pada diagram ini atau dapat menjadi objek anonim milik kelas. Urutan eksekusi pesan antara objek saat *runtime* dimodelkan dengan baik oleh diagram *sequence* (Unhelkar, 2018). Simbol-simbol dalam *sequence diagram* yaitu: (Hendini, 2016)

- Entity class*, merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data.
- Boundary class*, berisi kumpulan kelas yang menjadi interfaces atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan form entry dan form cetak.
- Control class*, suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek.
- Message*, simbol mengirim pesan antar *class*.

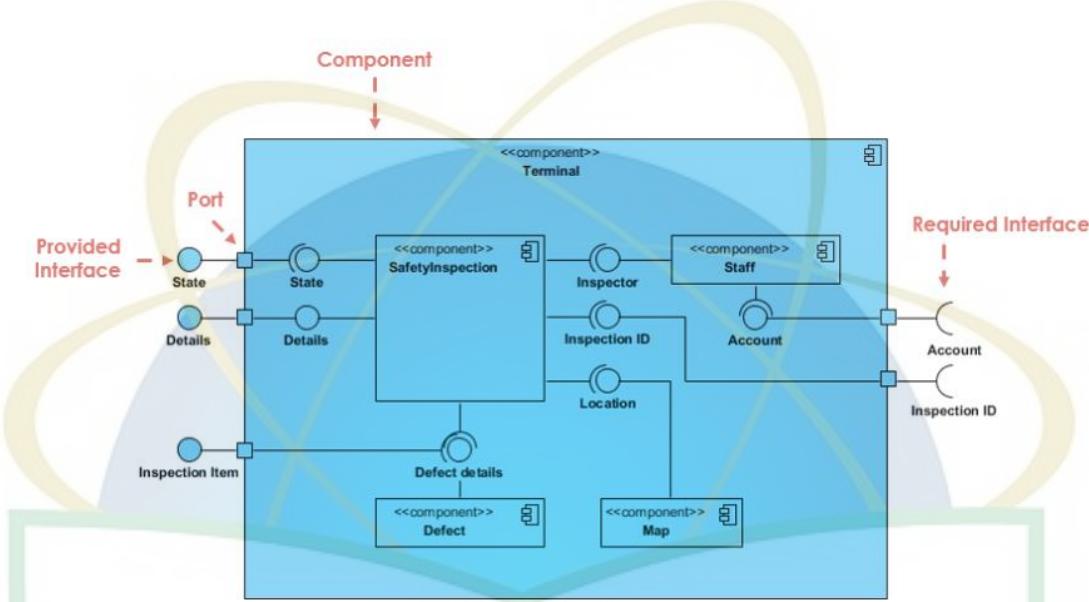
- e. *Recursive*, menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.
- f. *Activation*, mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivasi sebuah operasi.
- g. *Lifeline*, garis titik-titik yang terhubung dengan objek, sepanjang *lifeline* terdapat *activation*.



2.10.5 Component Diagram

Component Diagram adalah menggambarkan komponen-komponen dan hubungan antar komponen dalam sistem secara struktural. Pada *component diagram* ditunjukkan komposisi, organisasi dan dependensi antara komponen perangkat lunak. Komponen bertanggung jawab untuk satu tujuan yang jelas dalam seluruh sistem dan hanya berinteraksi dengan elemen-elemen penting lainnya.

Contoh komponen diantaranya *user interface*, *business processing*, *data*, *security*, *source code*, dan *executable file* yang dilepas ke pengguna (Unhelkar, 2018; Visual Paradigm, 2020a). Dalam *component diagram* terdapat beberapa bagian yaitu: (Visual Paradigm, 2020a)



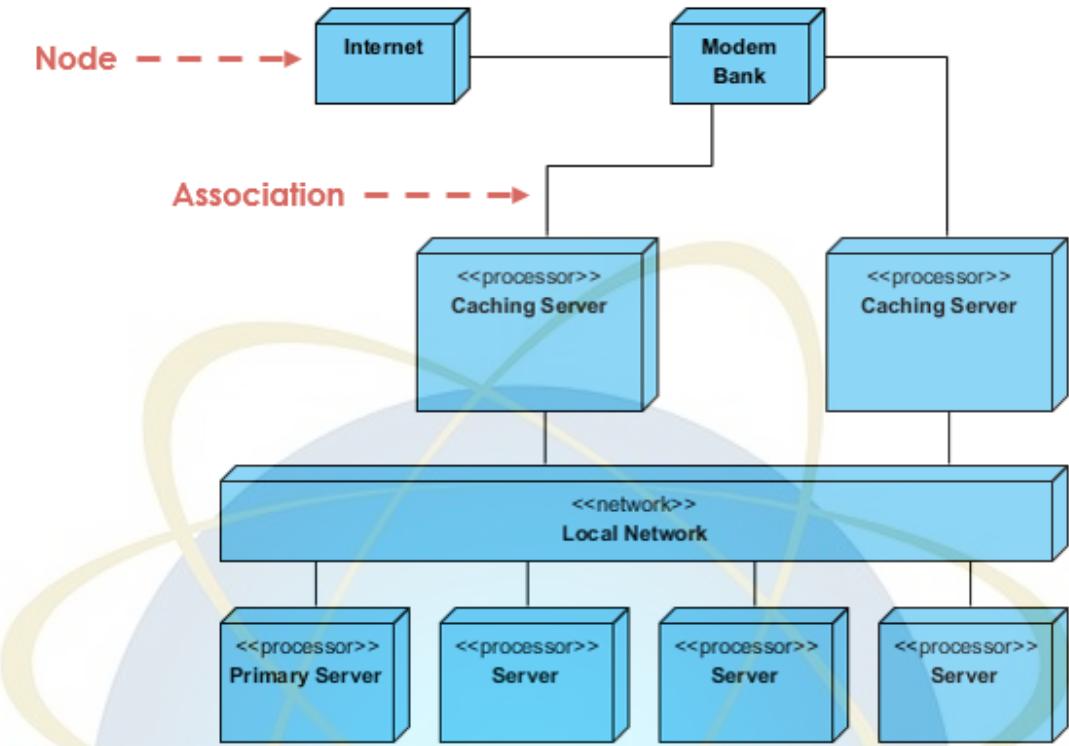
Gambar 2.10 Contoh *Component Diagram* (Visual Paradigm, 2020a)

- Component*, merupakan bagian modular dari sistem yang menggambarkan isi dari sistem. *Component* digambarkan dalam bentuk kotak.
- Interface*, terdapat dua simbol dalam *interface* yaitu *provided interface* dan *required interface*. *Provided interface* merupakan lingkaran sempurna yang menggambarkan *interface* yang disediakan oleh suatu komponen. Sedangkan, *required interface* merupakan setengah lingkaran yang menggambarkan *interface* yang dibutuhkan oleh suatu komponen.
- Subsystem*, merupakan *component* yang lebih khusus.

- d. *Port*, digunakan untuk membantu mengekspos *required interface* dan *provided interface* dari sebuah *component*.
- e. *Relationships*, terdapat beberapa notasi di dalamnya yaitu:
 - 1) *Association*, menspesifikasikan hubungan semantik antara masing-masing ujung yang saling terhubung.
 - 2) *Composition*, bentuk agregasi yang lebih kuat dimana jika komposit dihapus, semua bagiannya ikut terhapus.
 - 3) *Aggregation*, suatu jenis asosiasi yang salah satu ujungnya ditandai dibagikan sebagai jenis agregasi, yang berarti bahwa ia memiliki agregasi bersama.
 - 4) *Constraint*, kondisi atau batasan dalam bahasa yang dapat dibaca mesin untuk mendeklarasikan beberapa semantik elemen.
 - 5) *Dependency*, hubungan dimana suatu elemen model memerlukan elemen model lain untuk spesifikasi atau implementasi.
 - 6) *Links*, hubungan dari umum ke lebih spesifik.

2.10.6 Deployment Diagram

Deployment diagram adalah diagram yang menggambarkan bentuk fisik dari sistem yaitu *hardware* atau perangkat keras. Pada diagram ini ditunjukkan komponen yang dieksekusi pada *node* dan bagaimana *node* dapat menjalankan *software*. Dengan adanya diagram ini dapat membantu dalam memutuskan kebutuhan operasional seperti kecepatan dan volume, lokasi dan keamanan *node*, dan cara penggunaan *executable* di seluruh jaringan (Unhelkar, 2018).



Gambar 2.11 Contoh *Deployment Diagram* (Visual Paradigm, 2020b)

2.11 Konsep Dasar Basis Data

2.11.1 Basis Data

Basis data terdiri atas dua kata, basis dan data. Basis dapat diartikan sebagai markas, gudang, tempat berkumpul. Sedangkan data adalah fakta yang mewakili suatu objek (Yanto, 2016). Basis data adalah sekumpulan data yang saling berhubungan yang digunakan secara bersama-sama oleh berbagai aplikasi atau pemakai pada suatu organisasi dan harus dibangun dengan mengikuti kaidah-kaidah perancangan yang sistematis dan langkah-langkah yang telah ditetapkan, baik berdasarkan teori maupun berdasarkan pengalaman yang telah terbukti berhasil

(Pahlevi, 2013). Basis data mempunyai beberapa tujuan, di antaranya: (Yanto, 2016)

- a. Kecepatan dan kemudahan dalam menyimpan atau melakukan perubahan atau menampilkan kembali data.
- b. Efisiensi ruang penyimpanan dengan menerapkan sejumlah pengkodean atau membuat relasi antar kelompok data yang saling berhubungan.
- c. Keakuratan dengan pengkodean atau pembentukan relasi antar data bersama dengan penerapan aturan dan keunikan data.
- d. Ketersediaan data yang dapat diakses pada lokasi lain.
- e. Kelengkapan data dapat diakomodasi melalui basis data.
- f. Keamanan dapat diterapkan pada basis data dengan menentukan pemakaian basis data dan objek-objek di dalamnya serta jenis-jenis operasi apa saja yang boleh dilakukannya.
- g. Kebersamaan pemakaian memungkinkan basis data dapat dikelola *multiuser*.

Komponen penting dalam sistem basis data terdiri atas: (Yanto, 2016)

- a. Data, informasi yang disimpan dalam suatu struktur tertentu yang terintegrasi.
- b. *Hardware*, perangkat keras berupa komputer dengan media penyimpanan yang digunakan untuk menyimpan data.
- c. Sistem operasi, program yang mengaktifkan dan memfungsikan sistem komputer.
- d. Basis data, inti sistem basis data untuk menyimpan data serta struktur sistem basis data.

- e. *Database Management System*, perangkat lunak yang digunakan untuk melakukan pengelolaan basis data.
- f. *User*, pengguna yang menggunakan data yang tersimpan dan terkelola.
- g. Aplikasi lainnya, program yang dibuat untuk memberikan *interface* kepada *user* sehingga lebih mudah dan terkontrol dalam mengakses basis data.

2.11.2 Database Management System (DBMS)

Database Management System (DBMS) adalah data yang saling berhubungan yang dikelompokkan dalam sebuah tabel atau beberapa tabel dan sebuah aplikasi program yang mengatur cara mengakses data tersebut untuk menyediakan sebuah cara menyimpan dan mengambil informasi basis data secara efisien dan nyaman (Widodo & Kumianingtyas, 2017). Secara konsep, dalam *Database Management System* (DBMS) dilakukan hal-hal berikut: (Yanto, 2016)

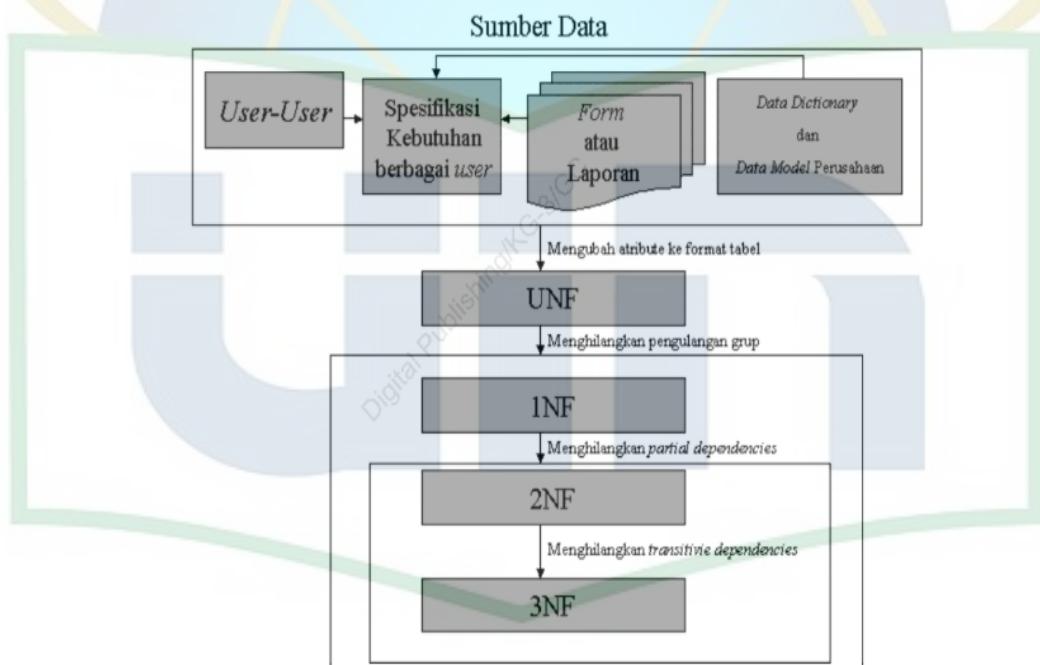
- a. *User* melakukan pengaksesan basis data untuk informasi yang diperlukan menggunakan bahasa *query* biasa disebut dengan SQL.
- b. DBMS menerima permintaan dari *user* dan menganalisis permintaan tersebut.
- c. DBMS memeriksa skema eksternal *user*, skema konseptual dan struktur penyimpanan.
- d. DBMS mengeksekusi operasi-operasi yang diperlukan untuk memenuhi permintaan *user*.

2.11.3 Normalisasi Basis Data

Normalisasi adalah suatu teknik formal yang dapat digunakan dalam perancangan basis data dengan pendekatan *bottom-up* yang menghasilkan sekumpulan hubungan yang mendukung data dengan sifat-sifat yang diinginkan dan memenuhi kebutuhan. Hubungan tersebut mencakup: (Indrajani, 2015)

- a. Minimal jumlah atribut.
- b. Atribut dengan hubungan logika yang menjelaskan mengenai *functional dependencies*.
- c. Minimal duplikasi tiap atribut.

Terdapat beberapa bentuk normal yang biasa digunakan, yaitu: (Indrajani, 2015)



Gambar 2.12 Bentuk Normalisasi (Indrajani, 2015)

a. *Unnormalized Form* (UNF)

Pada tabel UNF masih bersi satu atau lebih grup yang berulang karena pada tabel ini data dipindahkan dari sumber informasi tanpa perubahan atau kondisi-kondisi tertentu.

b. *First Normal Form* (1NF)

Pada tabel 1NF, setiap baris dan kolom berisi satu dan hanya satu nilai. Dari tabel UNF ke 1NF:

- Tentukan satu atribut sebagai kunci untuk tabel *unnormalized*.
- Identifikasikan grup yang berulang dalam table *unnormalized* yang berulang untuk kunci atribut.
- Hapus grup yang berulang dengan cara masukkan data yang semestinya ke kolom kosong pada baris data berulang atau ganti data yang ada dengan menulis ulang dari kunci atribut yang sesungguhnya.

c. *Second Normal Form* (2NF)

Pada tabel 2NF, relasi dalam 1NF dan setiap atribut selain *primary-key* bersifat *fully functionally dependent* pada *primary-key*. *Fully functional dependent* yaitu A dan B merupakan atribut sebuah relasi. B dikatakan *fully dependent* terhadap A jika B *functionally dependent* pada A tetapi tidak *proper subset* dari A. Dari tabel 1NF ke 2NF:

- Identifikasi *primary key* untuk relasi 1NF.
- Identifikasi *functional dependencies* dalam relasi.
- Hapus *partial dependencies* terhadap *primary key*.

d. *Third Normal Form (3NF)*

Pada tabel 3NF, tidak terdapat atribut selain *primary key* yang bersifat *transitively dependent* pada *primary key*. *Transitive dependency* adalah kondisi dimana A, B, C merupakan atribut sebuah relasi, maka A ke B dan B ke C, sehingga *transitively dependent* pada A melalui B. Dari tabel 2NF ke 3NF:

- Identifikasi *primary key* dalam relasi 2NF.
- Identifikasi *functional dependencies* dalam relasi.
- Hapus *transitive dependencies* terhadap *primary key*.

2.12 ReactJS

ReactJS adalah *library* untuk pengembangan *user interface* berbasis JavaScript. ReactJS dikembangkan oleh Facebook. *Library* ini pertama kali muncul pada Mei 2013 dan sekarang merupakan salah satu *library frontend* yang paling banyak digunakan untuk pengembangan web (Deshpande, 2021). Airbnb, Facebook, Uber, Pinterest, Instagram adalah contoh *platform* yang menggunakan ReactJS. ReactJS mempunyai beberapa keunggulan, di antaranya: (Deshpande, 2021)

- a. Memudahkan untuk membuat *web* yang dinamis karena hanya membutuhkan sedikit kode dan lebih banyak menggunakan fungsionalitas.
- b. Menggunakan *Virtual DOM* yang membuat pembuatan web lebih cepat karena hanya meng-*update item* yang berubah.
- c. Menggunakan *reusable component* atau komponen yang dapat digunakan kembali. Dalam satu aplikasi biasanya terdapat beberapa komponen. Masing-

masing komponen mempunyai logikanya sendiri. Dengan *reusable component*, komponen-komponen tersebut dapat digunakan kembali pada seluruh aplikasi.

- d. Menggunakan aliran data searah sehingga memudahkan untuk mencari kesalahan.
- e. Mudah untuk dipelajari karena menggabungkan konsep dasar HTML dan JavaScript.
- f. Dapat digunakan untuk pengembangan aplikasi *web* dan *mobile*.

2.13 PHP

Hypertext Preprocessor (PHP) adalah bahasa pemrograman untuk *scripting* dari sisi *server*. PHP biasa digunakan untuk membuat *query* dari *database* agar dapat mengirim *feedback* ke tampilan *web* melalui HTML. Hal tersebut dapat terjadi karena PHP dapat disisipkan ke dalam HTML sehingga web dapat menjadi dinamis dan interaktif (EMS, 2016; Jubilee Enterprise, 2018; Minnick, 2016). Dalam PHP terdapat beberapa tipe data seperti *string*, *integer*, *float*, *Boolean*, *array*, *object*, *null*. Beberapa kelebihan PHP antara lain: (EMS, 2016)

- a. Tidak melakukan kompilasi dalam penggunaannya.
- b. Didukung oleh banyak *web server*.
- c. Lebih mudah dalam sisi pengembangan karena banyak *developer* yang siap membantu dalam pengembangan.
- d. Mudah dipahami karena banyak referensi.

- e. Bahasa pemrograman *open source* yang dapat digunakan pada berbagai mesin seperti Linux, Unix, Macintosh, Windows dan dapat dijalankan secara *runtime* melalui *console*, serta dapat menjalankan perintah-perintah sistem.

2.14 MySQL

MySQL adalah *server* untuk melayani *database* sehingga user dapat menginput data menggunakan form HTML, kemudian diolah PHP agar bisa disimpan ke *database* MySQL (Jubilee Enterprise, 2018). MySQL adalah DBMS yang *open source* dengan dua bentuk lisensi, *Free Software* (perangkat lunak bebas) dan *Shareware* (perangkat lunak berpemilik yang penggunaannya terbatas).

Kelebihan MySQL di antaranya: (Yasin, 2019)

- a. Mendukung integrasi dengan bahasa pemrograman lain.
- b. Tidak membutuhkan RAM besar.
- c. *Open Source* sehingga tidak memerlukan biaya.
- d. Struktur tabel fleksibel dan mudah dipakai.
- e. Tipe data bervariasi seperti *float*, *integer*, *date*, *char*, *text*, *timestamp*, *double*, dan lain sebagainya.
- f. Keamanan terjamin karena menerapkan beberapa lapisan keamanan dan mengatur hak akses *user* dengan enkripsi *password*.

2.15 Tools

2.15.1 draw.io

draw.io adalah sebuah website yang didesain khusus untuk menggambarkan diagram secara online. Semua fitur yang ada pada *web* ini dapat digunakan hanya dengan bermodalkan *browser* yang mendukung HTML 5. Draw.io menawarkan kemudahan untuk membuat diagram tanpa batasan jumlah diagram yang dibuat, dapat diintegrasikan dengan Google Drive, Github, Google Docs, Google Slide, One Drive dan GitLab. Hasil diagram juga dapat diekspor ke dalam bentuk *file* PNG/JPG/SVG/ XML. Beberapa diagram yang disediakan di antaranya: *Flowchart, BPMN, UML, Engineering Diagram, Network Diagram, Sequence Diagram, Org Chart, Mindmap*. Diagram-diagram tersebut dapat dikustomisasi sesuai kebutuhan (Lathif, 2019).

2.15.2 Figma

Figma adalah aplikasi desain berbasis *cloud* dan alat *prototyping* untuk membuat desain UI/UX yang interaktif untuk proyek digital. Figma memungkinkan pengguna berkolaborasi dalam proyek darimana saja. Figam dapat digunakan untuk berbagai desain grafis seperti *wireframe websites, mobile app interfaces, prototype designs, crafting social media posts*, dan lainnya (Pramudita *et al*, 2021).

2.16 Studi Penelitian Sejenis

Studi literatur sejenis digunakan sebagai pembanding ataupun acuan dalam penelitian dengan mengumpulkan beberapa penelitian sejenis yang pernah

dilakukan sebelumnya yaitu dari lima skripsi dan lima jurnal yang dijabarkan pada Tabel 2.2.



Tabel 2.2 Penelitian Sejenis

No	Penulis	Uraian	Metode	Tools	Hasil	Kelebihan	Kekurangan
1	Syahrul (2017)	Mengembangkan sistem informasi manajemen proyek pada PT. Mozaik Bintang Persada karena sistem yang ada kurang informatif.	Rapid Application Development (RAD)	Unified Modeling Language (UML).	Sistem informasi manajemen proyek yang dapat memonitoring perkembangan proyek dan dapat berkoordinasi secara <i>real time</i>	<ol style="list-style-type: none"> 1. Terdapat notifikasi untuk aktivitas yang telah dikerjakan 2. Terdapat fitur komentar pada tugas untuk koordinasi 3. Klien dapat memantau langsung proses penggerjaan proyek. 	Tidak ada laporan proyek.
2	Nugroho (2012)	Pengembangan sistem informasi manajemen proyek konstruksi pada kampus UNDIP Tembalang. Sistem menggunakan bahasa pemrograman PHP dan <i>database server MySQL</i> .	Waterfall	Data Flow Diagram (DFD) dan Entity Relationship Diagram	Sistem yang dapat memberikan informasi kemajuan pekerjaan, pengarsipan dokumen proyek, <i>history</i> kegiatan.	<ol style="list-style-type: none"> 1. Terdapat pengolahan keuangan 2. Terdapat pembuatan laporan 3. Terdapat <i>database</i> untuk dokumen-dokumen pendukung 	<ol style="list-style-type: none"> 1. Tidak terdapat notifikasi pada sistem 2. Tidak terdapat pembagian tugas

						4. Terdapat forum diskusi untuk masalah dan solusi	
3	Firmansyah (2017)	Menganalisis perencanaan proyek kapal cepat rudal untuk mencegah keterlambatan proyek dengan menggunakan.	<i>what if analysis</i> pada model <i>Critical Path Method</i>	-	Dari hasil <i>Critical Path Method</i> diketahui lintasan kritis dari proyek. Dari analisis <i>what if</i> diketahui jumlah pekerja dan waktu serta biaya tambahan untuk mencegah keterlambatan proyek	Solusi dari <i>what if</i> sampai memperhitungkan biaya tambahan yang diperlukan untuk mencegah keterlambatan proyek	Analisis <i>what if</i> hanya dilakukan pada tiga aktivitas proyek.
4	Apriyanto (2018)	Merancang dan membangun sistem informasi manajemen proyek di PT. Yudha Perkasa Utama dikarenakan banyak hambatan dan kurangnya pengawasan yang mengakibatkan keterlambatan. Selain itu juga tidak adanya menggunakan	<i>Critical Path Method</i> (CPM) untuk penjadwalan, pengendalian waktu dan biaya	-	Sistem informasi manajemen proyek yang dapat membantu perencanaan penjadwalan, mengendalikan waktu dan biaya, manajemen risiko, dan komunikasi.	Menggunakan tiga metode untuk penjadwalan, pengendalian waktu dan biaya serta manajemen risiko.	Sistem tidak terdapat notifikasi

		manajemen risiko dan kurangnya komunikasi.	<i>Earned Value Management</i> (EVM), analisis risiko secara kualitatif menggunakan <i>Probability Impact Matrix</i> (PIM)				
5	Fatimah (2019)	Menganalisis keterlambatan waktu penyelesaian proyek pembangunan Jembatan Kaligawe menggunakan metode <i>what if analysis</i> dan <i>crash program</i> pada <i>critical path method</i> .	<i>what if analysis</i> dan <i>crash program</i> pada <i>critical path method</i> .	<i>Microsoft Project</i> 2010	Jalur kritis, hasil perhitungan percepatan durasi kegiatan, total jam-orang, penambahan jumlah pekerja, dan nilai <i>float</i> . Sehingga proyek dapat diselesaikan secara tepat waktu	Menggunakan dua metode yaitu <i>what if analysis</i> dan <i>crash program</i> untuk mengetahui jumlah waktu tambahan atau lembur dan biaya tambahan yang dibutuhkan agar proyek tetap selesai tepat waktu	Tidak memberikan rekomendasi dari hasil <i>what if analysis</i>

6	Abdurasyid <i>et al.</i> (2019)	Mengimplementasikan metode PERT dan CPM pada proyek pembangunan kapal dalam bentuk sistem informasi berbasis web.	PERT dan CPM	UML	Sistem mampu secara otomatis memetakan jalur kritis, mengidentifikasi awal dan akhir waktu setiap kegiatan untuk mencari jadwal proyek, dan menghitung jumlah waktu slack untuk setiap kegiatan sehingga dapat meminimalisasi terjadinya keterlambatan proyek dan dapat meminimalisasi biaya-biaya yang ditimbulkan akibat keterlambatan penggerjaan proyek	Menggunakan dua metode yaitu PERT dan CPM untuk mendapatkan jalur kritis dan waktu optimis penyelesaian proyek.	Sistem hanya untuk penjadwalan PERT dan CPM, tidak ada fitur manajemen lainnya.
---	---------------------------------	---	--------------	-----	---	---	---

7	Rompas (2020)	Mengembangkan sistem informasi manajemen proyek untuk kontraktor dan konsultan di Kota Manado.	Metode waterfall	UML	Perancangan sistem informasi manajemen proyek	Klien dapat mengakses sistem	Tidak ada fitur untuk penjadwalan proyek
8	Aprisa & Monalisa (2015)	Merancang dan membangun sistem informasi monitoring perkembangan proyek berbasis web pada PT. Inti Pratama Semesta karena penggerjaan proyek tidak dapat diketahui dengan cepat, sering terjadi kesalahan pencatatan data, tidak adanya gambar penggerjaan proyek.	Waterfall	UML	Sistem informasi monitoring perkembangan proyek berbasis web	1.Terdapat laporan proyek 2.Terdapat tindak turun tangan	Tidak ada fitur untuk penjadwalan proyek
9.	Arianie & Puspitasari (2017)	Merencanakan manajemen proyek untuk meningkatkan efisiensi dan efektivitas sumber daya perusahaan pada Qiscus	WBS, CPM, PERT, <i>crashing</i>	-	Estimasi percepatan penggerjaan proyek selama 6 hari, dengan penambahan biaya sebesar 2,8 juta.	Membandingkan hasil biaya dengan adanya <i>crashing</i> dan tidak.	Sistem hanya untuk penjadwalan PERT dan CPM,

		Pte Ltd dengan menyusun <i>work breakdown structure</i> , penjadwalan dengan <i>critical path method</i> dan <i>Program Evaluation and Review Technique</i> , Mengidentifikasi biaya proyek dan biaya <i>crashing</i> dari proyek Hayyan.					tidak ada fitur manajemen lainnya.
10	Agustina & Kusniant (2018)	Merancang dan membangun sistem informasi manajemen proyek PT. Xerindo Teknologi untuk memudahkan dan membantu pengguna dalam mengelola dan mengontrol setiap perencanaan kerja, proses kerja hingga laporan hasil dari suatu pekerjaan.	SDLC	DFD, ERD	Aplikasi sistem informasi manajemen proyek yang memudahkan dan membantu pengguna dalam mengelola dan mengontrol setiap perencanaan kerja, proses kerja hingga laporan hasil dari suatu pekerjaan.	Sistem memungkinkan untuk membuat laporan pekerjaan	Task-task yang disediakan statis dan tidak ada penjadwalan proyek.

Berdasarkan Tabel 2.2, penelitian sejenis yang dikumpulkan terkait dengan manajemen proyek, *critical path method* dan juga *what if analysis*. Diperoleh kelebihan dan kelemahan untuk dijadikan referensi dalam penelitian yang akan diteliti. Penelitian yang sangat relevan dan menjadi acuan dalam penelitian ini adalah penelitian oleh Syahrul (2017) dan Firmansyah (2017). Kelebihan penelitian ini dari penelitian sebelumnya adalah penggunaan metode *critical path method* dan *what if analysis* dalam rancang bangun sistem informasi manajemen proyek berbasis web.



BAB 3

METODOLOGI PENELITIAN

3.1 Metode Pengumpulan Data

Dalam penelitian dibutuhkan data-data yang dapat mendukung kebenaran materi uraian dan pembahasan. Pengumpulan data dilakukan dengan tiga metode yaitu studi pustaka, studi lapangan dan studi literatur sejenis.

3.1.1 Studi Pustaka

Studi Pustaka dilakukan dengan cara mempelajari dan membaca buku, jurnal, skripsi dan sumber lainnya yang berkaitan dengan sistem informasi manajemen proyek dan rancang bangun sistem informasi manajemen proyek. Dalam penelitian ini digunakan 5 skripsi dan 5 jurnal sebagai acuan perancangan sistem informasi manajemen proyek, dan kurang lebih 16 jurnal sebagai bahan referensi pembuatan skripsi ini. Dari hasil membaca berbagai sumber peneliti mempelajari bagaimana cara merancang dan membangun sistem informasi manajemen proyek.

3.1.2 Studi Lapangan

Dalam studi lapangan, dilakukan observasi dan wawancara.

a. Observasi

Pengamatan dilakukan dengan melakukan peninjauan secara langsung ke lokasi yang berkaitan yaitu PT. Triprima Karya. Hal yang diamati yaitu melihat secara

langsung bagaimana proses penyeleksian proyek. Observasi dilaksanakan di Kantor PT. Triprima Karya tepatnya Jl. Arjuna VI DF 3 No. 6, Villa Pamulang, Pondok Benda, Pamulang, Tangerang Selatan, Banten dan membutuhkan waktu satu bulan, dari 7 Februari sampai 6 Maret 2020. Dari hasil observasi didapatkan hasil yaitu:

- 1) PT. Triprima Karya belum menerapkan sistem informasi manajemen proyek sehingga sulit untuk memantau *progress* proyek yang banyak disaat bersamaan.
 - 2) Masih terdapat beberapa proyek yang mengalami keterlambatan.
- b. Wawancara

Wawancara dilakukan dengan pihak yang terkait dengan pengambil keputusan perusahaan yaitu direktur utama PT. Triprima Karya, Ibu Ir. Sarwo Sri. Wawancara dilakukan pada tanggal 2 Maret 2020 di Kantor PT. Triprima Karya. Hasil wawancara yang didapatkan adalah sebagai berikut:

- 1) Gambaran umum mengenai PT. Triprima Karya.
- 2) Struktur organisasi PT. Triprima Karya.
- 3) Proses bisnis yang sedang berjalan pada PT. Triprima Karya.
- 4) Permasalahan atau kelemahan-kelemahan dari sistem yang sedang berjalan pada PT. Triprima Karya.
- 5) Saran untuk sistem yang akan dirancang dan bangun.

3.2 Metode Analisis Sistem Berjalan

Peneliti menggunakan pendekatan untuk analisis permasalahan sistem, yaitu *Strength, Weakness, Opportunity, and Threat (SWOT) Analysis*. Pada Metode

SWOT diidentifikasi kekuatan, kelemahan, peluang dan ancaman melalui observasi dan wawancara dengan pihak PT. Triprima Karya.

3.3 Metode Pengembangan Sistem

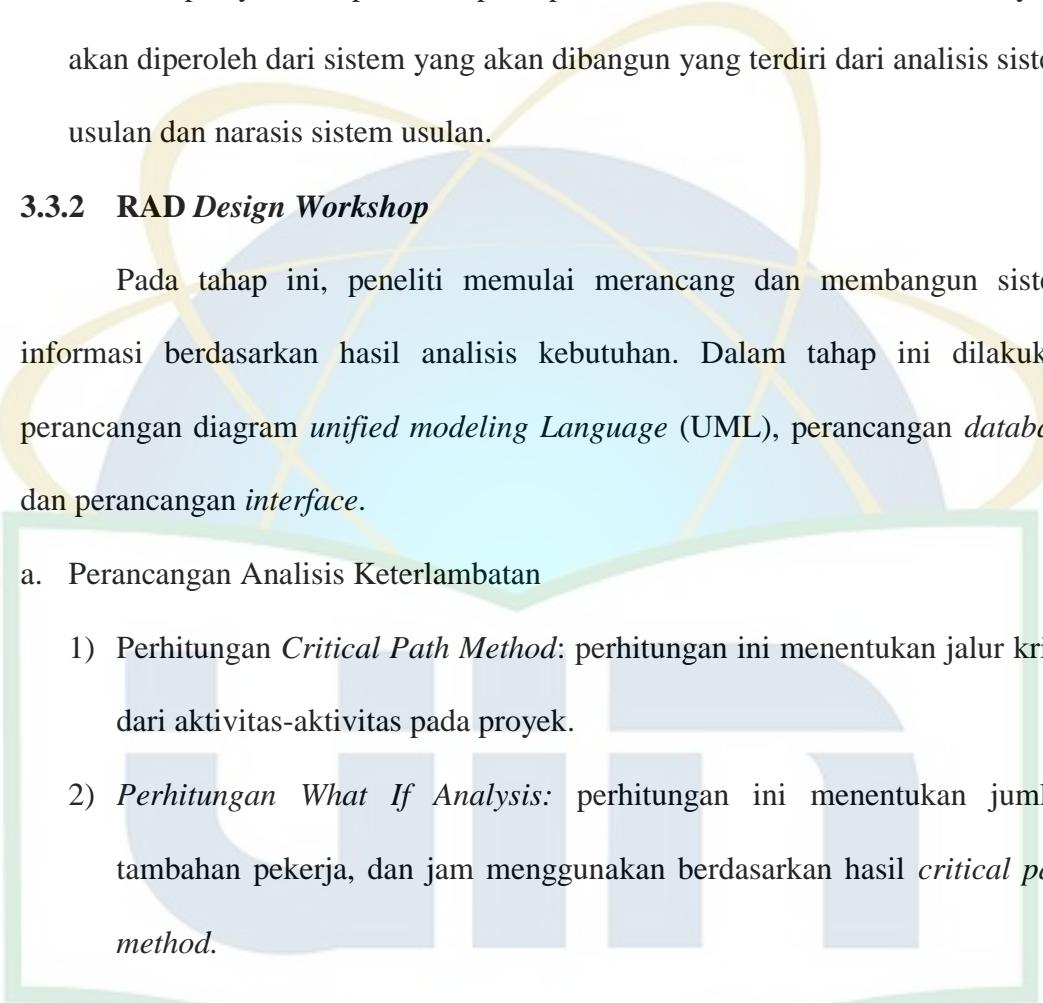
Metode pengembangan sistem yang digunakan adalah *Rapid Application Development* (RAD). Penggunaan metode RAD ini dikarenakan:

- a) Metode RAD dapat memenuhi kebutuhan karena pada tahap *requirements planning*, semua data yang dibutuhkan dikumpulkan.
- b) Metode RAD dapat menghemat waktu pembuatan sistem karena pada tahap *design workshop* dilakukan proses iterasi perancangan dan pembangunan sistem tanpa perlu mengulang dari awal.
- c) Metode RAD menggunakan *prototype* untuk memudahkan pengguna berinteraksi dengan sistem yang akan digunakan.

3.3.1 *Requirement Planning*

Pada tahap ini dilakukan pertemuan antara peneliti dan pihak PT. Triprima Karya untuk melakukan identifikasi tujuan dari sistem informasi manajemen proyek dan melakukan identifikasi kebutuhan informasi yang nantinya diterapkan kedalam sistem. Hasil dari pertemuan ini berupa:

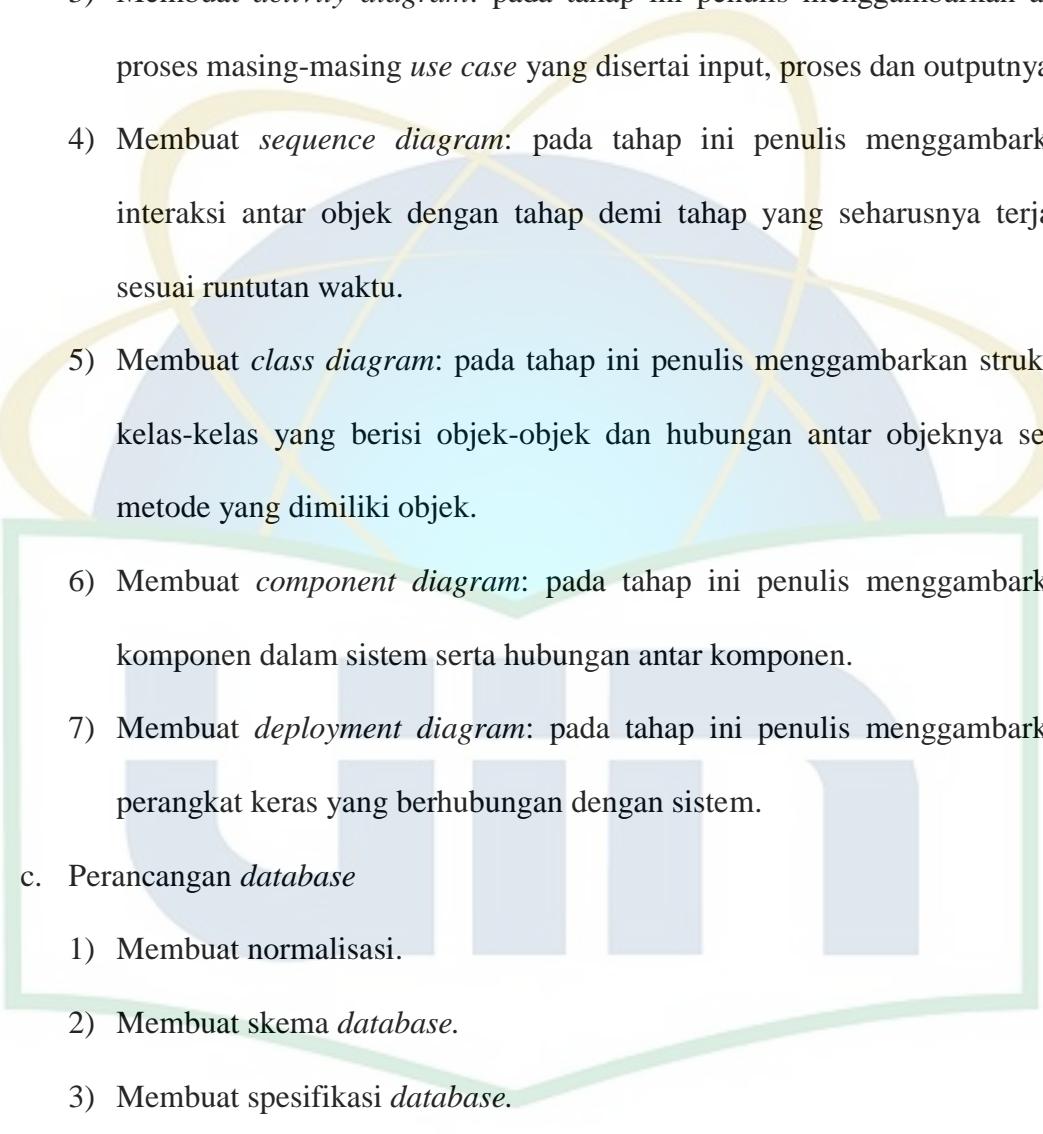
- a. Gambaran umum perusahaan: pada tahap ini penulis menjelaskan tentang PT. Triprima Karya yang terdiri dari gambaran umum, struktur organisasi, tugas dan tanggung jawab.

- 
- b. Analisis permasalahan: pada tahap ini penulis mengidentifikasi kekurangan dan masalah dari sistem yang sedang berjalan yang terdiri dari analisis sistem berjalan, narasi sistem berjalan dan analisis sistem SWOT.
 - c. Analisis persyaratan: pada tahap ini penulis memberikan rumusan hasil yang akan diperoleh dari sistem yang akan dibangun yang terdiri dari analisis sistem usulan dan narasis sistem usulan.

3.3.2 RAD Design Workshop

Pada tahap ini, peneliti memulai merancang dan membangun sistem informasi berdasarkan hasil analisis kebutuhan. Dalam tahap ini dilakukan perancangan diagram *unified modeling Language* (UML), perancangan *database* dan perancangan *interface*.

- a. Perancangan Analisis Keterlambatan
 - 1) Perhitungan *Critical Path Method*: perhitungan ini menentukan jalur kritis dari aktivitas-aktivitas pada proyek.
 - 2) *Perhitungan What If Analysis*: perhitungan ini menentukan jumlah tambahan pekerja, dan jam menggunakan berdasarkan hasil *critical path method*.
- b. Perancangan proses
 - 1) Membuat *use case diagram*: pada tahap ini penulis menggambarkan siapa saja aktor yang terlibat, apa saja aksi yang dapat dilakukan aktor dan interaksi antar aksi yang berhubungan.

- 
- 2) Membuat narasi *use case*: pada tahap ini penulis memberi penjelasan berbentuk tabel yang memberikan penjelasan lebih detail dari masing-masing *use case*.
 - 3) Membuat *activity diagram*: pada tahap ini penulis menggambarkan alur proses masing-masing *use case* yang disertai input, proses dan outputnya.
 - 4) Membuat *sequence diagram*: pada tahap ini penulis menggambarkan interaksi antar objek dengan tahap demi tahap yang seharusnya terjadi sesuai runtutan waktu.
 - 5) Membuat *class diagram*: pada tahap ini penulis menggambarkan struktur kelas-kelas yang berisi objek-objek dan hubungan antar objeknya serta metode yang dimiliki objek.
 - 6) Membuat *component diagram*: pada tahap ini penulis menggambarkan komponen dalam sistem serta hubungan antar komponen.
 - 7) Membuat *deployment diagram*: pada tahap ini penulis menggambarkan perangkat keras yang berhubungan dengan sistem.
- c. Perancangan *database*
- 1) Membuat normalisasi.
 - 2) Membuat skema *database*.
 - 3) Membuat spesifikasi *database*.
- d. Perancangan *interface*

Pada tahap ini, peneliti merancang tampilan yang akan dilihat oleh pengguna menggunakan Figma.

3.3.3 Implementasi

Pada tahap implementasi dilakukan pembangunan sistem berdasarkan rancangan yang telah dibuat menjadi sistem aplikasi yang akan digunakan. Dalam tahap ini dilakukan perhitungan analisis *what if*, *coding* atau pemrograman dan *testing* atau pengujian.

a. Pemrograman atau *coding*

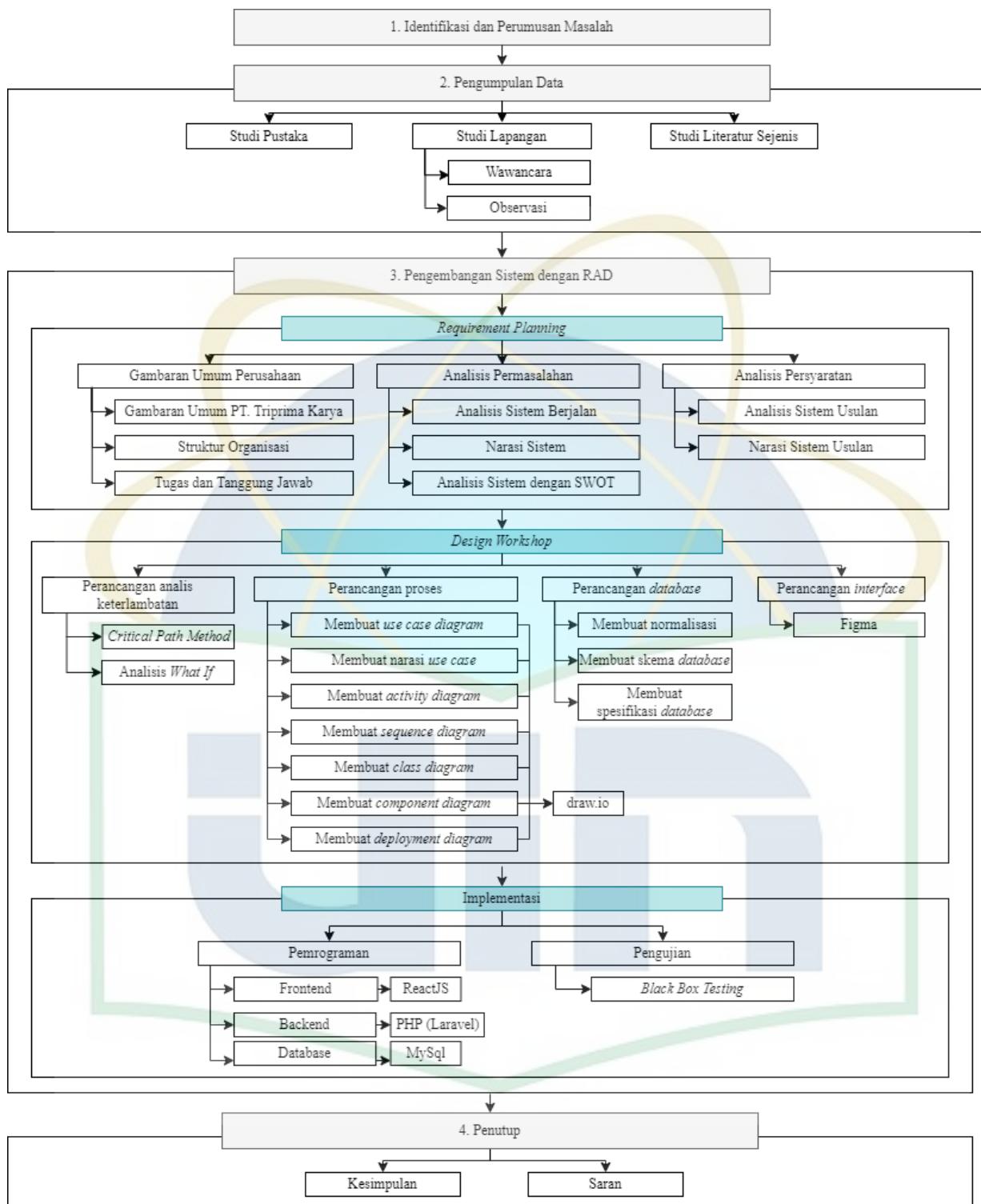
Pada pemrograman peneliti menggunakan PHP sebagai bahasa pemrograman dengan framework Laravel untuk *backend programming* dan ReactJS untuk *frontend programming*.

b. Pengujian atau *testing*

Setelah program aplikasi terbentuk, dilakukan pengujian. Peneliti menggunakan *black box testing*.

3.4. Tahapan Penelitian

Penyusunan penelitian rancang bangun sistem informasi manajemen proyek ini mempunyai beberapa tahap yang tergambar pada tahapan atau alur penelitian berikut:



Gambar 3.1 Tahapan Penelitian

BAB 4

PEMBAHASAN

4.1 Requirements Planning

Pada tahap *requirements planning*, peneliti mengumpulkan data-data yang dibutuhkan seperti gambaran umum perusahaan, analisis permasalahan, dan analisis persyaratan.

4.1.1 Gambaran Umum Perusahaan

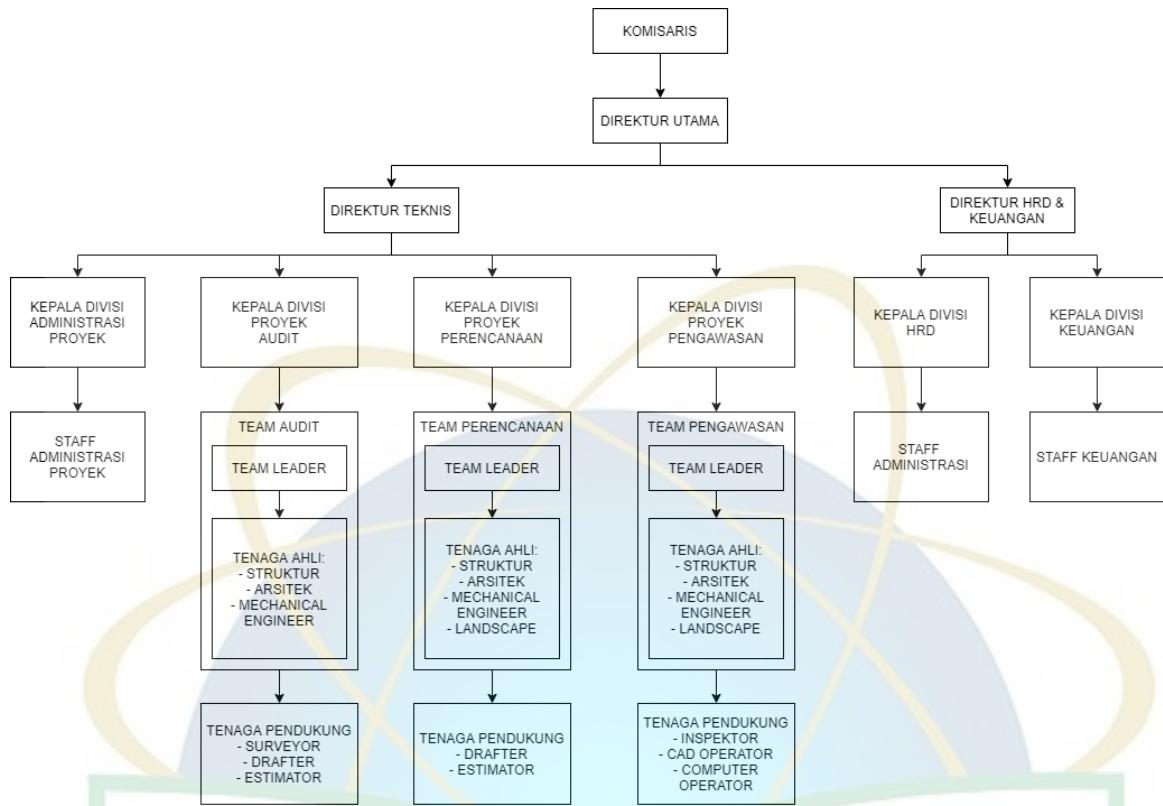
Gambaran umum perusahaan terdiri dari gambaran umum, struktur organisasi, tugas dan tanggung jawab.

4.1.1.1 Gambaran Umum PT. Triprima Karya

PT. Triprima Karya adalah usaha profesional yang didirikan di Jakarta pada 2003 yang bergerak dalam bidang jasa konsultan, dengan jangkauan layanan Bidang Perancangan Teknik, Pengawasan, Manajemen dan Perencanaan Pembangunan dari kalangan pemerintah maupun swasta yang secara umum bertujuan untuk mendukung dan berpartisipasi dalam gerak langkah pembangunan Bangsa dan Negara Indonesia (*Company Profile* PT. Triprima Karya, 2020).

4.1.1.2 Struktur Organisasi

Berikut struktur organisasi dari perusahaan PT. Triprima Karya.



Gambar 4.1 Struktur Organisasi PT. Triprima Karya
(Data Primer PT. Triprima Karya, 2020)

4.1.1.3 Tugas dan Tanggung Jawab

Adapun tugas dan tanggung jawab bagian-bagian pada perusahaan. (Data Primer PT. Triprima Karya, 2020)

a. Komisaris:

- 1) Melakukan pengawasan dengan itikad baik dan kehati-hatian untuk kepentingan perseroan dan sesuai dengan maksud dan tujuan perseroan.
- 2) Tidak mempunyai kepentingan pribadi baik langsung maupun tidak langsung atas tindakan pengurusan direksi yang mengakibatkan kerugian.
- 3) Telah memberikan nasihat kepada direksi untuk mencegah timbul atau berlanjutnya kerugian tersebut.

b. Direktur Utama:

- 1) Direksi wajib bertanggung jawab penuh atas pengurusan perseroan untuk kepentingan dan tujuan perseroan serta mewakili perseroan baik di dalam maupun di luar pengadilan. Sebagai organ yang wajib bertanggung jawab, direksi mempertanggungjawabkan kepengurusan itu kepada RUPS.
- 2) Direksi wajib menyiapkan laporan tahunan (termasuk pertanggungjawaban tahunan) untuk RUPS.
- 3) Memimpin dan bertanggung jawab menjalankan perusahaan.
- 4) Bertanggung jawab terhadap kerugian yang mungkin dihadapi perusahaan, pun bertanggung jawab terhadap keuntungan perusahaan.
- 5) Menentukan, merumuskan, dan memutuskan sebuah kebijakan dalam perusahaan.
- 6) Menyusun dan menetapkan berbagai strategi strategis untuk mencapai visi dan misi perusahaan.
- 7) Mengkoordinasikan dan mengawasi semua kegiatan di perusahaan, mulai bidang administrasi, kepegawaian hingga pengadaan barang.
- 8) Menjadi perwakilan perusahaan dalam hubungannya dengan dunia luar perusahaan.
- 9) Pengesahan segala dokumen administrasi pekerjaan.
- 10) Mengecek, mengawasi dan menentukan semua kebutuhan dalam proses operasional perusahaan.

11) Merencanakan, menentukan, mengawasi, mengambil keputusan serta melakukan koordinasi dalam hal keuangan untuk kebutuhan operasional perusahaan.

c. Direktur Teknis:

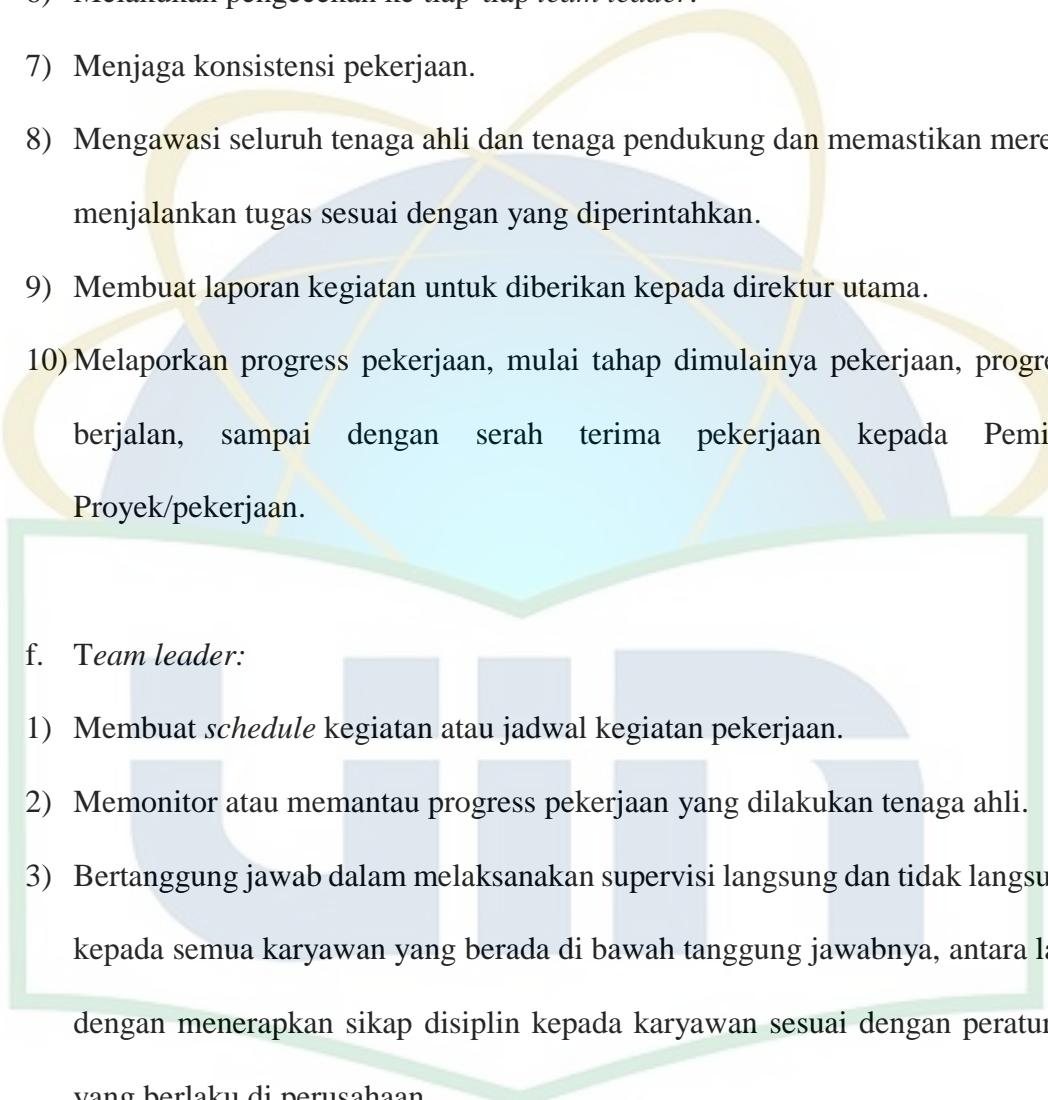
- 1) Penanggung jawab utama terhadap semua kegiatan proyek.
- 2) Pengesahan segala dokumen teknis pekerjaan.
- 3) Penunjukan tenaga ahli dan tenaga pendukung, serta memberikan mandat dan fungsi tugasnya sesuai kapasitas dan kebutuhan pada suatu proyek atau pekerjaan.
- 4) Pengambil keputusan tertinggi dalam progress pekerjaan. Dalam kapasitas dan fungsi tertentu, segala bentuk pelaksanaan pekerjaan harus diketahui dan mendapatkan persetujuan dari direktur utama.
- 5) Membantu tugas-tugas direktur utama.
- 6) Bertanggung jawab terhadap seluruh proses operasional, pelayanan jasa, proyek hingga kualitas hasil produk jasa.
- 7) Bertanggung jawab terhadap pengembangan kualitas produk maupun karyawan yang terlibat.
- 8) Menyusun strategi dalam pemenuhan target perusahaan, dan cara mencapai target tersebut.
- 9) Merencanakan, menentukan, mengawasi, mengambil keputusan serta melakukan koordinasi dalam hal keuangan untuk kebutuhan operasional proyek.

d. Direktur HRD & Keuangan:

- 1) Melakukan seleksi penerimaan tenaga kerja.
- 2) Penilaian kinerja karyawan.
- 3) Memberikan penghargaan.
- 4) Mengangkat dan memberhentikan karyawan perusahaan.
- 5) *Monitoring*, mengukur dan melakukan pelaporan mengenai masalah, strategi dalam mengembangkan SDM dan pencapaiannya sesuai kesepakatan.
- 6) Bertugas dalam pengembangan dan mengatur staf.
- 7) Bertugas mengevaluasi dan memberi penilaian terhadap kinerja para karyawan yang bekerja sama dengan tim eksekutif.
- 8) Memastikan bahwa setiap aktivitas memiliki inti dan tujuan serta terintegrasi dengan persyaratan perusahaan untuk manajemen keselamatan kerja, mutu, kesehatan, hukum, dan kebijakan.
- 9) Bertanggung jawab terhadap kinerja keuangan sebuah perusahaan.
- 10) Bertanggung jawab membuat laporan keuangan perusahaan.
- 11) Mengawasi laporan keuangan perusahaan.
- 12) Meminimalisir resiko keuangan yang mungkin merugikan perusahaan.
- 13) Melihat secara jeli peluang perusahaan.

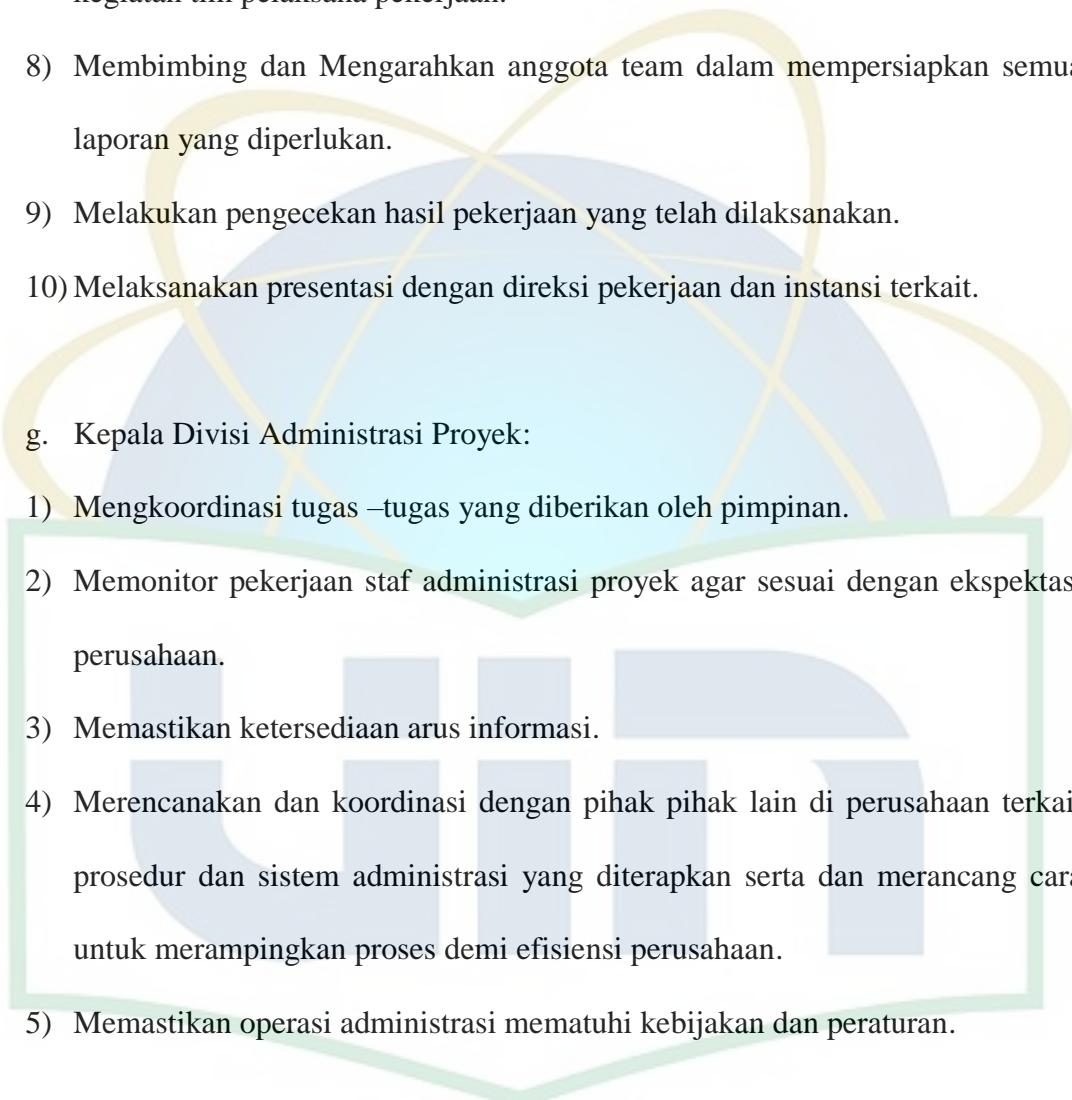
e. Kepala Divisi Proyek:

- 1) Mengawasi pengelolaan pelaksanaan proyek.
- 2) Mengelola pelaksanaan kegiatan perencanaan pelelangan.

- 
- 3) Melakukan koordinasi kerja harian.
 - 4) Melaksanakan jadwal dan kegiatan pelelangan.
 - 5) Mengecek status kegiatan pelaksanaan proyek.
 - 6) Melakukan pengecekan ke tiap-tiap *team leader*.
 - 7) Menjaga konsistensi pekerjaan.
 - 8) Mengawasi seluruh tenaga ahli dan tenaga pendukung dan memastikan mereka menjalankan tugas sesuai dengan yang diperintahkan.
 - 9) Membuat laporan kegiatan untuk diberikan kepada direktur utama.
 - 10) Melaporkan progress pekerjaan, mulai tahap dimulainya pekerjaan, progress berjalan, sampai dengan serah terima pekerjaan kepada Pemilik Proyek/pekerjaan.

f. *Team leader:*

- 1) Membuat *schedule* kegiatan atau jadwal kegiatan pekerjaan.
- 2) Memonitor atau memantau progress pekerjaan yang dilakukan tenaga ahli.
- 3) Bertanggung jawab dalam melaksanakan supervisi langsung dan tidak langsung kepada semua karyawan yang berada di bawah tanggung jawabnya, antara lain dengan menerapkan sikap disiplin kepada karyawan sesuai dengan peraturan yang berlaku di perusahaan.
- 4) Bertanggung jawab dalam melaksanakan koordinasi dalam membina kerja sama team yang solid.
- 5) Bertanggung jawab dalam mencapai suatu target pekerjaan yang telah ditetapkan dan sesuai dengan aturan.

- 
- 6) Mengkoordinasi seluruh aktivitas Tim dalam mengelola seluruh kegiatan baik di lapangan maupun di kantor.
 - 7) Bertanggung jawab terhadap Pemberi Pekerjaan yang berkaitan terhadap kegiatan tim pelaksana pekerjaan.
 - 8) Membimbing dan Mengarahkan anggota team dalam mempersiapkan semua laporan yang diperlukan.
 - 9) Melakukan pengecekan hasil pekerjaan yang telah dilaksanakan.
 - 10) Melaksanakan presentasi dengan direksi pekerjaan dan instansi terkait.

g. Kepala Divisi Administrasi Proyek:

- 1) Mengkoordinasi tugas –tugas yang diberikan oleh pimpinan.
- 2) Memonitor pekerjaan staf administrasi proyek agar sesuai dengan ekspektasi perusahaan.
- 3) Memastikan ketersediaan arus informasi.
- 4) Merencanakan dan koordinasi dengan pihak pihak lain di perusahaan terkait prosedur dan sistem administrasi yang diterapkan serta merancang cara untuk merampingkan proses demi efisiensi perusahaan.
- 5) Memastikan operasi administrasi mematuhi kebijakan dan peraturan.

h. Tenaga Administrasi Proyek:

- 1) Membuat yaitu Surat Penawaran Harga (SPH) yang terdiri dari dokumen administrasi, dokumen teknis, dan dokumen biaya.

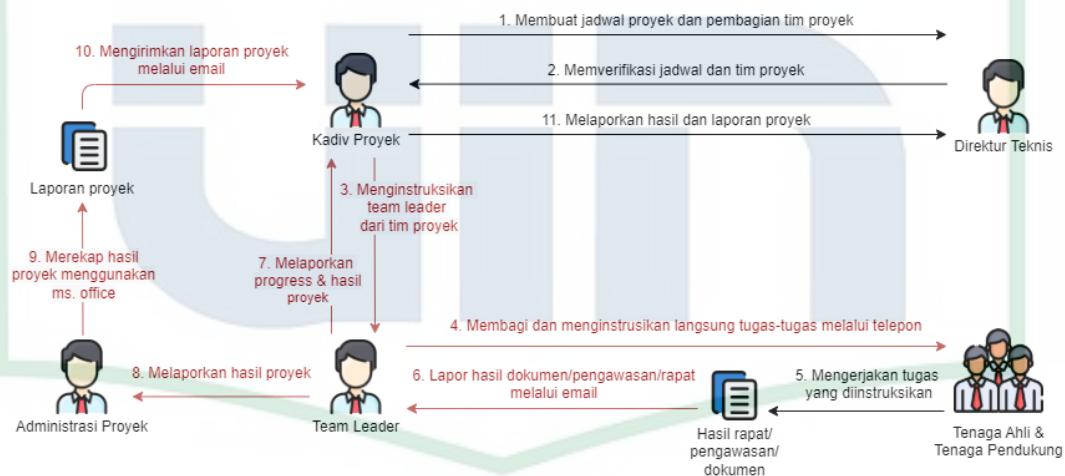
- 2) Mengarsip kontrak proyek, akta, NPWP, Surat Keterangan Terdaftar (SKT), Pengusaha Kena Pajak (PKP), Nomor Induk Berusaha (NIB), Surat Ijin Usaha Perdagangan (SIUP), IUJK, Kartu Tanda Anggota (KTA) Asosiasi, SBU.
- 3) Membuat laporan berkala (mingguan, bulanan atau periode tertentu).

4.1.2 Analisis Permasalahan

Pada analisis permasalahan, dilakukan analisis pada sistem berjalan dengan menggunakan metode SWOT untuk memperkuat analisis pada sistem berjalan.

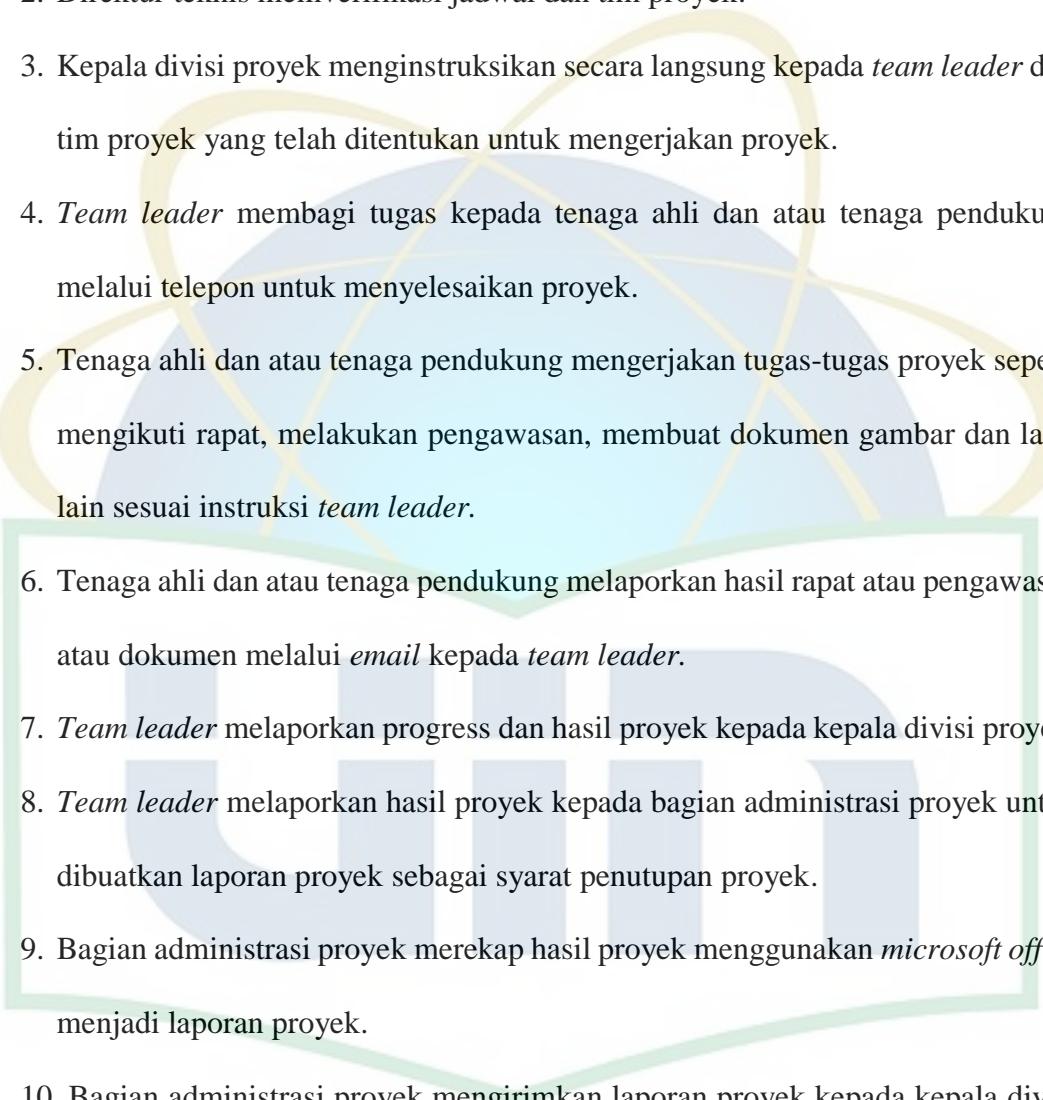
4.1.2.1 Identifikasi Sistem Berjalan

Peneliti mengidentifikasi sistem manajemen proyek yang saat ini sedang berjalan yang ditunjukkan pada Gambar 4.2.



Gambar 4.2 Rich Picture Sistem Berjalan

Berdasarkan Gambar 4.2, berikut tahapan atau penjelasan sistem berjalan:

- 
1. Kepala divisi proyek membuat jadwal dan pembagian tim proyek dan diserahkan kepada direktur teknis. Dalam pembagian tim, satu karyawan dapat terlibat dalam satu sampai tiga proyek.
 2. Direktur teknis memverifikasi jadwal dan tim proyek.
 3. Kepala divisi proyek menginstruksikan secara langsung kepada *team leader* dari tim proyek yang telah ditentukan untuk mengerjakan proyek.
 4. *Team leader* membagi tugas kepada tenaga ahli dan atau tenaga pendukung melalui telepon untuk menyelesaikan proyek.
 5. Tenaga ahli dan atau tenaga pendukung mengerjakan tugas-tugas proyek seperti mengikuti rapat, melakukan pengawasan, membuat dokumen gambar dan lain-lain sesuai instruksi *team leader*.
 6. Tenaga ahli dan atau tenaga pendukung melaporkan hasil rapat atau pengawasan atau dokumen melalui *email* kepada *team leader*.
 7. *Team leader* melaporkan progress dan hasil proyek kepada kepala divisi proyek.
 8. *Team leader* melaporkan hasil proyek kepada bagian administrasi proyek untuk dibuatkan laporan proyek sebagai syarat penutupan proyek.
 9. Bagian administrasi proyek merekap hasil proyek menggunakan *microsoft office* menjadi laporan proyek.
 10. Bagian administrasi proyek mengirimkan laporan proyek kepada kepala divisi proyek menggunakan *email*.
 11. Kepala divisi proyek memberikan laporan proyek kepada direktur teknis.

4.1.2.2 Analisis Sistem dengan SWOT

Dalam melakukan analisis sistem yang sedang berjalan, peneliti menggunakan metode yaitu SWOT untuk mengetahui kekurangan maupun kelebihan dari sistem. Berikut hasil analisis dari metode SWOT dimana ditentukan faktor kekuatan, kelemahan, peluang dan ancaman pada sistem yang berjalan:

Kekuatan

- a. Sumber daya manusia cukup terbiasa menggunakan teknologi.
- b. Fasilitas lain seperti *printer* dan internet memadai.

Kelemahan

- a. Direksi sulit untuk memantau progress banyak proyek.
- b. Tidak dapat menganalisis proyek akan terlambat atau tepat waktu.
- c. Tidak ada pengingat waktu tenggat.
- d. Alur komunikasi cukup panjang yang memerlukan banyak waktu.

Peluang

- a. Perkembangan teknologi yang semakin maju, dimana proses transaksi sekarang ini banyak dilakukan dengan menggunakan sistem dan beralih dari proses manual.
- b. Pengguna online semakin banyak, tak dipungkiri manusia kini lebih memilih menggunakan teknologi informasi yang canggih dibandingkan dengan cara manual karena terdapat banyak kelebihan yang dimiliki.
- c. Kemudahan perolehan data secara *online*, teknologi informasi sekarang ini mempermudah manusia memperoleh informasi secara *online*.

Ancaman

- a. Kemungkinan *server down*.
- b. Kemungkinan dilakukan peretasan.

Tabel 4.1 Matriks SWOT Analisis Sistem Berjalan

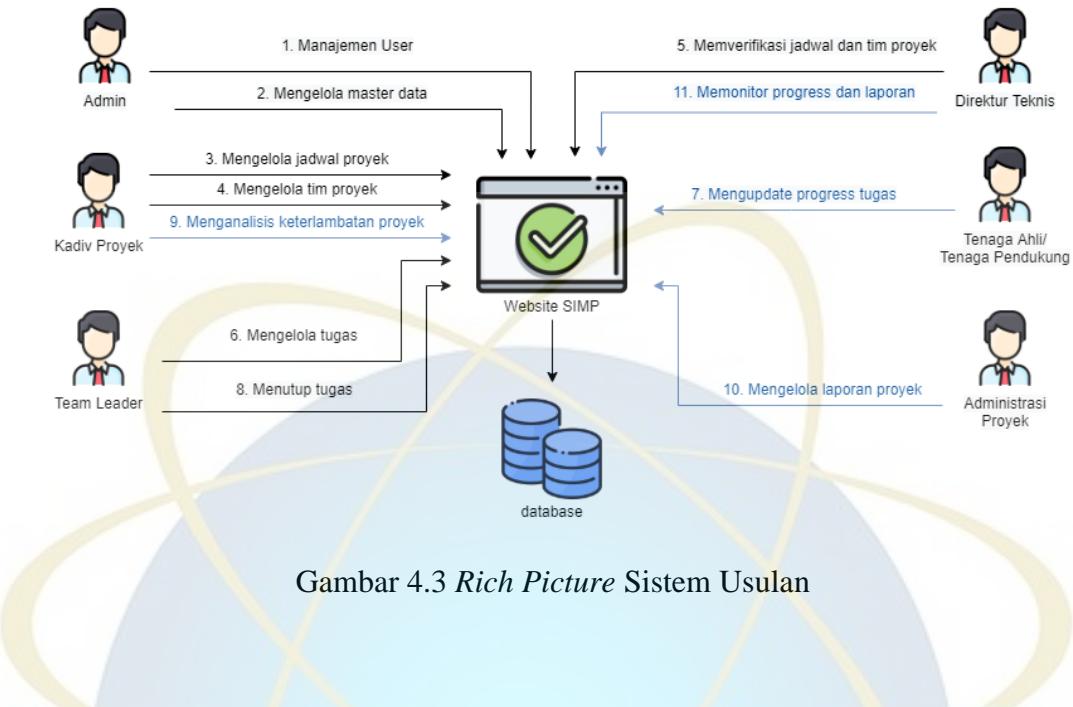
	<p>Kekuatan</p> <ol style="list-style-type: none"> 1. Sumber daya manusia cukup terbiasa menggunakan teknologi. 2. Fasilitas lain seperti <i>printer</i> dan internet memadai. 	<p>Kelemahan</p> <ol style="list-style-type: none"> 1. Direksi sulit untuk memantau progress banyak proyek. 2. Tidak dapat menganalisis proyek akan terlambat atau tepat waktu. 3. Tidak ada pengingat waktu tenggat. 4. Alur komunikasi cukup panjang yang memerlukan banyak waktu.
<p>Peluang</p> <ol style="list-style-type: none"> 1. Perkembangan teknologi yang semakin maju, dimana proses transaksi sekarang ini banyak dilakukan dengan menggunakan sistem dan beralih dari proses manual. 2. Pengguna online semakin banyak, tak 	<p>STRATEGI SO</p> <p>Merancang dan membangun sistem manajemen proyek berbasis website yang yang dapat digunakan untuk laporan secara <i>real time</i> dari lapangan.</p>	<p>STRATEGI WO</p> <p>Merancang dan membangun sistem manajemen proyek yang dapat menampilkan progress pada <i>dashboard</i>, memberikan notifikasi, dan menganalisis keterlambatan proyek menggunakan metode <i>what if analysis</i>.</p>



<p>dipungkiri manusia kini lebih memilih menggunakan teknologi informasi yang canggih dibandingkan dengan cara manual karena terdapat banyak kelebihan yang dimiliki.</p> <p>3. Kemudahan perolehan data secara online, teknologi informasi sekarang ini mempermudah manusia memperoleh informasi secara <i>online</i>.</p>		
<p>Ancaman</p> <ol style="list-style-type: none"> 1. Kemungkinan <i>server down</i>. 2. Kemungkinan dilakukan peretasan. 	<p>STRATEGI ST Memanfaatkan sumber daya yang ada untuk memaksimalkan penggunaan web.</p>	<p>STRATEGI WT Membuat sistem otomatis yang mempunyai hak akses.</p>

4.1.3 Analisis Sistem Usulan

Berdasarkan hasil analisis sistem berjalan didapatkan kekuatan, kelemahan, peluang dan ancaman serta strategi untuk menjadi acuan sistem usulan yang ditunjukkan pada Gambar 4.3.



Gambar 4.3 *Rich Picture* Sistem Usulan

Berikut tahapan atau penjelasan dari sistem usulan pada Gambar 4.3:

- Admin melakukan manajemen user untuk mengatur hak akses pengguna pada sistem.
- Admin mengelola master data seperti data proyek, karyawan, klien.
- Kepala divisi proyek mengelola jadwal proyek
- Kepala divisi proyek mengelola tim proyek yang terdiri dari *team leader*, tenaga ahli dan atau tenaga pendukung dimana satu karyawan dapat terlibat dalam satu sampai tiga proyek.
- Direktur teknis memverifikasi jadwal dan tim proyek yang diajukan oleh kepala divisi proyek dan sistem akan memberikan notifikasi kepada *team leader*, tenaga ahli dan pendukung untuk memulai proyek.
- Team leader* mengelola tugas dan sistem akan memberikan notifikasi kepada tenaga ahli dan atau tenaga pendukung yang diberikan tugas oleh *team leader*.

- g. Tenaga ahli dan atau tenaga pendukung *update progress* tugas yang sedang dikerjakan.
- h. *Team leader* menutup tugas yang telah diselesaikan oleh tenaga ahli atau pendukung.
- i. Kepala divisi proyek menganalisis keterlambatan proyek yang secara otomatis generate jadwal dengan *critical path method* dan metode *what if analysis*.
- j. Administrasi proyek mengelola laporan proyek untuk penutupan proyek berdasarkan tugas yang telah di-update oleh tenaga ahli dan atau tenaga pendukung.
- k. Direktur teknis memonitor *progress* proyek-proyek melalui *dashboard web* dan menerima laporan proyek-proyek yang telah dibuat oleh administrasi proyek.

4.2 Design Workshop

4.2.1 Perancangan Analisis Keterlambatan

Untuk mengantisipasi keterlambatan proyek, jadwal yang sudah dibuat dapat ditentukan jalur kritisnya menggunakan metode *Critical Path Method*. Dari jalur kritis tersebut dapat dilakukan perhitungan untuk menentukan jumlah jam dan pekerja tambahan jika aktivitas proyek terlambat menggunakan metode *what if analysis*.

4.2.1.1 Perhitungan *Critical Path Method*

Tahapan pertama dalam perhitungan menggunakan *critical path method* adalah pengidentifikasi kegiatan dan durasinya serta ketergantungan antar kegiatan yang akan menjadi *successor*.

Tabel 4.2 Identifikasi Aktivitas

Kode	Kegiatan	Durasi (hari)	Pendahulu	Successor
A	Pekerjaan pendahuluan	14	-	B
B	Pekerjaan persiapan	14	A	C,D
C	Pekerjaan konstruksi	70	B	E
D	Perlengkapan penunjang	63	B	-
E	Pekerjaan mekanikal dan <i>plumbing</i>	63	C	F
F	Pekerjaan elektrikal	56	E	H
G	Pekerjaan penghijauan	28	-	-
H	Penyambungan daya listrik baru	21	F	-

Setelah menentukan kegiatan serta hubungan dan ketergantungan antar aktivitas, dapat dilakukan perhitungan maju dan mundur. Dalam perhitungan maju didapatkan *earliest start* (ES) dan *earliest finish* (EF). Sesuai persamaan (2.1), maka didapatkan:

$$EF(A) = 0 + 14 \Rightarrow 14$$

$$EF(B) = 14 + 14 \Rightarrow 28$$

$$EF(C) = 28 + 70 \Rightarrow 98$$

$$EF(D) = 28 + 63 \Rightarrow 91$$

$$EF(E) = 98 + 63 \Rightarrow 161$$

$$EF(F) = 161 + 56 \Rightarrow 217$$

$$EF(G) = 0 + 28 \Rightarrow 28$$

$$EF(H) = 217 + 21 \Rightarrow 238, EF(F) > EF(D) > EF(G)$$

Pada perhitungan mundur didapatkan *latest start* (LS) dan *latest finish* (LF) menggunakan persamaan (2.2). Berikut hasil yang didapatkan:

$$LS(H) = 238 - 21 \Rightarrow 217$$

$$LS(G) = 238 - 28 \Rightarrow 210$$

$$LS(F) = 217 - 56 \Rightarrow 161$$

$$LS(E) = 161 - 63 \Rightarrow 98$$

$$LS(D) = 238 - 63 \Rightarrow 175$$

$$LS(C) = 98 - 70 \Rightarrow 28$$

$$LS(B) = 28 - 14 \Rightarrow 14, LS(C) < LS(D)$$

$$LS(A) = 14 - 14 \Rightarrow 0$$

Tabel 4.3 Hasil Perhitungan Maju dan Mundur

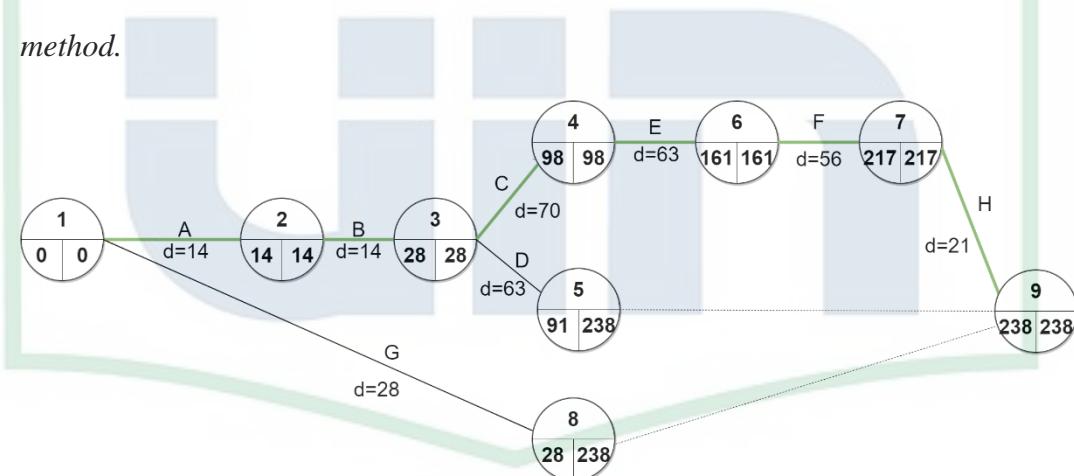
Kode	Durasi	Pendahulu	ES	EF	LS	LF
A	14	-	0	14	0	14
B	14	A	14	28	14	28
C	70	B	28	98	28	98
D	63	B	28	91	175	238
E	63	C	98	161	98	161
F	56	E	161	217	161	217
G	28	-	0	28	210	238
H	21	F	217	238	217	238

Untuk menentukan jalur kritis, perlu dilakukan perhitungan *float* menggunakan persamaan (2.3).

Tabel 4.4 Hasil Perhitungan *Critical Path Method*

Kode	Durasi	ES	EF	LS	LF	Float
A	14	0	14	0	14	0
B	14	14	28	14	28	0
C	70	28	98	28	98	0
D	63	28	91	175	238	147
E	63	98	161	98	161	0
F	56	161	217	161	217	0
G	28	0	28	210	238	210
H	21	217	238	217	238	0

Gambar 4.4 menggambarkan grafik dari hasil perhitungan *critical path method*.



Gambar 4.4 Grafik Hasil *Critical Path Method*

Aktivitas yang mempunyai nilai *float* 0 merupakan kegiatan yang kritis.

Maka jalur kritis yaitu pada aktivitas Pekerjaan pendahuluan (A), Pekerjaan

persiapan (B), Pekerjaan konstruksi (C), Pekerjaan mekanikal dan *plumbing* (E), Pekerjaan elektrikal (F), Penyambungan daya listrik baru (H).

4.2.1.2 Perhitungan *What If Analysis*

Pekerjaan pendahuluan (A), Pekerjaan persiapan (B), Pekerjaan konstruksi (C), Pekerjaan mekanikal dan *plumbing* (E), Pekerjaan elektrikal (F), Penyambungan daya listrik baru (H) tidak mempunyai waktu tambahan sehingga tidak boleh mengalami keterlambatan karena akan mempengaruhi waktu selesain proyek. Namun, tidak menutup kemungkinan keterlambatan pada aktivitas-aktivitas tersebut dapat terjadi. Untuk mengantisipasi keterlambatan penyelesaian proyek, aktivitas dapat dilakukan percepatan dengan menggunakan *what if analysis* untuk menghitung jumlah pekerja dan waktu tambahan yang dibutuhkan. Berikut contoh perhitungan *what if analysis*, jika aktivitas Pekerjaan pendahuluan (A) mengalami keterlambatan selama 7 hari.

Aktivitas = Pekerjaan pendahuluan (A)

Durasi rencana atau $d = 14$ hari

$Float = 0$

Delay aktivitas atau *delay* menggunakan persamaan (2.4):

$$delay = \frac{7}{14} \times 100\% \times 14 = 7 \text{ hari}$$

Delay proyek atau *delay'p* menggunakan persamaan (2.5):

$$delay'p = 7 - 0 = 7 \text{ hari}$$

Jika $delay'p > 0$, maka proyek akan terlambat akibat aktivitas Pekerjaan pendahuluan (A). Karena $delay'p$ bernilai 7 yaitu lebih besar dari 0, maka proyek akan terlambat. Selanjutnya periksa aktivitas yang menjadi *successor* dari aktivitas Pekerjaan pendahuluan (A). Kegiatan yang akan berpengaruh atau *successor* dari aktivitas Pekerjaan pendahuluan (A) adalah Pekerjaan persiapan (B), Pekerjaan konstruksi (C), Pekerjaan mekanikal dan *plumbing* (E), Pekerjaan elektrikal (F), Penyambungan daya listrik baru (H), Perlengkapan penunjang (D). Setelah mendapatkan aktivitas *successor*, dilakukan perhitungan percepatan pada masing-masing aktivitas. Berikut contoh perhitungan aktivitas *successor* Pekerjaan persiapan (B):

Aktivitas = Pekerjaan persiapan (B)

Durasi rencana atau $ds = 14$ hari

$Float = 0$

Jumlah pekerja normal atau $n = 5$ orang

Jam kerja normal per hari atau $H = 8$ jam/hari

$Manhour$ atau $mh = n \times H \times d = 5 \times 8 \times 14 = 560$

Percepatan atau $d's$ menggunakan persamaan (2.6):

$$d's = 14 + 0 - 7 = 7 \text{ hari}$$

Memeriksa aktivitas B dapat dilakukan percepatan dengan batasan:

1. $d's < ds$, percepatan yang dilakukan tidak boleh lebih besar dari durasi awal kegiatan
2. $1/2ds \geq delay$, keterlambatan harus 2 kali lebih kecil daripada durasi

Kondisi percepatan aktivitas B:

1. $7 < 14$, memenuhi
2. $\frac{1}{2} \times 14 = 7$, memenuhi

Karena dua batasan terpenuhi maka percepatan pada Pekerjaan persiapan (B) dapat dilakukan.

Penambahan jumlah pekerja menggunakan persamaan (2.9):

$$\Delta n = \frac{560}{7 \times 8} - 5 = 5 \text{ orang}$$

Penambahan jumlah jam menggunakan persamaan (2.10):

$$\Delta H = \frac{560}{7 \times 5} - 8 = 8 \text{ jam}$$

Aktivitas Pekerjaan persiapan (B) dapat melakukan penambahan jumlah pekerja sebanyak 5 orang dan jam per hari sebanyak 8 jam.

Selanjutnya lakukan perhitungan percepatan pada aktivitas *successor* lainnya yaitu Pekerjaan konstruksi (C), Pekerjaan mekanikal dan *plumbing* (E), Pekerjaan elektrikal (F), Penyambungan daya listrik baru (H), Perlengkapan penunjang (D).

Tabel 4.5 Hasil Perhitungan *What If Analysis*

Kode	<i>ds</i>	<i>float</i>	<i>n</i>	<i>H</i>	<i>mh</i>	<i>delay'p</i>	<i>d's</i>	<i>d's < ds</i>	<i>1/2ds >= delay</i>	Δn	ΔH
B	14	0	5	8	560	7	7	true	true	5	8
C	70	0	10	8	5600	7	63	true	true	1	0,9
E	63	0	8	8	4032	7	56	true	true	1	1
F	56	0	7	8	3136	7	49	true	true	1	1,1
H	21	0	5	8	840	7	14	true	true	3	4
D	63	147	3	8	1512	7	203	false	-	-	-

4.2.2 Perancangan UML

Berdasarkan hasil analisis kebutuhan, selanjutnya dilakukan perancangan diagram *unified modeling Language* (UML), perancangan *database* dan perancangan *interface*.

4.2.2.1 Use Case Diagram

Pada *use case diagram*, digambarkan aktor-aktor yang terlibat dan interaksinya dengan sistem. Sebelum membuat diagram perlu untuk mengidentifikasi aktor dan aktivitas yang ada dalam sistem.

Tabel 4.6 Identifikasi Aktor

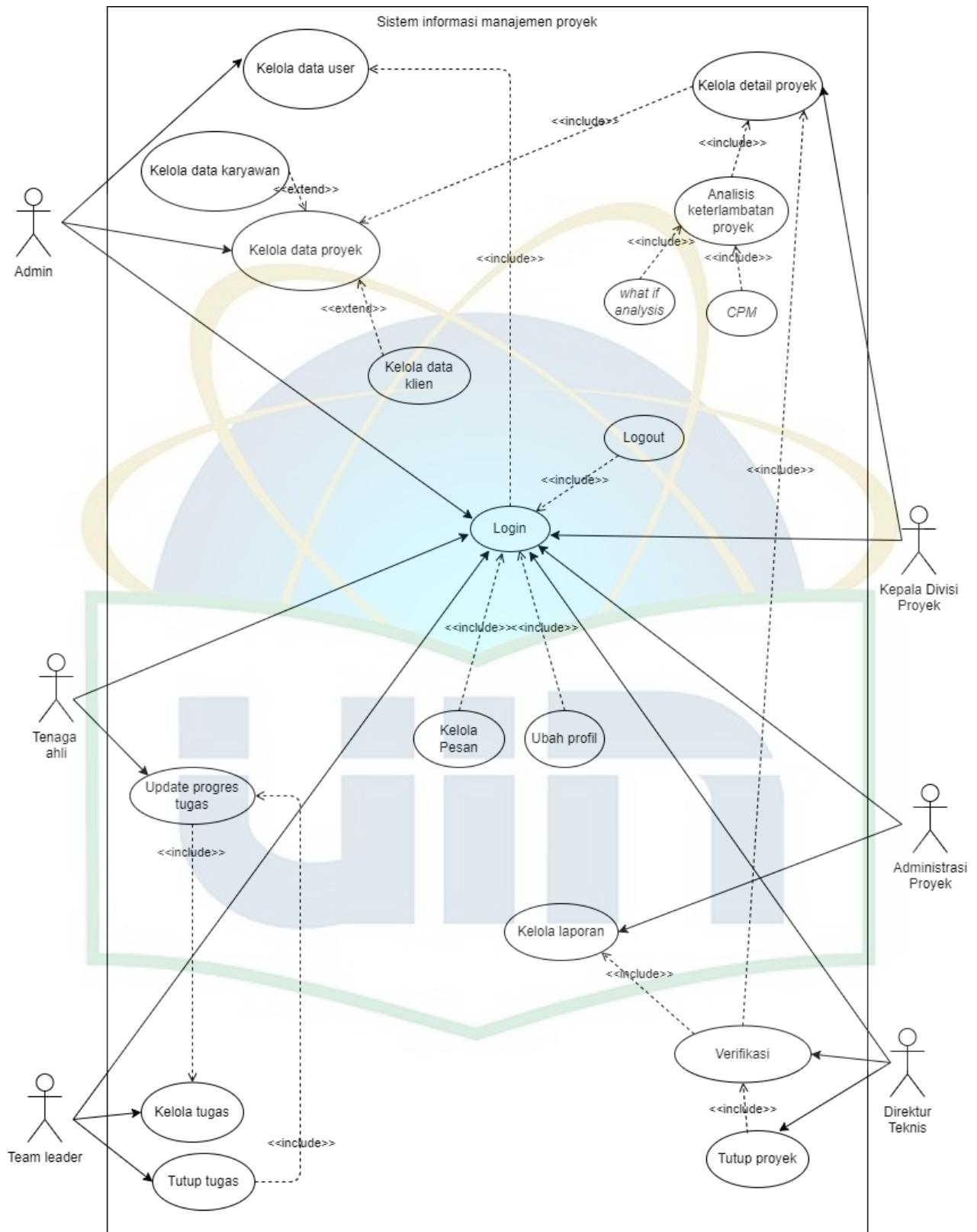
No	Aktor	Deskripsi
1	Admin	Aktor yang melakukan manajemen user, mengelola master data dan bertanggung jawab penuh atas seluruh kegiatan di sistem.
2	Direktur Teknis	Aktor yang memverifikasi jadwal dan tim proyek.
3	Kepala Divisi Proyek	Aktor yang membuat jadwal dan tim proyek serta menganalisis keterlambatan proyek.
4	<i>Team leader</i>	Aktor yang memimpin jalannya proyek dengan mengelola tugas-tugas tenaga ahli atau tenaga pendukung dan menutup tugas.
5	Tenaga ahli	Aktor yang mengerjakan tugas yang telah diberikan oleh <i>team leader</i> .
6	Administrasi Proyek	Aktor yang menginput, mengubah dan mengirimkan laporan-laporan proyek.

Tabel 4.7 Identifikasi *Use Case*

No	<i>Use Case</i>	Deskripsi	Aktor
1	Kelola data user	Kegiatan mengatur hak akses <i>user</i> .	<i>Admin</i>
2	Login	Kegiatan memasukkan <i>username</i> dan <i>Password</i> agar dapat mengakses sistem. Masing-masing aktor mempunyai hak akses yang berbeda.	Semua aktor
3	Kelola data proyek	Kegiatan menginput, mengubah, dan menghapus data proyek.	<i>Admin</i>
4	Kelola data karyawan	Kegiatan menginput, mengubah, dan menghapus data karyawan.	<i>Admin</i>
5	Kelola data klien	Kegiatan menginput, mengubah, dan menghapus data klien perusahaan.	<i>Admin</i>
6	Kelola detail proyek	Kegiatan menginput, mengubah, dan menghapus jadwal proyek dan tim proyek.	Kepala divisi proyek
7	Verifikasi	Kegiatan menyetujui atau menolak jadwal, tim proyek dan laporan.	Direktur Teknis
8	Kelola tugas	Kegiatan menginput, mengubah, dan menghapus tugas-tugas tenaga ahli dan atau tenaga pendukung.	<i>Team leader</i>
9	<i>Update progress</i> tugas	Kegiatan update progress tugas.	Tenaga ahli/tenaga pendukung

10	Tutup tugas	Kegiatan menutup tugas yang telah selesai dikerjakan.	<i>Team leader</i>
11	Analisis keterlambatan proyek	Kegiatan menganalisis keterlambatan proyek dengan metode <i>critical path</i> method dan what if analysis.	Kepala divisi proyek
12	Kelola laporan	Kegiatan menginput, mengubah, dan menghapus laporan-laporan proyek.	Administrasi proyek
13	Tutup proyek	Kegiatan menutup proyek yang telah selesai.	Direktur teknis
14	Ubah profil	Kegiatan mengubah profil akun pada sistem.	Semua aktor
15	Kelola Pesan	Kegiatan mengirim dan menerima pesan.	Semua aktor
16	<i>Logout</i>	Pengguna keluar dari sistem.	Semua aktor

Setelah mengidentifikasi aktor dan kegiatan *use case*, berikut *use case diagram* sistem informasi manajemen proyek:



Gambar 4.5 Use Case Diagram Sistem Informasi Manajemen Proyek

Narasi *use case* digunakan untuk menggambarkan kegiatan yang dilakukan oleh aktor dan respon sistem.

Tabel 4.8 Narasi *Use Case* Kelola Data *User*

Use Case Name	Kelola data <i>user</i>	
Use Case Id	1	
Actor	<i>Admin</i>	
Description	<i>Use Case</i> ini menggambarkan kegiatan menambah, mengubah, menghapus dan melihat data-data pengguna sistem termasuk mengatur <i>role</i> untuk hak akses sistem	
Precondition	-	
Typical Course of Events	Actor Action <ul style="list-style-type: none"> 1. Klik Tambah 2. Menampilkan <i>Form Tambah User</i> 3. Mengisi data <i>user</i> pada <i>form user</i> 4. Klik Simpan 5. Validasi data 6. Menambah & menyimpan data <i>user</i> 7. Menampilkan pesan “Berhasil” 	System Response

		menambah data!”
	8. Klik OK	9. Menampilkan halaman data <i>user</i>
Alternate Courses	<p>4. Klik Batal, kembali ke langkah 9</p> <p>6. Jika validasi data gagal maka sistem akan menampilkan pesan kesalahan dan kembali pada langkah 2</p> <p>1a. Klik Ubah</p> <p>2a. Menampilkan <i>Form</i> Ubah Data</p> <p>3a. Mengubah data yang perlu diubah</p> <p>4a1. Klik Simpan</p> <p>5a1. Sistem memvalidasi data</p> <p>6a1.1. Mengubah & menyimpan data <i>user</i></p> <p>7a1.1. Menampilkan pesan “Berhasil mengubah data!”</p> <p>8a.1.1. Klik OK, kembali ke langkah 9</p> <p>6a1.2. Jika validasi data gagal maka sistem akan menampilkan pesan kesalahan dan kembali pada langkah 2.a</p> <p>4a2. Klik Batal, kembali ke langkah 9</p> <p>1b. Klik Hapus</p> <p>2b. Menampilkan pesan “Yakin ingin menghapus data?”</p> <p>3b1. Klik Iya</p> <p>4b1. Menghapus data <i>user</i></p>	

	<p>5b1. Menampilkan pesan “Berhasil menghapus data!”</p> <p>6b1. Klik OK, kembali ke langkah 9</p> <p>3b2. Klik Batal, kembali ke langkah 9</p>
Conclusion	<i>Admin berhasil melakukan salah satu aksi tambah, ubah atau hapus data user</i>
Post Condition	<i>Login</i>

Tabel 4.9 Narasi Use Case Login

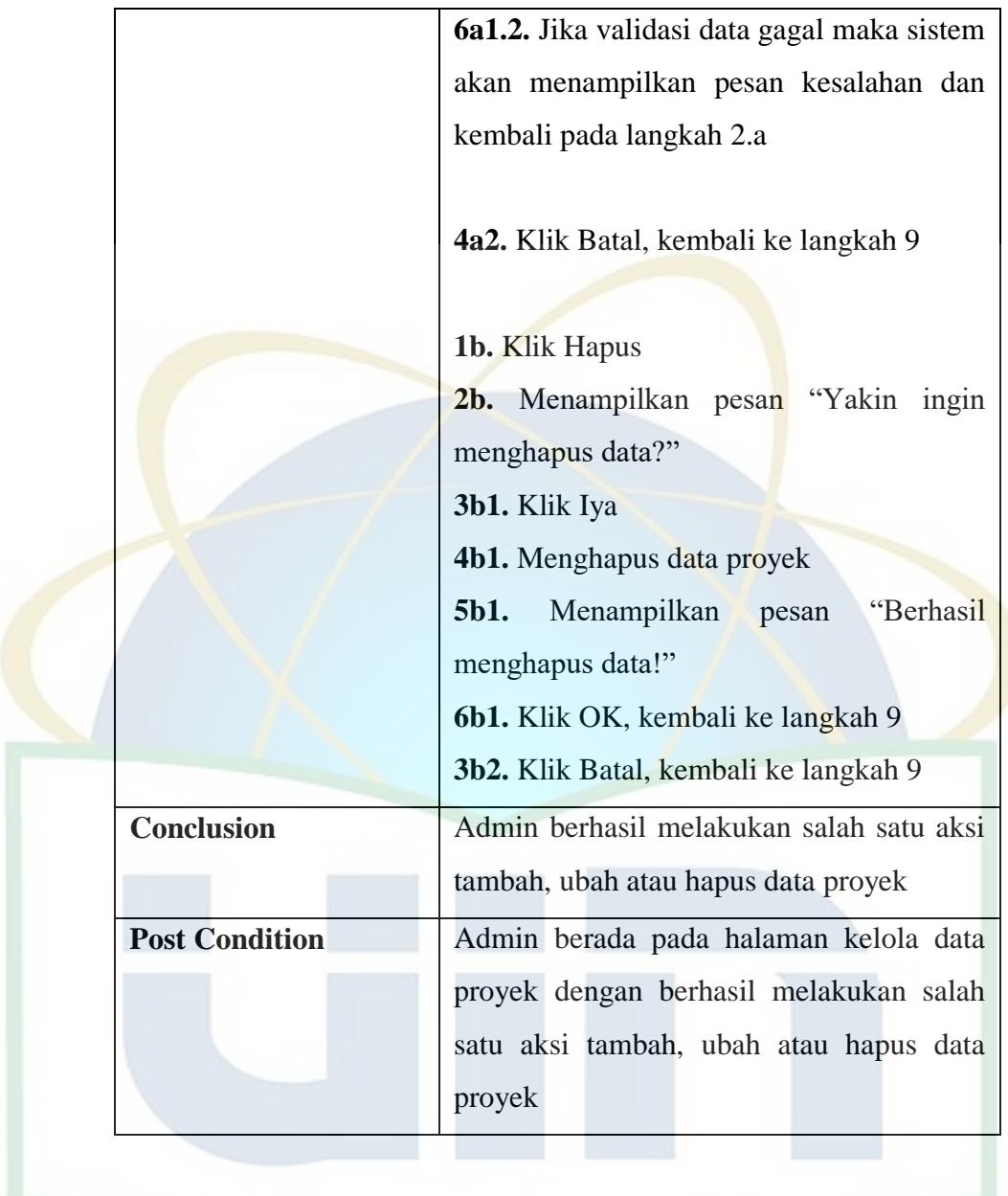
Use Case Name	<i>Login</i>							
Use Case Id	2							
Actor	<i>Admin, Direktur Teknis, Kepala divisi proyek, team leader, tenaga ahli/pendukung, administrasi proyek</i>							
Description	<i>Use Case menggambarkan kegiatan masukkan username dan password untuk diverifikasi oleh sistem</i>							
Precondition	<i>Kelola data user</i>							
Typical Course of Events	<table border="1"> <thead> <tr> <th>Actor Action</th> <th>System Response</th> </tr> </thead> <tbody> <tr> <td> <ol style="list-style-type: none"> 1. Input username dan password 2. Klik Login </td><td> <ol style="list-style-type: none"> 3. Validasi username dan password </td></tr> <tr> <td></td><td> <ol style="list-style-type: none"> 4. Jika validasi sesuai, sistem menampilkan </td></tr> </tbody> </table>	Actor Action	System Response	<ol style="list-style-type: none"> 1. Input username dan password 2. Klik Login 	<ol style="list-style-type: none"> 3. Validasi username dan password 		<ol style="list-style-type: none"> 4. Jika validasi sesuai, sistem menampilkan 	
Actor Action	System Response							
<ol style="list-style-type: none"> 1. Input username dan password 2. Klik Login 	<ol style="list-style-type: none"> 3. Validasi username dan password 							
	<ol style="list-style-type: none"> 4. Jika validasi sesuai, sistem menampilkan 							

		dashboard utama sesuai <i>role</i> aktor.
Alternate Courses	4. jika validasi <i>username</i> dan atau <i>password</i> tidak sesuai maka sistem akan menampilkan pesan kesalahan	
Conclusion	<i>Admin</i> , Direktur Teknis, Kepala divisi proyek, <i>team leader</i> , tenaga ahli/pendukung, administrasi proyek <i>login</i> ke sistem	
Post Condition	Masuk <i>dashboard</i> utama sesuai <i>role</i> aktor	

Tabel 4.10 Narasi *Use Case* Kelola Data Proyek

Use Case Name	Kelola data proyek	
Use Case Id	3	
Actor	<i>Admin</i>	
Description	<i>Use Case</i> ini menggambarkan kegiatan menambah, mengubah, menghapus data-data proyek	
Precondition	Aktor telah melakukan proses <i>login</i>	
Typical Course of Events	Actor Action	System Response
	1. Klik Tambah	2. Menampilkan <i>Form</i> Tambah Proyek
	3. Mengisi data proyek pada <i>form</i> proyek	

	4. Klik Simpan	5. Validasi data
	6. Menambah & menyimpan data proyek dan menambah data <i>notification</i>	
	7. Menampilkan pesan “Berhasil menambah data!”	
	8. Klik OK	9. Menampilkan halaman data proyek
Alternate Courses	<p>4. Klik Batal, kembali ke langkah 9</p> <p>6. Jika validasi data gagal maka sistem akan menampilkan pesan kesalahan dan kembali pada langkah 2</p> <p>1a. Klik Ubah</p> <p>2a. Menampilkan <i>Form</i> Ubah Data</p> <p>3a. Mengubah data yang perlu diubah</p> <p>4a1. Klik Simpan</p> <p>5a1. Sistem memvalidasi data</p> <p>6a1.1. Mengubah & menyimpan data proyek</p> <p>7a1.1. Menampilkan pesan “Berhasil mengubah data!”</p> <p>8a.1.1. Klik OK, kembali ke langkah 9</p>	



	<p>6a1.2. Jika validasi data gagal maka sistem akan menampilkan pesan kesalahan dan kembali pada langkah 2.a</p> <p>4a2. Klik Batal, kembali ke langkah 9</p>
	<p>1b. Klik Hapus</p> <p>2b. Menampilkan pesan “Yakin ingin menghapus data?”</p> <p>3b1. Klik Iya</p> <p>4b1. Menghapus data proyek</p> <p>5b1. Menampilkan pesan “Berhasil menghapus data!”</p> <p>6b1. Klik OK, kembali ke langkah 9</p> <p>3b2. Klik Batal, kembali ke langkah 9</p>
Conclusion	Admin berhasil melakukan salah satu aksi tambah, ubah atau hapus data proyek
Post Condition	Admin berada pada halaman kelola data proyek dengan berhasil melakukan salah satu aksi tambah, ubah atau hapus data proyek

Tabel 4.11 Narasi *Use Case* Kelola Data Karyawan

Use Case Name	Kelola data karyawan
Use Case Id	4
Actor	Admin

Description	<i>Use Case</i> ini menggambarkan kegiatan menambah, mengubah, menghapus data-data karyawan	
Precondition	Aktor telah melakukan proses <i>login</i>	
Typical Course of Events	Actor Action	System Response
	1. Klik Tambah 3. Mengisi data karyawan pada <i>form</i> karyawan 4. Klik Simpan 8. Klik OK	2. Menampilkan <i>Form</i> Tambah Karyawan 5. Validasi data 6. Menambah & menyimpan data karyawan 7. Menampilkan pesan “Berhasil menambah data!” 9. Menampilkan halaman data karyawan
Alternate Courses	4. Klik Batal, kembali ke langkah 9 6. Jika validasi data gagal maka sistem akan menampilkan pesan kesalahan dan kembali pada langkah 2 1a. Klik Ubah 2a. Menampilkan <i>Form</i> Ubah Data	

	<p>3a. Mengubah data yang perlu diubah</p> <p>4a1. Klik Simpan</p> <p>5a1. Sistem memvalidasi data</p> <p>6a1.1. Mengubah & menyimpan data karyawan</p> <p>7a1.1. Menampilkan pesan “Berhasil mengubah data!”</p> <p>8a.1.1. Klik OK, kembali ke langkah 9</p> <p>6a1.2. Jika validasi data gagal maka sistem akan menampilkan pesan kesalahan dan kembali pada langkah 2.a</p> <p>4a2. Klik Batal, kembali ke langkah 9</p> <p>1b. Klik Hapus</p> <p>2b. Menampilkan pesan “Yakin ingin menghapus data?”</p> <p>3b1. Klik Iya</p> <p>4b1. Menghapus data karyawan</p> <p>5b1. Menampilkan pesan “Berhasil menghapus data!”</p> <p>6b1. Klik OK, kembali ke langkah 9</p> <p>3b2. Klik Batal, kembali ke langkah 9</p>
Conclusion	Admin berhasil melakukan salah satu aksi tambah, ubah atau hapus data karyawan
Post Condition	Admin berada pada halaman kelola data proyek dengan berhasil melakukan salah satu aksi tambah, ubah atau hapus data karyawan

Tabel 4.12 Narasi Use Case Kelola Data Klien

Use Case Name	Kelola data klien	
Use Case Id	5	
Actor	Admin	
Description	<i>Use Case</i> ini menggambarkan kegiatan menambah, mengubah, menghapus data-data klien	
Precondition	Aktor telah melakukan proses <i>login</i>	
Typical Course of Events	Actor Action	System Response
	1. Klik Tambah 3. Mengisi data proyek pada form klien 4. Klik Simpan	2. Menampilkan Form Tambah Klien 5. Validasi data 6. Menambah & menyimpan data klien 7. Menampilkan pesan “Berhasil menambah data!”
	8. Klik OK	9. Menampilkan halaman data klien

Alternate Courses	<p>4. Klik Batal, kembali ke langkah 9</p> <p>6. Jika validasi data gagal maka sistem akan menampilkan pesan kesalahan dan kembali pada langkah 2</p> <p>1a. Klik Ubah</p> <p>2a. Menampilkan <i>Form</i> Ubah Data</p> <p>3a. Mengubah data yang perlu diubah</p> <p>4a1. Klik Simpan</p> <p>5a1. Sistem memvalidasi data</p> <p>6a1.1. Mengubah & menyimpan data klien</p> <p>7a1.1. Menampilkan pesan “Berhasil mengubah data!”</p> <p>8a.1.1. Klik OK, kembali ke langkah 9</p> <p>6a1.2. Jika validasi data gagal maka sistem akan menampilkan pesan kesalahan dan kembali pada langkah 2.a</p> <p>4a2. Klik Batal, kembali ke langkah 9</p> <p>1b. Klik Hapus</p> <p>2b. Menampilkan pesan “Yakin ingin menghapus data?”</p> <p>3b1. Klik Iya</p> <p>4b1. Menghapus data klien</p> <p>5b1. Menampilkan pesan “Berhasil menghapus data!”</p> <p>6b1. Klik OK, kembali ke langkah 9</p> <p>3b2. Klik Batal, kembali ke langkah 9</p>
Conclusion	Admin berhasil melakukan salah satu aksi tambah, ubah atau hapus data klien

Post Condition	Admin berada pada halaman kelola data proyek dengan berhasil melakukan salah satu aksi tambah, ubah atau hapus data klien
-----------------------	---

Tabel 4.13 Narasi *Use Case* Kelola Detail Proyek

Use Case Name	Kelola detail proyek	
Use Case Id	6	
Actor	Kepala divisi proyek	
Description	<i>Use Case</i> ini menggambarkan kegiatan menambah, mengubah, menghapus jadwal proyek dan tim proyek	
Precondition	Kelola data proyek	
Typical Course of Events	Actor Action 1. Klik proyek yang akan dikelola	System Response 2. Menampilkan halaman detail proyek
	3. Klik Jadwal	4. Menampilkan halaman jadwal
	5. Klik Tambah	6. Menampilkan Form Tambah Data
	7. Mengisi data proyek pada form jadwal	
	8. Klik Simpan	9. Validasi data

		10. Menambah & menyimpan jadwal
		11. Menampilkan pesan “Berhasil menambah jadwal!”
Alternate Courses	<p>8. Klik Batal, kembali ke langkah 4</p> <p>10. Jika validasi data gagal maka sistem akan menampilkan pesan kesalahan dan kembali pada langkah 4</p> <p>5.a. Klik Ubah</p> <p>6.a. Menampilkan <i>Form</i> Ubah Data</p> <p>7.a. Mengubah data yang perlu diubah</p> <p>8.a.1. Klik Simpan</p> <p>9.a.1. Sistem memvalidasi data</p> <p>10.a.1.1. Mengubah & menyimpan data jadwal</p> <p>11.a.1.1. Menampilkan pesan “Berhasil mengubah data!”</p> <p>10.a.1.2. Jika validasi data gagal maka sistem akan menampilkan pesan kesalahan dan kembali pada langkah 4</p> <p>8.a.2. Klik Batal, kembali ke langkah 4</p> <p>5.b. Klik Hapus</p> <p>6.b. Menampilkan pesan “Yakin ingin menghapus data?”</p> <p>7.b.1. Klik Iya</p>	

	<p>8.b.1. Menghapus jadwal</p> <p>9.b.1. Menampilkan pesan “Berhasil menghapus data!”</p> <p>10.b.2. Klik Batal, kembali ke langkah 4</p> <p>3.a. Klik Tim</p> <p>4.a.1. Menampilkan halaman tim proyek</p> <p>5.a.1. Klik Tambah</p> <p>6.a.1 Menampilkan <i>Form</i> Tambah Data</p> <p>7.a.1 Mengisi data yang perlu diubah</p> <p>8.a.1.1 Klik Simpan</p> <p>9.a.1.1 Sistem memvalidasi data</p> <p>10.a.1.1.1 Menambah & menyimpan data tim dan menambah data <i>notification</i></p> <p>11.a.1.1.1 Menampilkan pesan “Berhasil menambah data!”</p> <p>10.a.1.1.2. Jika validasi data gagal maka sistem akan menampilkan pesan kesalahan dan kembali pada langkah 4.a.1</p> <p>8.a.1.2. Klik Batal, kembali ke langkah 4.a.1</p> <p>5.a.2. Klik Ubah</p> <p>6.a.2. Menampilkan <i>Form</i> Ubah Data</p> <p>7.a.2. Mengubah data yang perlu diubah</p> <p>8.a.2.1. Klik Simpan</p> <p>9.a.2.1. Sistem memvalidasi data</p>
--	--

	<p>10.a.2.1.1 Mengubah & menyimpan data jadwal</p> <p>11.a.2.1.1. Menampilkan pesan “Berhasil mengubah data!”</p> <p>10.a.2.1.2. Jika validasi data gagal maka sistem akan menampilkan pesan “Gagal mengubah data!” dan kembali pada langkah 4.a.1</p> <p>8.a.2.2 Klik Batal, kembali ke langkah 4.a.1</p> <p>5.a.3 Klik Hapus</p> <p>6.a.3 Menampilkan pesan “Yakin ingin menghapus data?”</p> <p>7.a.3.1. Klik Iya</p> <p>8.a.3.1. Menghapus jadwal</p> <p>9.a.3.1. Menampilkan pesan “Berhasil menghapus data!”</p> <p>7.a.3.2. Klik Batal, kembali ke langkah 4.a.1</p>
Conclusion	Kepala divisi proyek berhasil melakukan salah satu aksi tambah, ubah atau hapus jadwal dan tim proyek
Post Condition	Kepala divisi proyek berada pada halaman kelola data proyek dengan berhasil melakukan salah satu aksi tambah, ubah atau hapus jadwal dan tim proyek

Tabel 4.14 Narasi Use Case Verifikasi

Use Case Name	Verifikasi	
Use Case Id	7	
Actor	Direktur Teknis	
Description	<p><i>Use Case</i> ini menggambarkan kegiatan menyetujui atau menolak jadwal, tim proyek, laporan dan hasil analisis keterlambatan</p>	
Precondition	Kelola detail proyek dan atau kelola laporan	
Typical Course of Events	Actor Action	System Response
	1. Klik Jadwal	2. Menampilkan halaman verifikasi jadwal
	3. Klik Verifikasi	4. Menampilkan Form “Yakin untuk memverifikasi jadwal”
	5. Klik Setuju	6. Validasi data
		7. Menyimpan & mengganti status data jadwal menjadi Diverifikasi
		8. Menampilkan pesan “Berhasil”

		menyetuji data!”
Alternate Courses	<p>7. Jika validasi data gagal maka sistem akan menampilkan pesan kesalahan dan kembali pada tahap 4</p> <p>3.a. Klik Tolak</p> <p>4.a Menampilkan <i>Form</i> “Yakin untuk menolak jadwal”</p> <p>5.a. Isi alasan dan klik Kirim</p> <p>6.a. Sistem memvalidasi data</p> <p>7.a.1. Jika validasi berhasil, Menyimpan & mengganti status data jadwal menjadi Ditolak</p> <p>8.a.1. Menampilkan pesan “Berhasil menolak data!”</p> <p>7.a.2. Jika validasi gagal, menampilkan pesan kesalahan dan kembali pada tahap 4.a</p> <p>1.a. Klik Tim</p> <p>2.a.1 Menampilkan halaman verifikasi tim proyek</p> <p>3.a.1 Klik Verifikasi pada tim proyek yang ingin diverifikasi</p> <p>4.a.1 Menampilkan <i>Form</i> “Yakin untuk memverifikasi tim”</p> <p>5.a.1 Klik Setuju</p> <p>6.a.1. Sistem memvalidasi data</p> <p>7.a.1.1 Menyimpan & mengganti status data tim menjadi Diverifikasi</p>	

	<p>8.a.1.1 Menampilkan pesan “Berhasil menyetujui data!”</p> <p>7.a.1.2 Jika validasi data gagal maka sistem akan menampilkan pesan kesalahan dan kembali pada tahap 4.a.1</p> <p>5.a.2 Klik Tolak</p> <p>6.a.2 Menampilkan <i>Form</i> “Yakin untuk menolak tim”</p> <p>7.a.2. Isi alasan dan Klik Kirim</p> <p>8.a.2 Sistem memvalidasi data</p> <p>9.a.2.1 Menyimpan & mengganti status data tim menjadi Ditolak</p> <p>10.a.2.1 Menampilkan pesan “Berhasil menolak data!”</p> <p>9.a.2.2 Jika validasi gagal, menampilkan pesan kesalahan dan kembali pada tahap 6.a.2</p> <p>1.b. Klik Laporan</p> <p>2.b.1 Menampilkan halaman verifikasi laporan proyek</p> <p>3.b.1 Klik Verifikasi pada laporan yang ingin diverifikasi</p> <p>4.b.1 Menampilkan <i>Form</i> “Yakin untuk memverifikasi laporan”</p> <p>5.b.1 Klik Setuju</p> <p>6.b.1. Sistem memvalidasi data</p> <p>7.b.1.1 Menyimpan & mengganti status data laporan menjadi Diverifikasi</p>
--	---

	<p>8.b.1.1 Menampilkan pesan “Berhasil menyetujui data!” dan menambah data <i>notification</i></p> <p>7.b.1.2 Jika validasi data gagal maka sistem akan menampilkan pesan kesalahan dan kembali pada tahap 4.b.1</p> <p>5.b.2 Klik Tolak</p> <p>6.b.2 Menampilkan <i>Form</i> “Yakin untuk menolak laporan”</p> <p>7.b.2. Isi alasan dan Klik Kirim</p> <p>8.b.2 Sistem memvalidasi data</p> <p>9.b.2.1 Menyimpan & mengganti status data laporan menjadi Ditolak</p> <p>10.b.2.1 Menampilkan pesan “Berhasil menolak data!”</p> <p>9.b.2.2 Jika validasi gagal, menampilkan pesan kesalahan dan kembali pada tahap 6.b.2</p> <p>1.c. Klik Analisis</p> <p>2.c.1 Menampilkan halaman verifikasi analisis</p> <p>3.c.1 Klik Verifikasi pada hasil analisis yang ingin diverifikasi</p> <p>4.c.1 Menampilkan <i>Form</i> “Yakin untuk memverifikasi analisis”</p> <p>5.c.1 Klik Setuju</p> <p>6.c.1. Sistem memvalidasi data</p>
--	---

	<p>7.c.1.1 Menyimpan & mengganti status data analisis menjadi Diverifikasi</p> <p>8.c.1.1 Menampilkan pesan “Berhasil menyetujui data!”</p> <p>7.c.1.2 Jika validasi data gagal maka sistem akan menampilkan pesan kesalahan dan kembali pada tahap 4.b.1</p> <p>5.c.2 Klik Tolak</p> <p>6.c.2 Menampilkan <i>Form</i> “Yakin untuk menolak analisis”</p> <p>7.c.2. Isi alasan dan Klik Kirim</p> <p>8.c.2 Sistem memvalidasi data</p> <p>9.c.2.1 Menyimpan & mengganti status data analisis menjadi Ditolak</p> <p>10.c.2.1 Menampilkan pesan “Berhasil menolak data!”</p> <p>9.c.2.2 Jika validasi gagal, menampilkan pesan kesalahan dan kembali pada tahap 6.c.2</p>
Conclusion	Direktur Teknis berhasil memverifikasi jadwal proyek, tim proyek, laporan dan hasil analisis proyek
Post Condition	Direktur Teknis berada pada halaman jadwal dengan data yang diverifikasi statusnya sudah berubah

Tabel 4.15 Narasi Use Case Kelola Tugas

Use Case Name	Kelola tugas	
Use Case Id	8	
Actor	<i>Team leader</i>	
Description	<i>Use Case</i> ini menggambarkan kegiatan menambah, mengubah, menghapus tugas	
Precondition	-	
Typical Course of Events	Actor Action <ol style="list-style-type: none"> 1. Klik Tambah 3. Mengisi data pada form tugas 4. Klik Simpan 	System Response <ol style="list-style-type: none"> 2. Menampilkan <i>Form Tambah Tugas</i> 5. Validasi data 6. Menambah & mengganti data tugas dan menambah data <i>notification</i> 7. Menampilkan pesan “Berhasil menambah data!”
	8. Klik OK	9. Menampilkan halaman tugas

Alternate Courses	<p>4. Klik Batal, kembali ke langkah 9</p> <p>6. Jika validasi data gagal maka sistem akan menampilkan pesan kesalahan dan kembali pada langkah 2</p> <p>1a. Klik Ubah</p> <p>2a. Menampilkan <i>Form</i> Ubah Data</p> <p>3a. Mengubah data yang perlu diubah</p> <p>4a1. Klik Simpan</p> <p>5a1. Sistem memvalidasi data</p> <p>6a1.1. Mengubah & menyimpan data tugas</p> <p>7a1.1. Menampilkan pesan “Berhasil mengubah data!”</p> <p>8a.1.1. Klik OK, kembali ke langkah 9</p> <p>6a1.2. Jika validasi data gagal maka sistem akan menampilkan pesan kesalahan dan kembali pada langkah 2.a</p> <p>4a2. Klik Batal, kembali ke langkah 9</p> <p>1b. Klik Hapus</p> <p>2b. Menampilkan pesan “Yakin ingin menghapus data?”</p> <p>3b1. Klik Iya</p> <p>4b1. Menghapus data tugas</p> <p>5b1. Menampilkan pesan “Berhasil menghapus data!”</p> <p>6b1. Klik OK, kembali ke langkah 9</p> <p>3b2. Klik Batal, kembali ke langkah 9</p>
Conclusion	<i>Team leader</i> berhasil melakukan salah satu aksi tambah, ubah atau hapus tugas

Post Condition	<i>Team leader</i> berada pada halaman kelola tugas dengan berhasil melakukan salah satu aksi tambah, ubah atau hapus tugas
-----------------------	---

Tabel 4.16 Narasi *Use Case Update progress tugas*

Use Case Name	<i>Update progress tugas</i>	
Use Case Id	9	
Actor	Tenaga ahli/pendukung	
Description	<i>Use Case</i> ini menggambarkan kegiatan melaporkan progress tugas proyek	
Precondition	Kelola tugas	
Typical Course of Events	Actor Action <ul style="list-style-type: none"> 1. Klik Tugas yang ingin dikerjakan 3. Klik Mulai 5. Mengisi hasil tugas dan mengunggah file atau dokumen tugas 	System Response <ul style="list-style-type: none"> 2. Menampilkan <i>Form</i> detail tugas yang akan dikerjakan 4. Mengganti status tugas menjadi “Sedang Dikerjakan” 6. Menampilkan pesan “Berhasil mengunggah data!”

	7. Klik Selesai	8. Mengganti status tugas menjadi “Butuh Koreksi” dan menambah data <i>notification</i>
		9. Menampilkan pesan “Berhasil menambahkan data!”
	10. Klik OK	11. Menampilkan halaman tugas
Alternate Courses	3. Klik Kembali, kembali pada tahap 11	
Conclusion	Tenaga ahli/pendukung berhasil mengupdate progress tugas proyek	
Post Condition	Tenaga ahli/pendukung berada pada halaman tugas dengan data yang dikerjakan statusnya sudah berubah	

Tabel 4.17 Narasi *Use Case* Tutup Tugas

Use Case Name	Tutup tugas	
Use Case Id	10	
Actor	<i>Team leader</i>	
Description	<i>Use Case</i> ini menggambarkan kegiatan menutup tugas	
Precondition	<i>Update</i> progress tugas	
	Actor Action	System Response

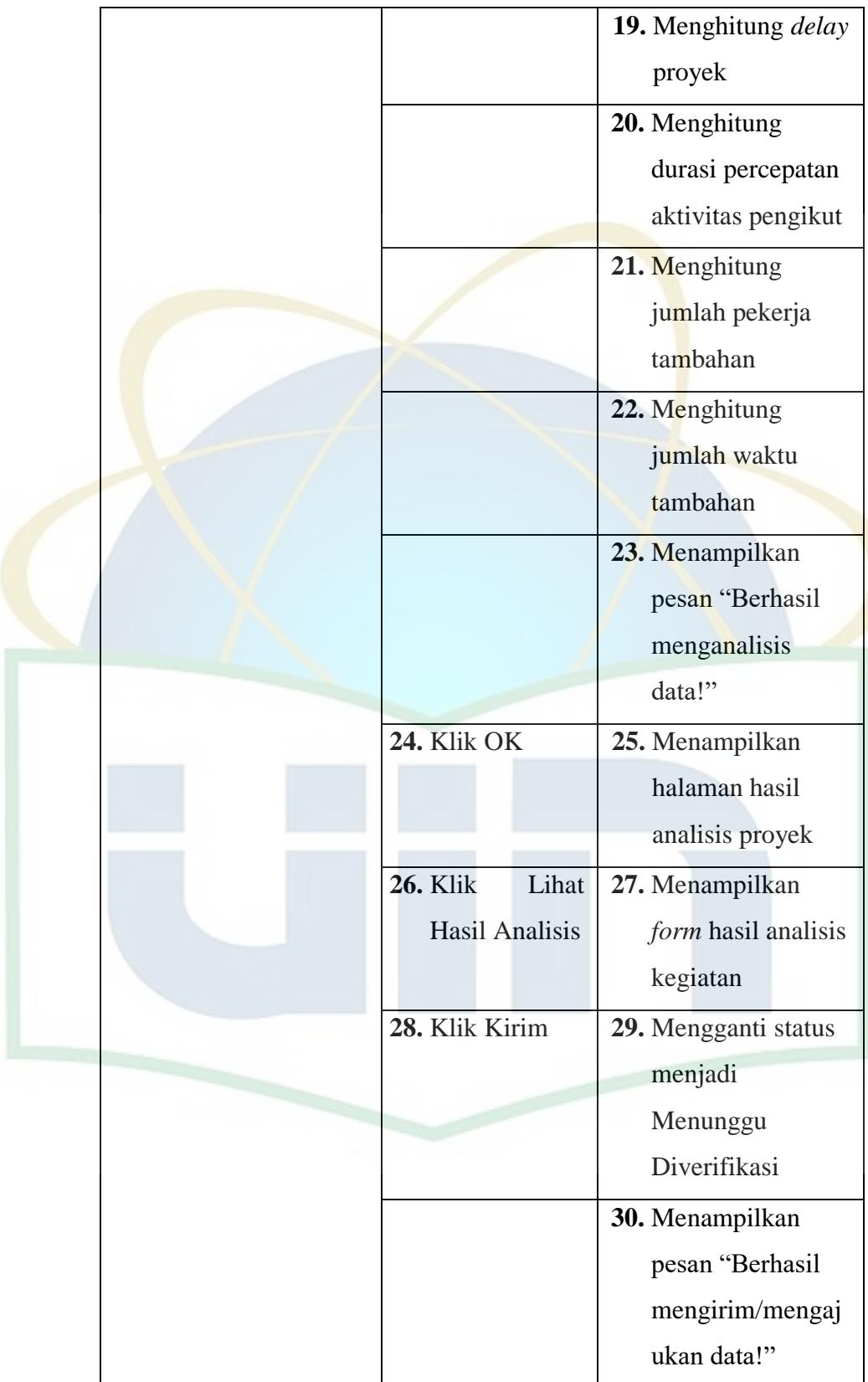
Typical Course of Events	1. Klik Tugas yang ingin ditutup	2. Menampilkan <i>Form</i> detail tugas yang akan ditutup
	3. Lihat <i>file</i> atau dokumen yang diunggah tenaga ahli	
	4. Klik tutup	5. Mengganti status data tugas menjadi “Selesai”
		6. Menampilkan pesan “Berhasil menutup tugas!”
	7. Klik OK	8. Menampilkan halaman data tugas
	4.a Klik Revisi	
	5.a Menampilkan <i>Form</i> Revisi	
	6.a Isi alasan dan Klik Simpan Revisi	
Alternate Courses	7.a Sistem memvalidasi data	
	8.a.1 Menyimpan & mengganti status data tugas menjadi Revisi	
	9.a.1 Menampilkan pesan “Berhasil merevisi data!”	
	10.a.1 Klik OK, kembali ke tahap 8	

	8.a.2 Jika validasi gagal, menampilkan pesan kesalahan dan kembali pada tahap 5.a
Conclusion	<i>Team leader</i> berhasil menutup tugas
Post Condition	<i>Team leader</i> berada pada halaman tugas dengan data yang ditutup statusnya sudah berubah

Tabel 4.18 Narasi *Use Case* Analisis Keterlambatan Proyek

Use Case Name	Analisis keterlambatan proyek	
Use Case Id	11	
Actor	Kepala divisi proyek	
Description	<i>Use Case</i> ini menggambarkan kegiatan membuat jalur kritis dari kegiatan-kegiatan proyek dengan metode <i>critical path method</i> dan menganalisis skenario keterlambatan menggunakan metode <i>what if analysis</i>	
Precondition	Kelola detail proyek	
Typical Course of Events	Actor Action	System Response
	1. Klik proyek yang akan dianalisis	2. Menampilkan halaman detail proyek
	3. Klik Analisis	4. Menampilkan halaman analisis
	5. Klik Buat Jalur Kritis	6. Menghitung <i>earliest activity</i>

		<i>start time (ES), earliest activity finish time (EF)</i>
	7. Menghitung <i>latest activity start time</i> (LS), <i>latest activity finish time</i> (LF)	
	8. Menghitung <i>float</i> atau <i>slack</i>	
	9. Menentukan jalur kritis	
	10. Menampilkan pesan “Berhasil membuat data!”	
11. Klik OK	12. Menampilkan hasil jalur kritis proyek	
13. Klik Analisis pada kegiatan yang terlambat	14. Menampilkan form Analisis Keterlambatan	
15. Mengisi detail keterlambatan		
16. Klik Simpan	17. Validasi data	
	18. Menghitung persentase keterlambatan	



	19. Menghitung <i>delay</i> proyek
	20. Menghitung durasi percepatan aktivitas pengikut
	21. Menghitung jumlah pekerja tambahan
	22. Menghitung jumlah waktu tambahan
	23. Menampilkan pesan “Berhasil menganalisis data!”
24. Klik OK	25. Menampilkan halaman hasil analisis proyek
26. Klik Lihat Hasil Analisis	27. Menampilkan <i>form</i> hasil analisis kegiatan
28. Klik Kirim	29. Mengganti status menjadi Menunggu Diverifikasi
	30. Menampilkan pesan “Berhasil mengirim/mengajukan data!”

Alternate Courses	<p>16. Klik Batal, kembali pada tahap 4</p> <p>18. Jika validasi gagal, menampilkan pesan kesalahan, kembali pada tahap 14</p> <p>28. Klik Batal, kembali pada tahap 4</p>
Conclusion	Kepala divisi proyek berhasil membuat jalur kritis proyek dan analisis percepatan
Post Condition	Kepala divisi proyek berada pada halaman jalur kritis proyek dan analisis percepatan

Tabel 4.19 Narasi *Use Case* Kelola Laporan

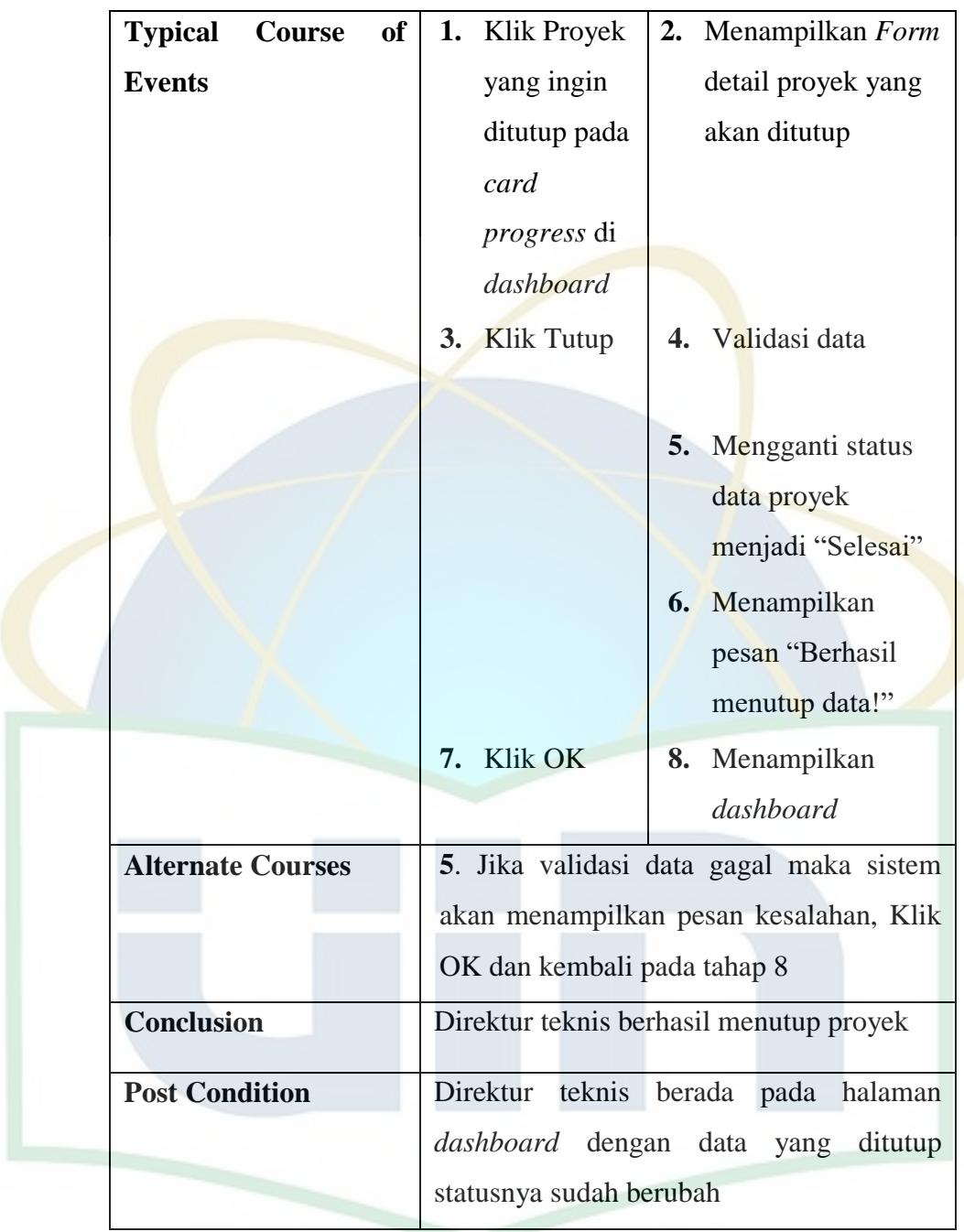
Use Case Name	Kelola laporan	
Use Case Id	12	
Actor	Administrasi proyek	
Description	<i>Use Case</i> ini menggambarkan kegiatan menambah, mengubah, menghapus laporan	
Precondition	-	
Typical Course of Events	Actor Action 1. Klik Tambah 3. Mengisi data laporan pada form laporan 4. Klik Simpan	System Response 2. Menampilkan Form Tambah Laporan 5. Validasi data

		<p>6. Menambah & mengganti data laporan</p> <p>7. Menampilkan pesan “Berhasil menambah data!”</p> <p>8. Klik OK</p> <p>9. Menampilkan halaman laporan</p>
Alternate Courses		<p>4. Klik Batal, kembali ke langkah 9</p> <p>6. Jika validasi data gagal maka sistem akan menampilkan pesan kesalahan dan kembali pada langkah 2</p> <p>1a. Klik Ubah</p> <p>2a. Menampilkan <i>Form</i> Ubah Data</p> <p>3a. Mengubah data yang perlu diubah</p> <p>4a1. Klik Simpan</p> <p>5a1. Sistem memvalidasi data</p> <p>6a1.1. Mengubah & menyimpan data laporan</p> <p>7a1.1. Menampilkan pesan “Berhasil mengubah data!”</p> <p>8a.1.1. Klik OK, kembali ke langkah 9</p> <p>6a1.2. Jika validasi data gagal maka sistem akan menampilkan pesan kesalahan dan kembali pada langkah 2.a</p> <p>4a2. Klik Batal, kembali ke langkah 9</p>

	<p>1b. Klik Hapus</p> <p>2b. Menampilkan pesan “Yakin ingin menghapus data?”</p> <p>3b1. Klik Iya</p> <p>4b1. Menghapus data laporan</p> <p>5b1. Menampilkan pesan “Berhasil menghapus data!”</p> <p>6b1. Klik OK, kembali ke langkah 9</p> <p>3b2. Klik Batal, kembali ke langkah 9</p>
Conclusion	Administrasi proyek berhasil melakukan salah satu aksi tambah, ubah atau hapus laporan
Post Condition	Administrasi proyek berada pada halaman kelola laporan dengan berhasil melakukan salah satu aksi tambah, ubah atau hapus laporan

Tabel 4.20 Narasi *Use Case* Tutup Proyek

Use Case Name	Tutup proyek	
Use Case Id	13	
Actor	Direktur teknis	
Description	<i>Use Case</i> ini menggambarkan kegiatan menutup proyek	
Precondition	Verifikasi	
	Actor Action	System Response



Typical Course of Events	<p>1. Klik Proyek yang ingin ditutup pada <i>card progress</i> di <i>dashboard</i></p> <p>3. Klik Tutup</p> <p>7. Klik OK</p>	<p>2. Menampilkan <i>Form detail</i> proyek yang akan ditutup</p> <p>4. Validasi data</p> <p>5. Mengganti status data proyek menjadi “Selesai”</p> <p>6. Menampilkan pesan “Berhasil menutup data!”</p> <p>8. Menampilkan <i>dashboard</i></p>
Alternate Courses	<p>5. Jika validasi data gagal maka sistem akan menampilkan pesan kesalahan, Klik OK dan kembali pada tahap 8</p>	
Conclusion	Direktur teknis berhasil menutup proyek	
Post Condition	Direktur teknis berada pada halaman <i>dashboard</i> dengan data yang ditutup statusnya sudah berubah	

Tabel 4.21 Narasi *Use Case* Ubah Profil

Use Case Name	Ubah Profil
Use Case Id	14

Actor	Admin, Direktur Teknis, Kepala divisi proyek, <i>team leader</i> , tenaga ahli/pendukung, administrasi proyek	
Description	<i>Use Case</i> ini menggambarkan kegiatan mengubah profil akun	
Precondition	Aktor telah melakukan proses <i>login</i>	
Typical Course of Events	Actor Action	System Response
	1. Klik Profil	2. Menampilkan <i>Form</i> profil akun
	3. Klik Ubah	4. Menampilkan <i>form</i> ubah profil
	5. Mengubah data profil yang ingin diubah	
	6. Klik Simpan	7. Validasi data
		8. Mengubah data user
		9. Menampilkan pesan “Berhasil mengubah data!”
Alternate Courses	7. Jika validasi data gagal maka sistem akan menampilkan pesan kesalahan dan kembali pada langkah 4	
Conclusion	Semua aktor berhasil mengubah profil akun	

Post Condition	Semua aktor berada pada halaman profil dengan data yang sudah diperbarui
-----------------------	--

Tabel 4.22 Narasi *Use Case* Kelola Pesan

Use Case Name	Kelola Pesan		
Use Case Id	15		
Actor	<i>Admin, Direktur Teknis, Kepala divisi proyek, team leader, tenaga ahli/pendukung, administrasi proyek</i>		
Description	<i>Use Case</i> ini menggambarkan mengirim dan menerima pesan		
Precondition	Aktor telah melakukan proses <i>login</i>		
Typical Course of Events	Actor Action	System Response	
	1. Klik Pesan	2. Menampilkan daftar pesan	
	3. Klik Cari Pengguna atau Buat dan pilih penerima	4. Menampilkan halaman pesan dengan orang yang dipilih	
	5. Isi pesan		
	6. Klik kirim	7. Mengirim pesan	
		8. Menampilkan pesan “Berhasil mengirim pesan!”	

Alternate Courses	<p>6. Klik kembali, dan kembali pada langkah 2</p> <p>3.a Klik salah satu daftar pesan yang mempunyai tanda pesan baru</p> <p>4.a Menampilkan halaman pesan baru dengan orang yang sudah dipilih</p>
Conclusion	Semua aktor berhasil mengelola pesan
Post Condition	Semua aktor berada pada daftar pesan

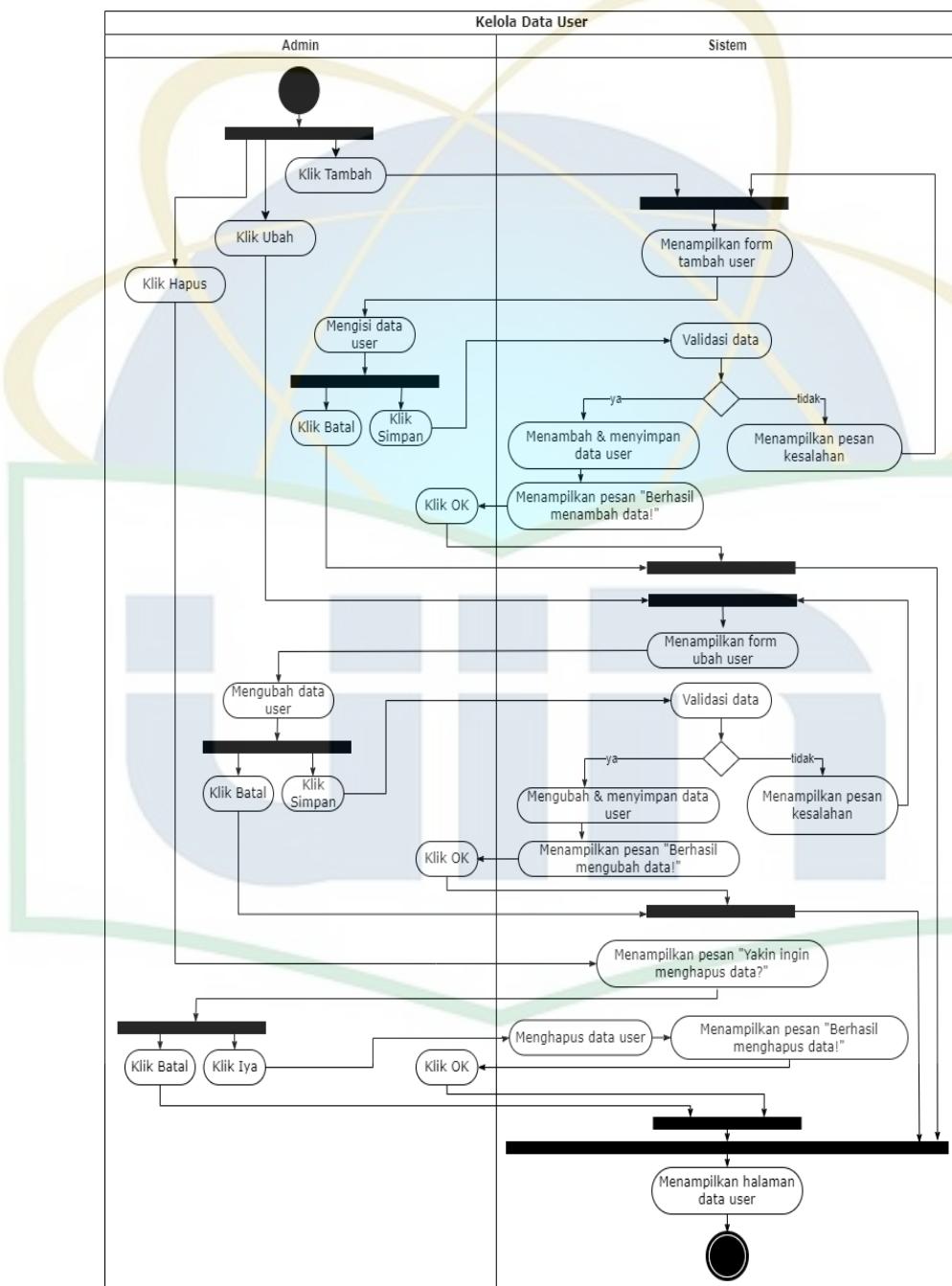
Tabel 4.23 Narasi *Use Case Logout*

Use Case Name	Logout	
Use Case Id	16	
Actor	<i>Admin</i> , Direktur Teknis, Kepala divisi proyek, <i>team leader</i> , tenaga ahli/pendukung, administrasi proyek	
Description	<i>Use Case</i> menggambarkan kegiatan keluar dari sistem	
Precondition	Aktor mengakses sistem	
Typical Course of Events	Actor Action 1. Klik Keluar	System Response 2. Menampilkan halaman <i>login</i>
Alternate Courses	-	
Conclusion	Semua aktor keluar dari sistem	
Post Condition	Berada pada halaman <i>login</i>	

4.2.2.2 Activity Diagram

Pada *activity diagram*, digambarkan alur proses pada masing-masing *use case*.

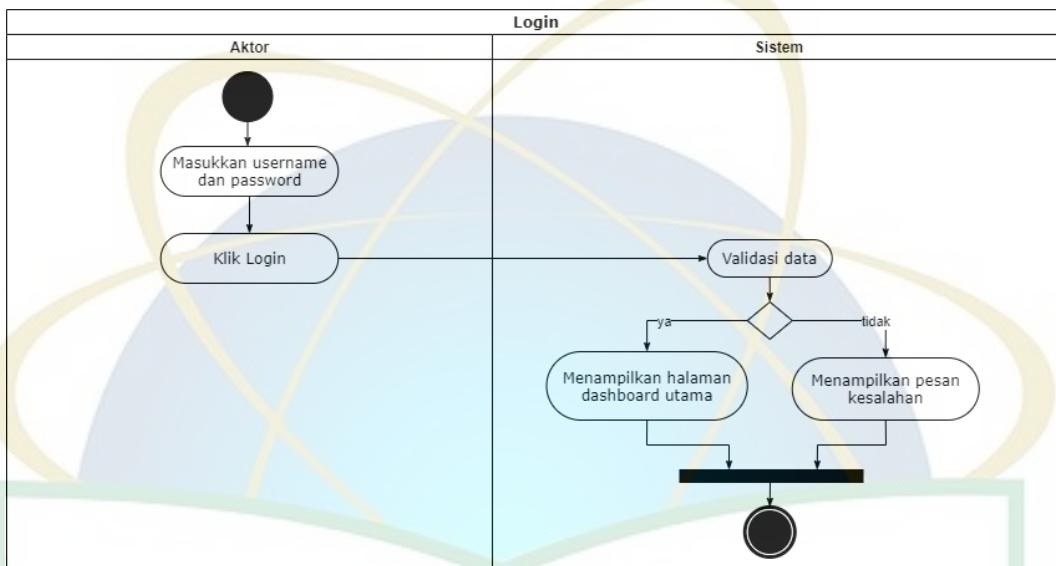
1. Activity Diagram Kelola Data User



Gambar 4.6 Activity Diagram Kelola Data User

Pada aktivitas kelola data user, admin melakukan pengelolaan data *user* dengan tambah, ubah, dan hapus data-data *user*.

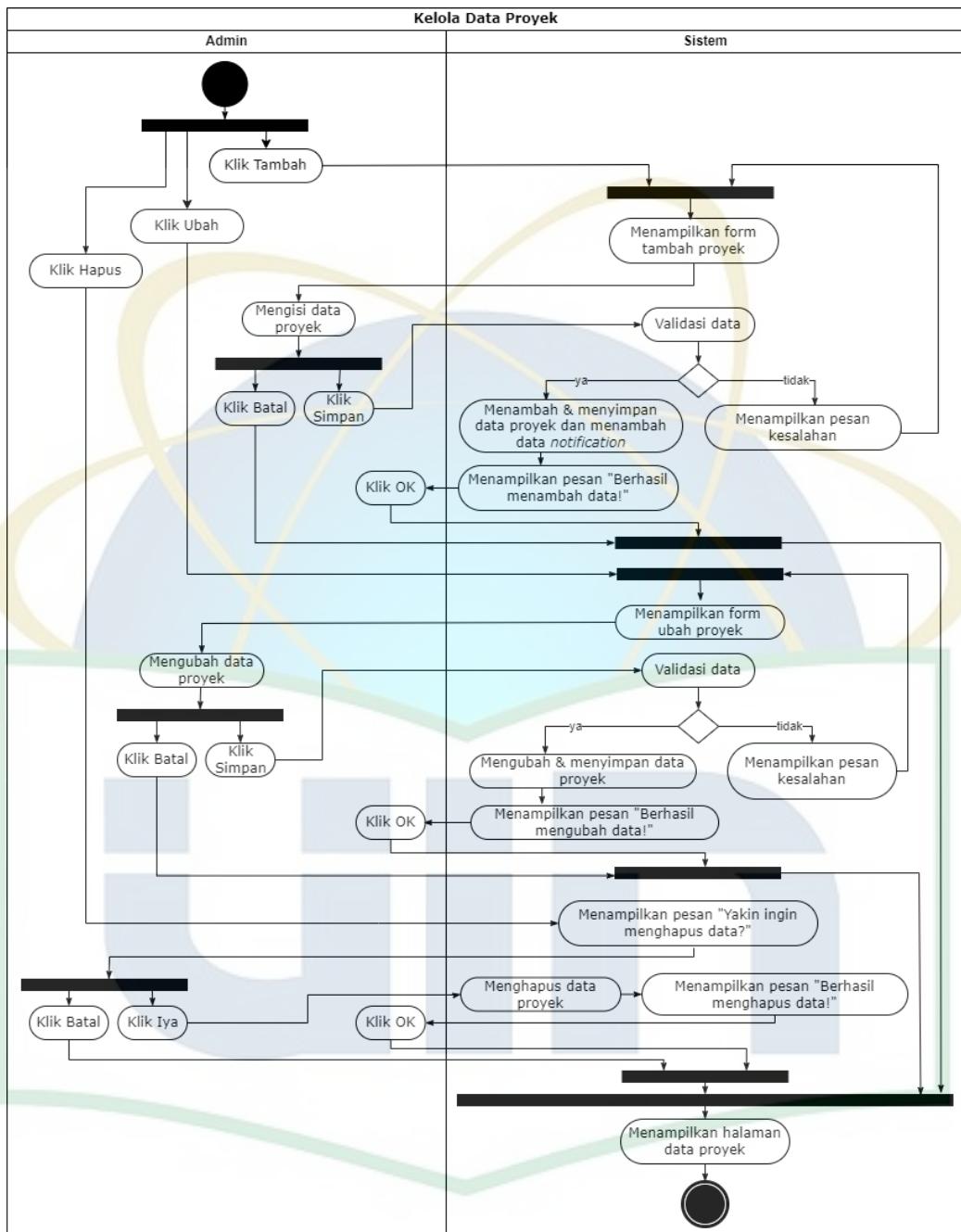
2. Activity Diagram Login



Gambar 4.7 Activity Diagram Login

Pada aktivitas *login*, seluruh actor yaitu admin, direktur teknis, kepala divisi proyek, *team leader*, tenaga ahli/pendukung, administrasi proyek mengisi *username* dan *password* untuk masuk ke sistem.

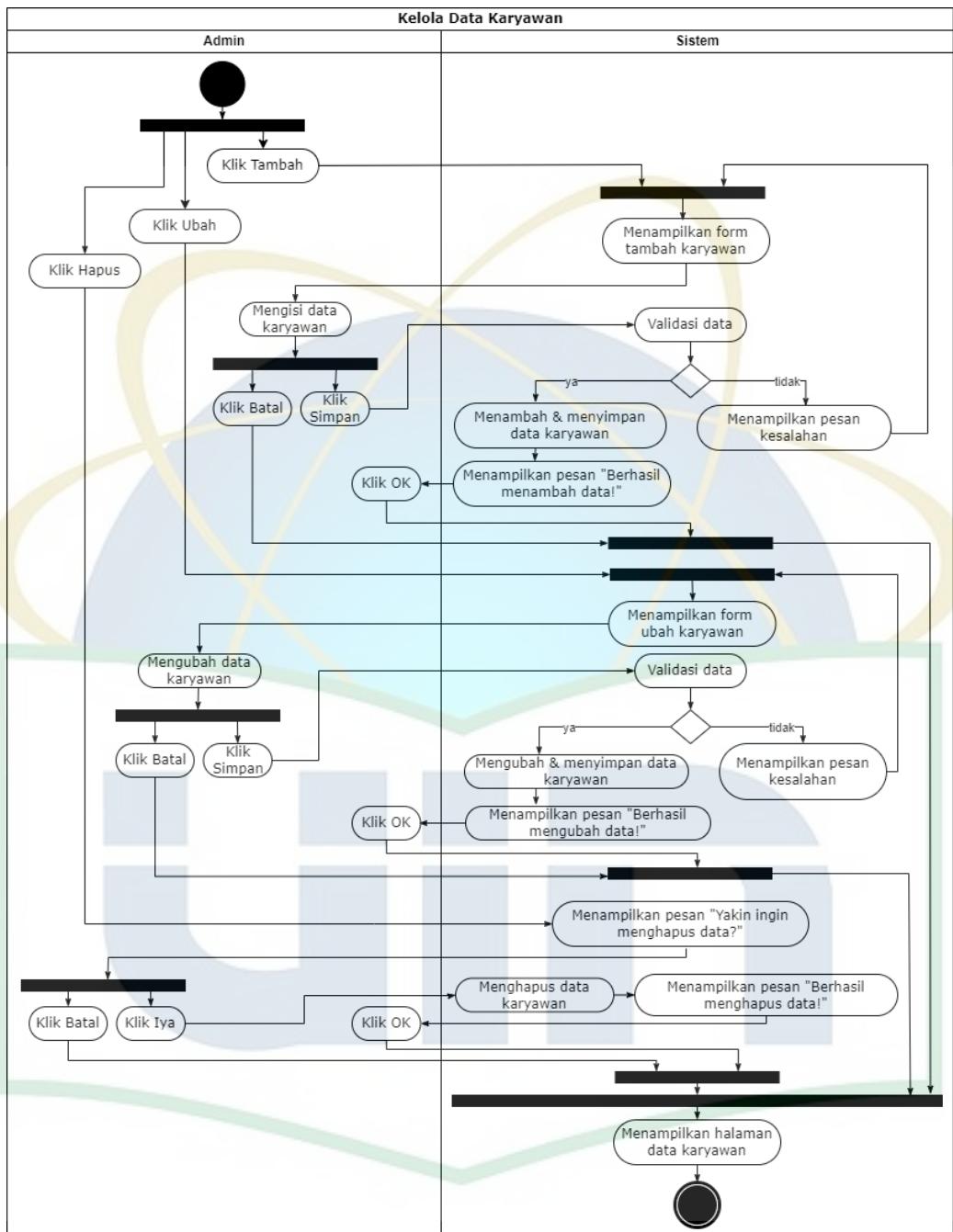
3. Activity Diagram Kelola Data Proyek



Gambar 4.8 Activity Diagram Kelola Data Proyek

Pada aktivitas kelola data proyek, admin melakukan pengelolaan data proyek dengan tambah, ubah, dan hapus proyek.

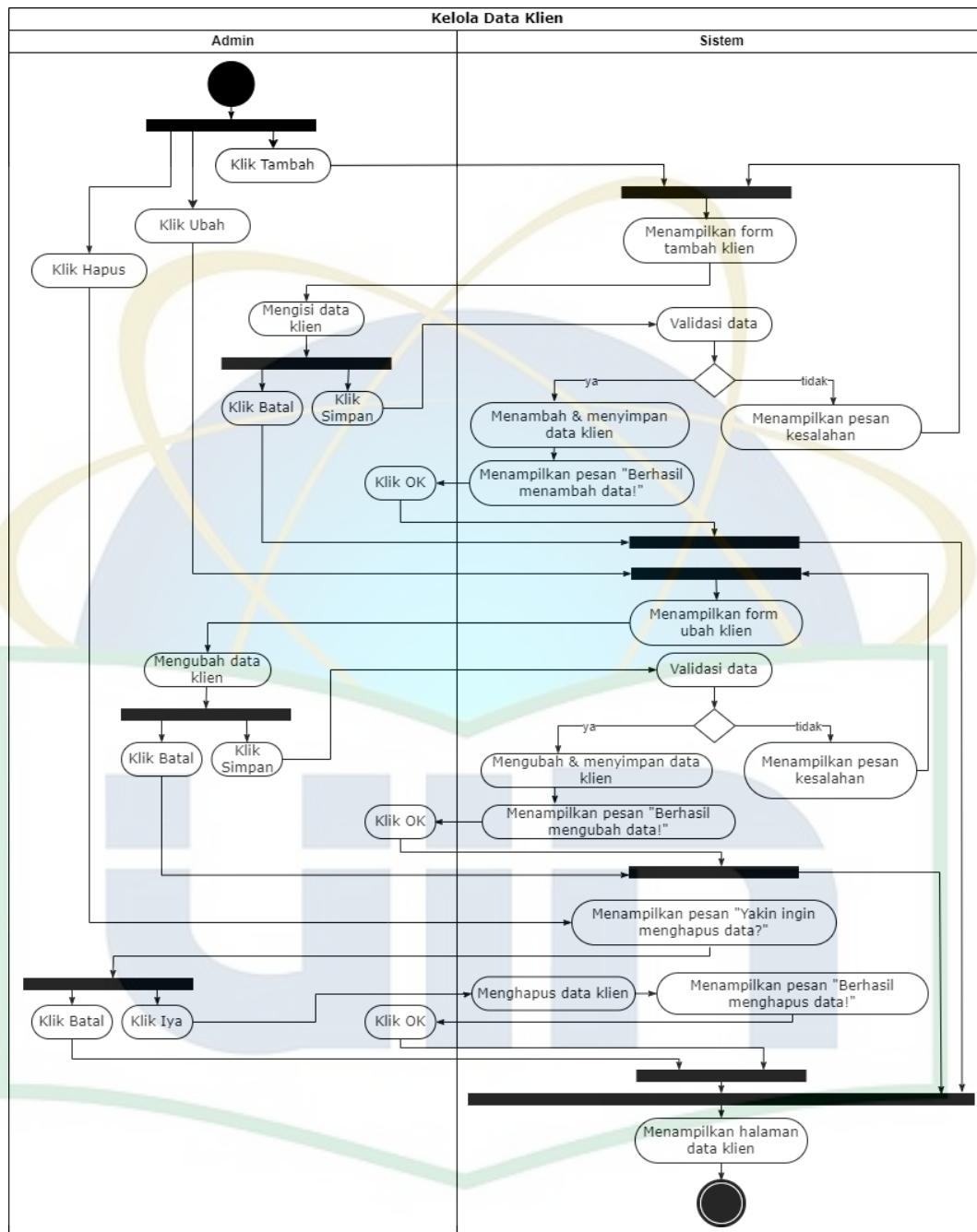
4. Activity Diagram Kelola Data Karyawan



Gambar 4.9 Activity Diagram Kelola Data Karyawan

Pada aktivitas kelola data karyawan, admin melakukan pengelolaan data karyawan dengan tambah, ubah, dan hapus karyawan.

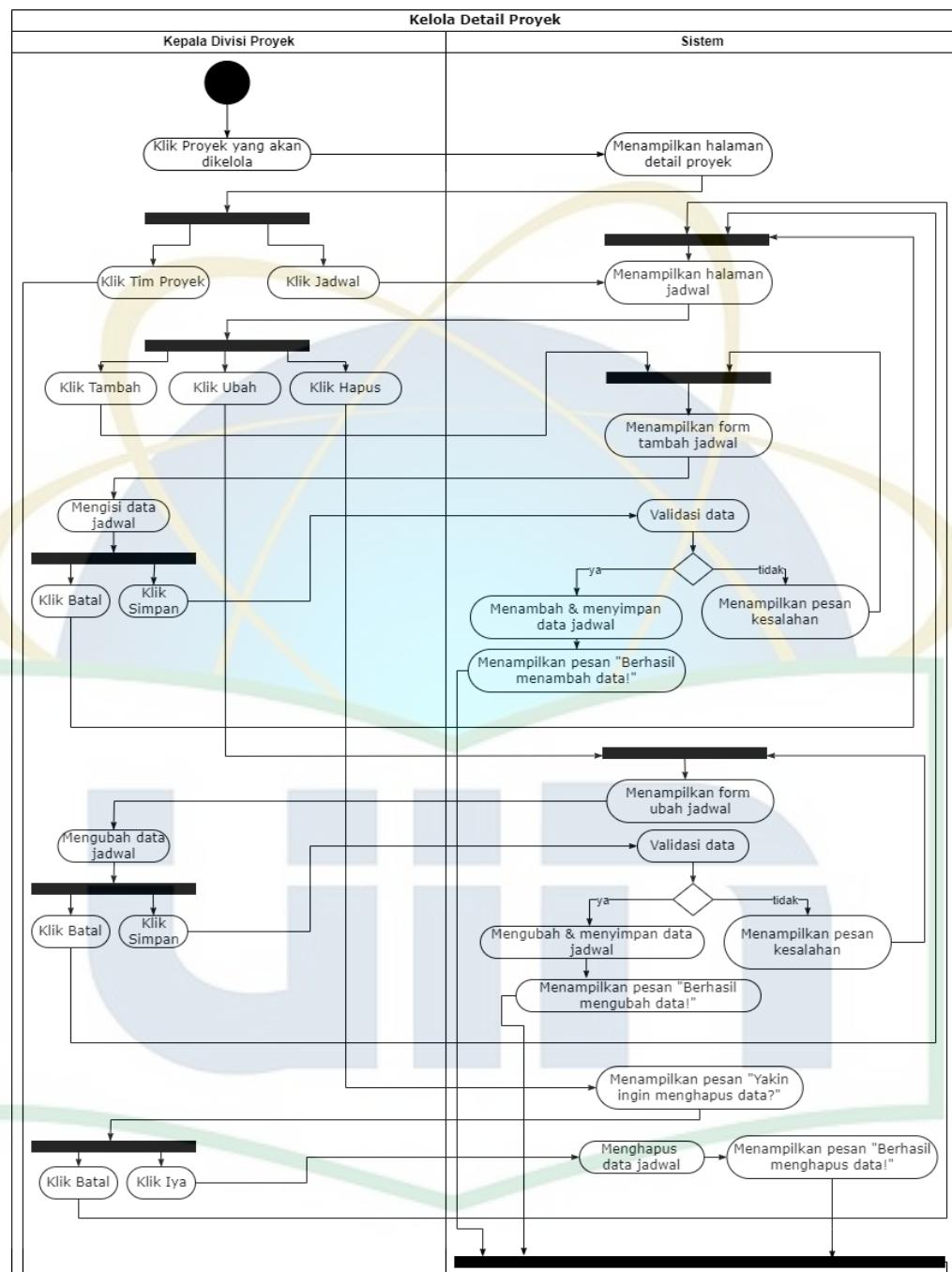
5. Activity Diagram Kelola Data Klien

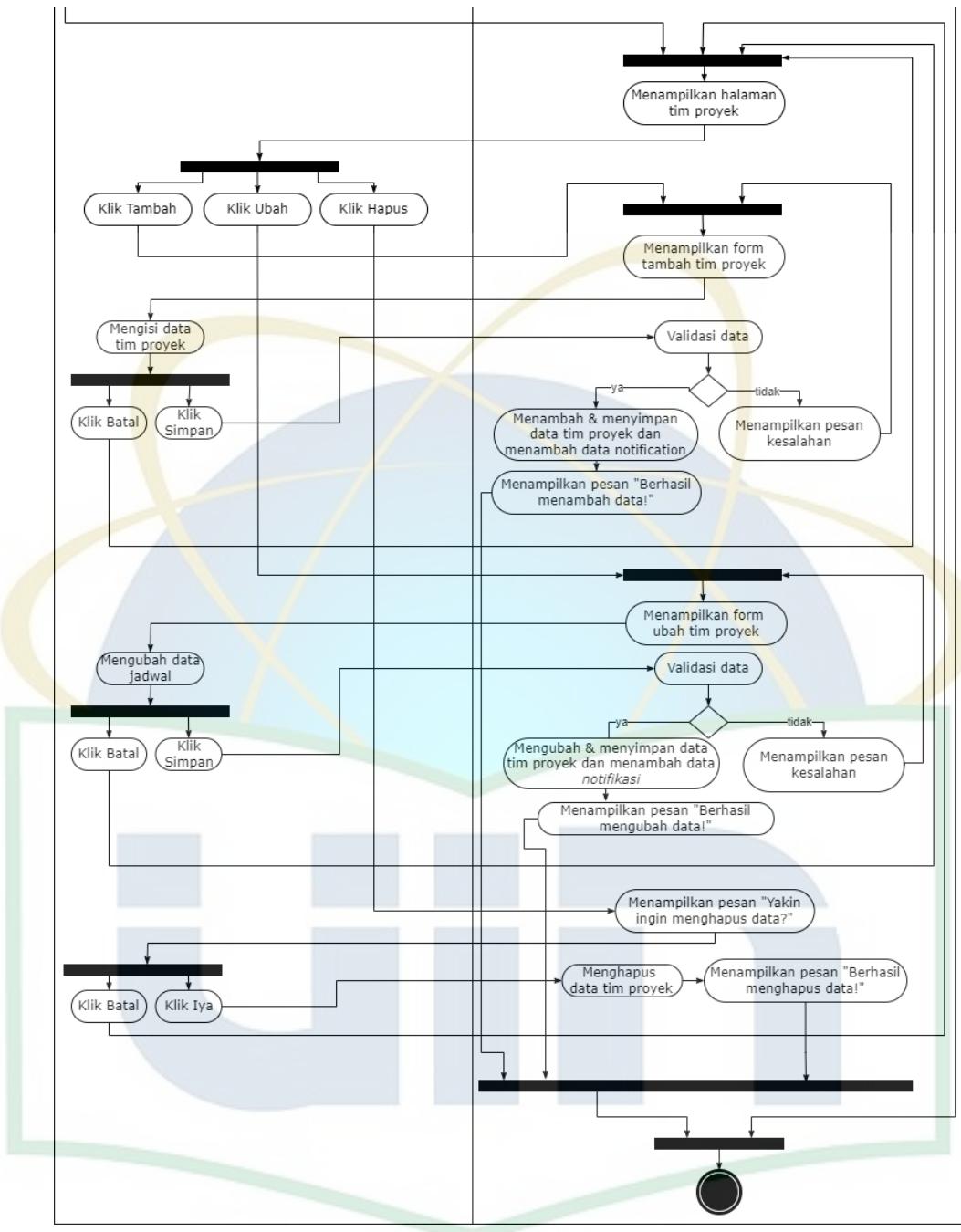


Gambar 4.10 Activity Diagram Kelola Data Klien

Pada aktivitas kelola data klien, admin melakukan pengelolaan data klien dengan tambah, ubah, dan hapus klien.

6. Activity Diagram Kelola Detail Proyek

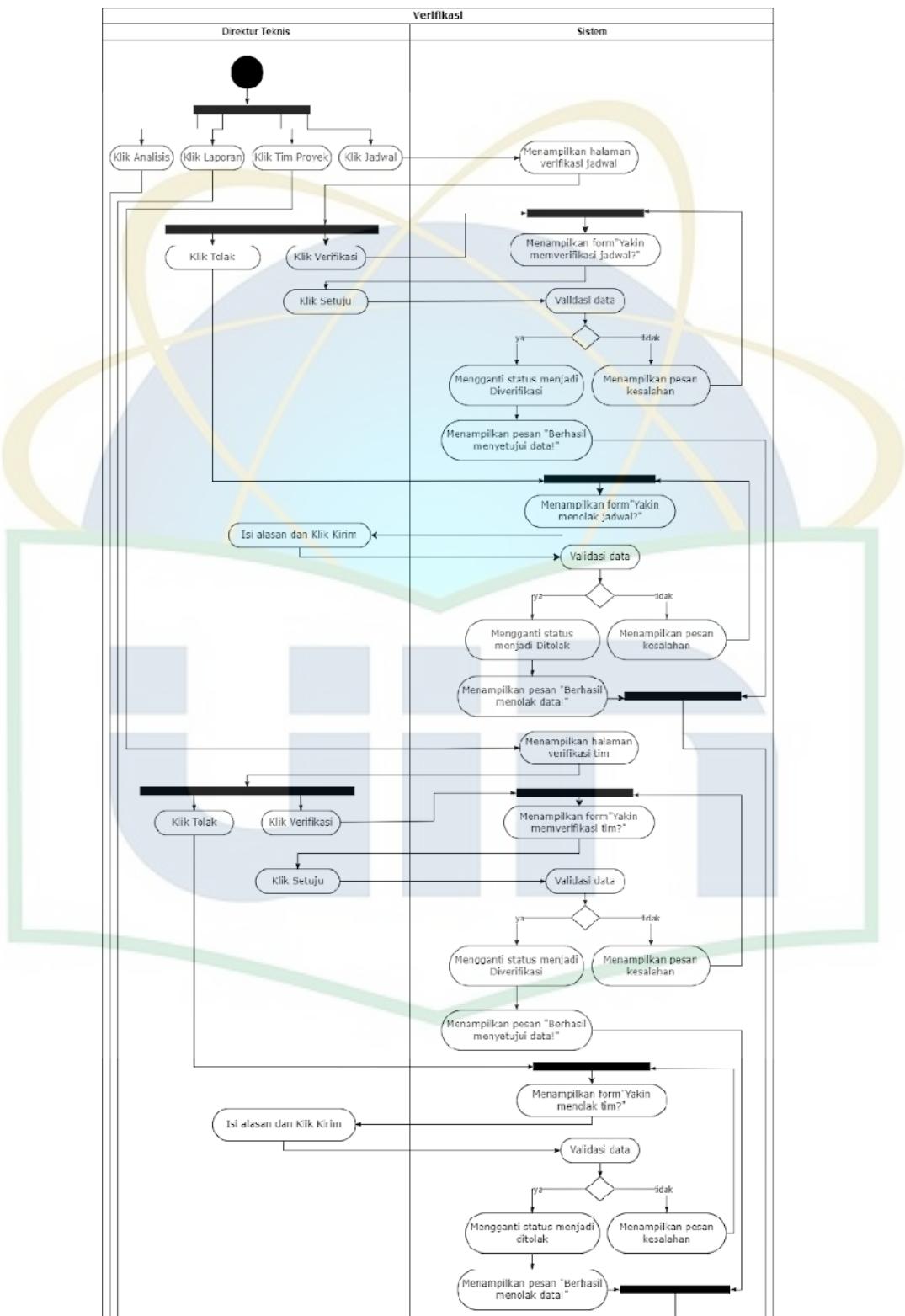


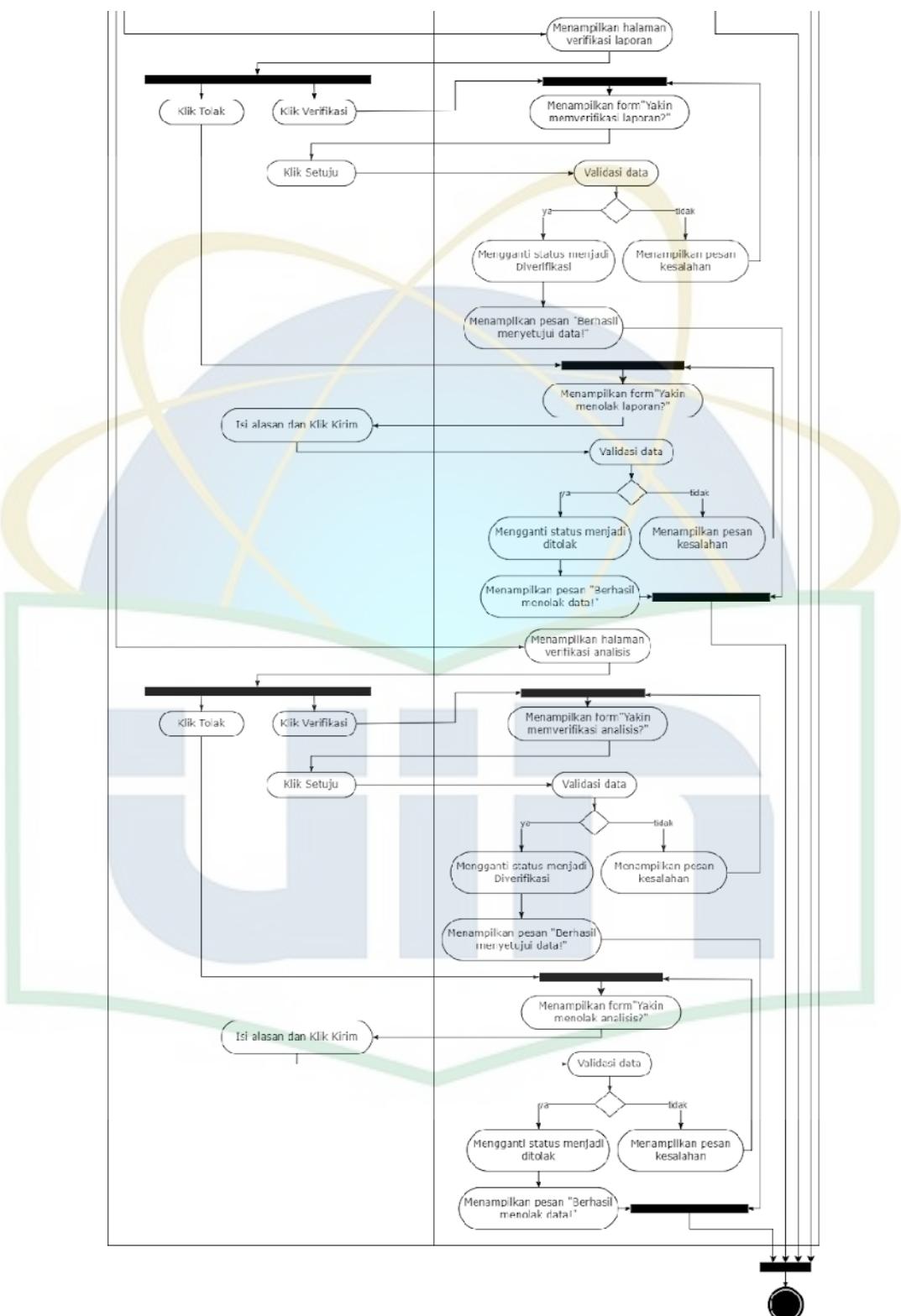


Gambar 4.11 *Activity Diagram Kelola Detail Proyek*

Pada aktivitas kelola detail proyek, kepala divisi proyek melakukan pengelolaan jadwal dan tim proyek dengan tambah, ubah, dan hapus jadwal dan tim proyek.

7. Activity Diagram Verifikasi

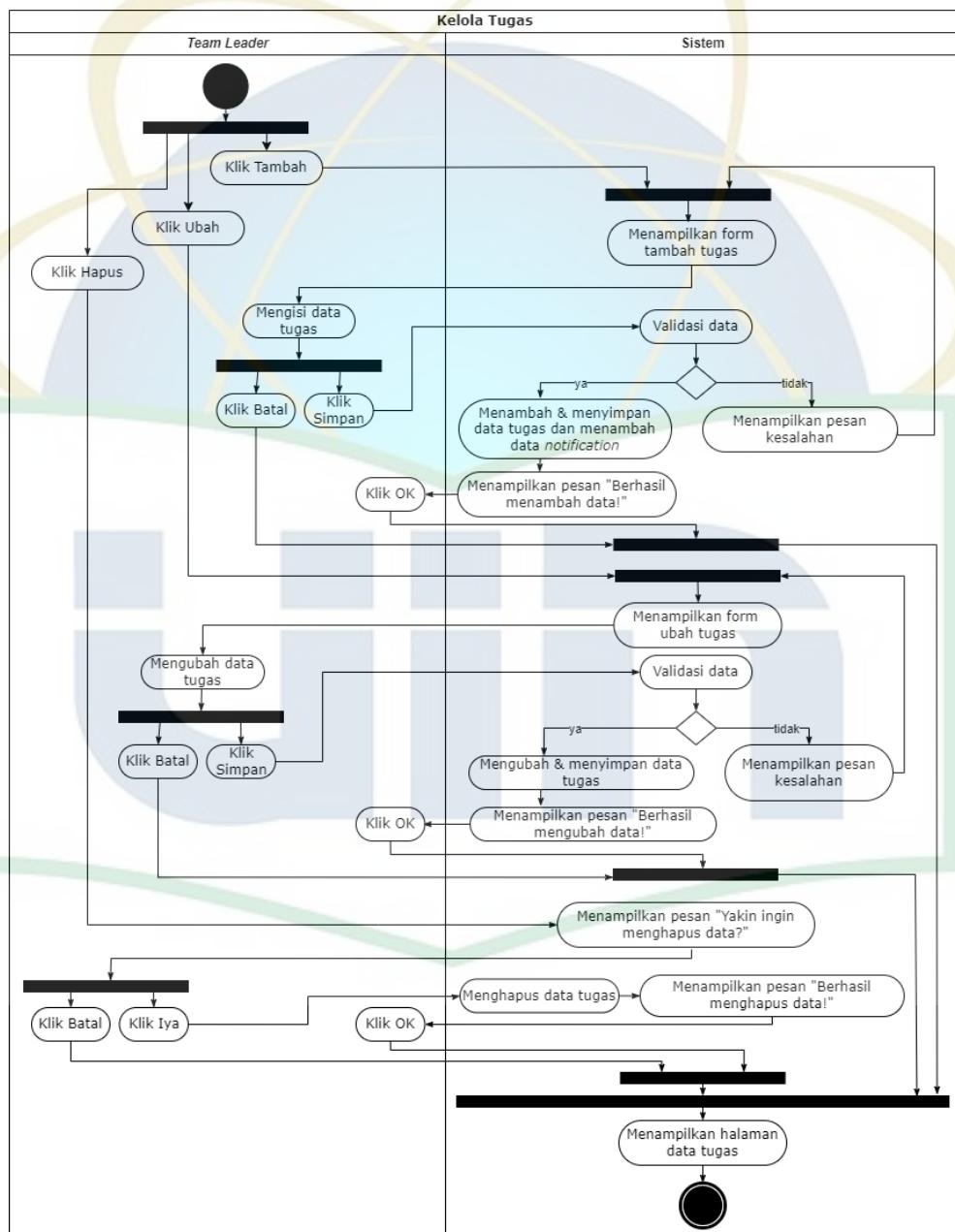




Gambar 4.12 *Activity Diagram* Verifikasi

Pada aktivitas verifikasi, direktur teknis memverifikasi jadwal proyek, tim proyek, laporan, dan analisis keterlambatan proyek dengan klik verifikasi dan jika menolak klik tolak serta mengisi alasan menolak.

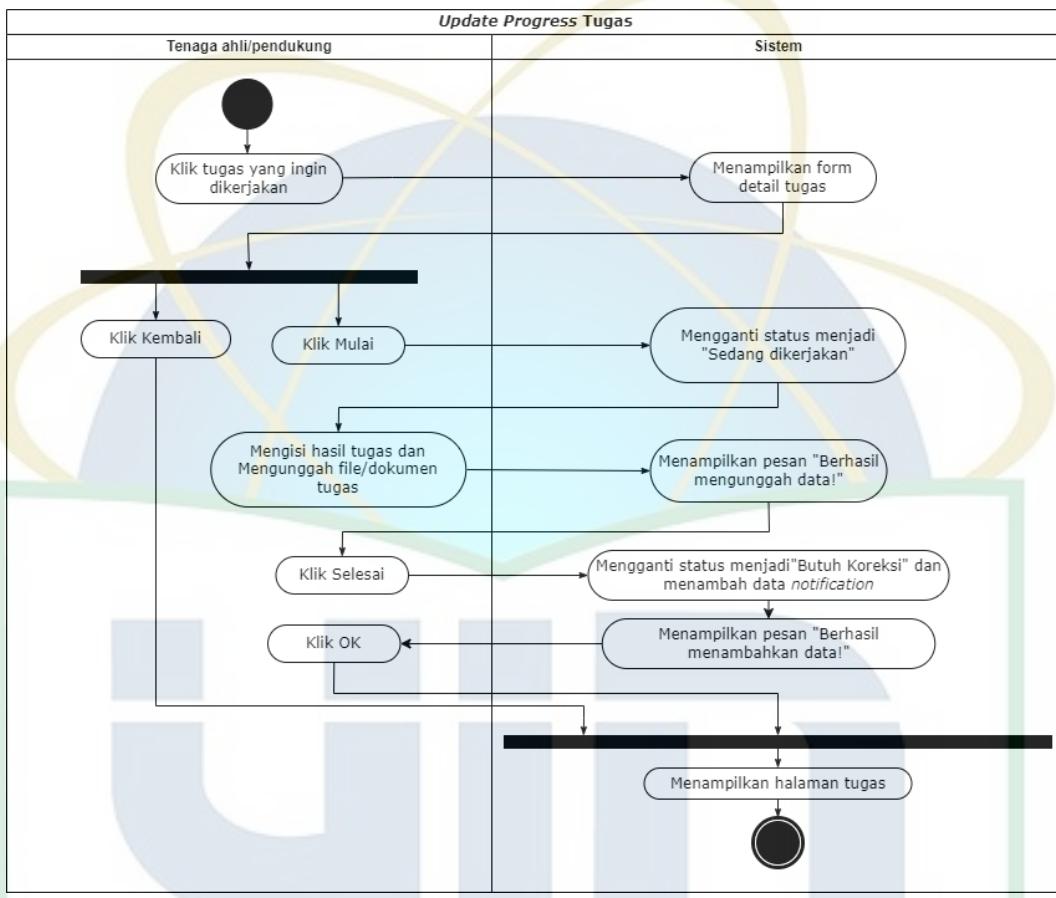
8. Activity Diagram Kelola Tugas



Gambar 4.13 Activity Diagram Kelola Tugas

Pada aktivitas kelola tugas, *team leader* melakukan pengelolaan tugas dengan tambah, ubah, dan hapus tugas.

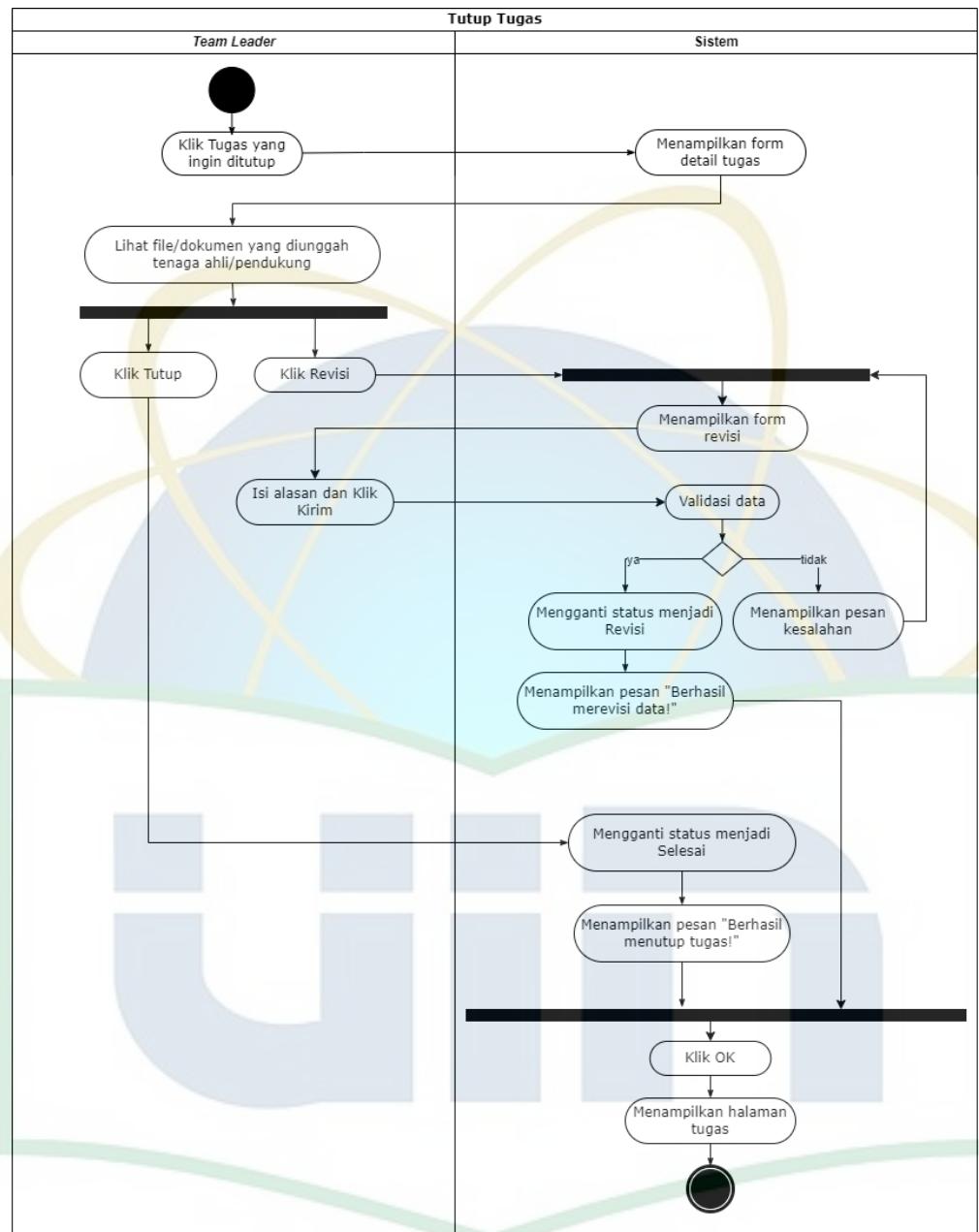
9. Activity Diagram Update Progress Tugas



Gambar 4.14 Activity Diagram Update Progress Tugas

Pada aktivitas *update progress* tugas, tenaga ahli atau pendukung melakukan tugas dan melaporkannya dengan klik mulai untuk menandai sedang mengerjakan tugas, upload *file*/dokumen hasil dari tugas yang diberikan dan klik selesai untuk menandakan tugas butuh koreksi dari *team leader*.

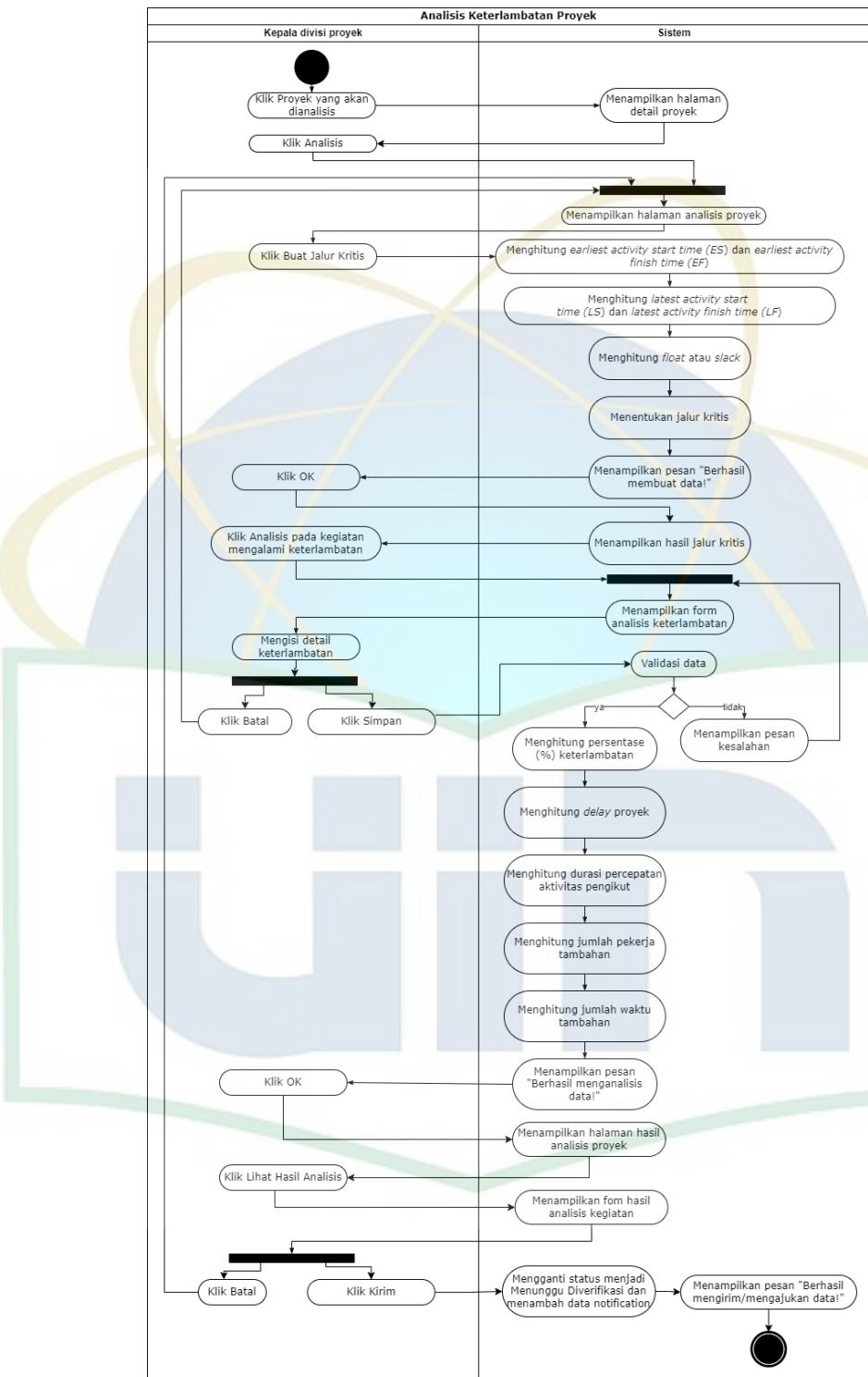
10. Activity Diagram Tutup Tugas



Gambar 4.15 Activity Diagram Tutup Tugas

Pada aktivitas tutup tugas, *team leader* menutup tugas yang telah diselesaikan oleh tenaga ahli/pendukung dengan cara klik tutup dan jika masih terdapat kekurangan klik revisi dan alasan.

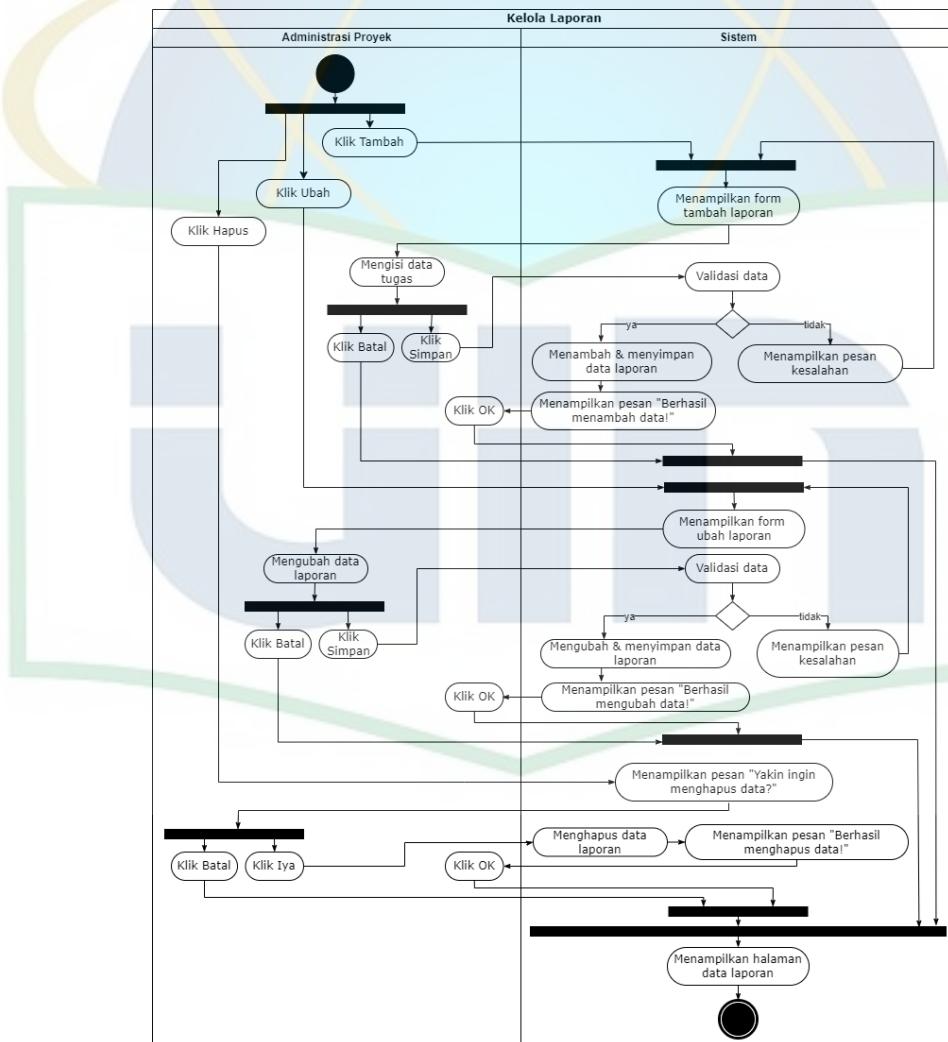
11. Activity Diagram Analisis Keterlambatan Proyek



Gambar 4.16 Activity Diagram Analisis Keterlambatan Proyek

Pada aktivitas keterlambatan proyek, kepala divisi proyek menganalisis keterlambatan proyek dengan menggunakan 2 metode yaitu *critical path method* dan *what if analysis*. Pertama sistem membuat jalur krisis setelah klik Buat Jalur Kritis, setelah mendapatkan hasil jalur kritis, klik analisis pada aktivitas proyek yang terlambat dan sistem akan otomatis menganalisis jumlah pekerja dan waktu tambahan yang diperlukan agar proyek tidak terlambat.

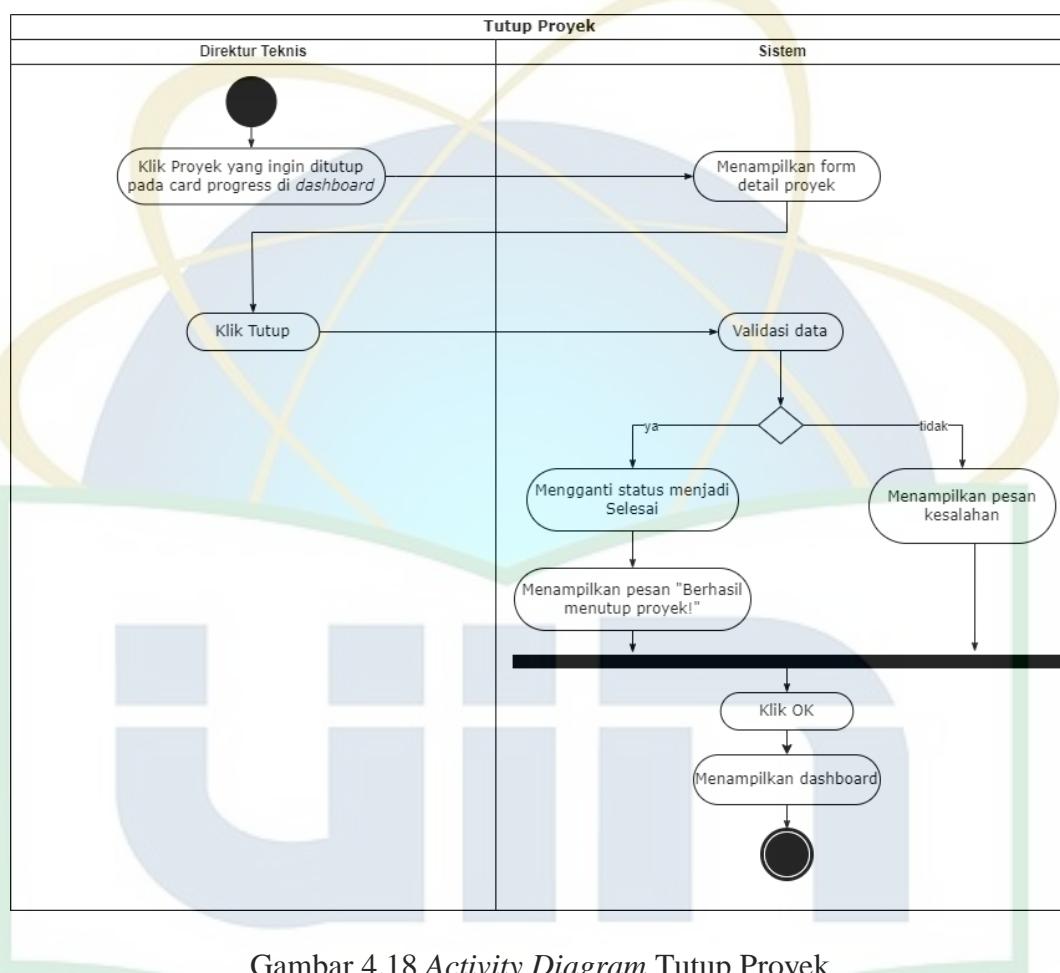
12. Activity Diagram Kelola Laporan



Gambar 4.17 Activity Diagram Kelola Laporan

Pada aktivitas kelola laporan, administrasi proyek melakukan pengelolaan laporan dengan tambah, ubah, dan hapus laporan.

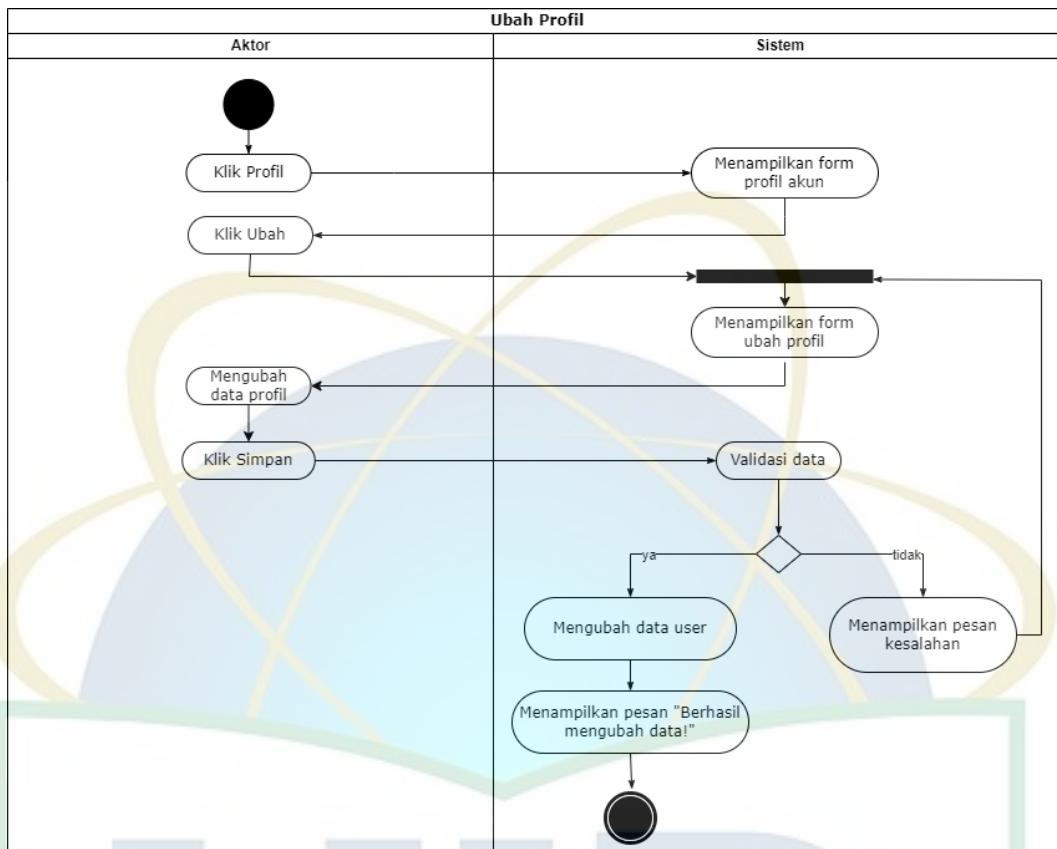
13. Activity Diagram Tutup Proyek



Gambar 4.18 Activity Diagram Tutup Proyek

Pada aktivitas tutup tugas, direktur teknis menutup proyek yang telah selesai dengan cara klik tutup dan jika masih terdapat kekurangan klik revisi dan alasan.

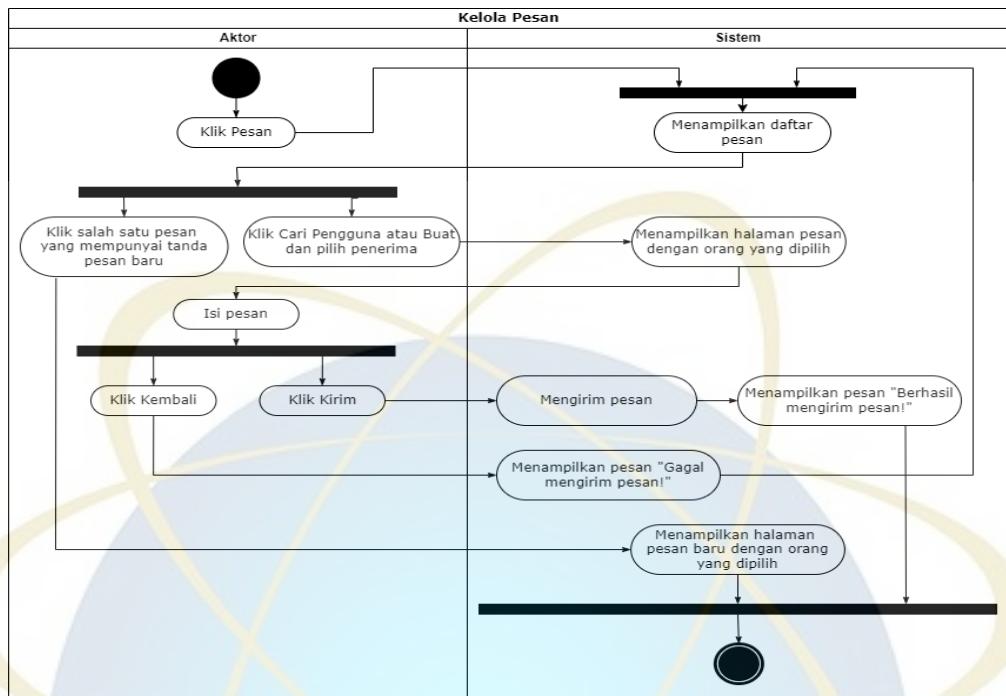
14. Activity Diagram Ubah Profil



Gambar 4.19 Activity Diagram Ubah Profil

Pada aktivitas ubah profil, semua aktor mengubah profil akunnya dengan cara mengisi dengan data baru dan klik simpan.

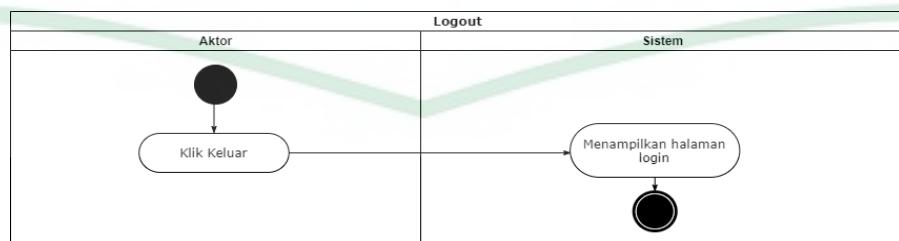
15. Activity Diagram Kelola Pesan



Gambar 4.20 Activity Diagram Kelola Pesan

Pada aktivitas kelola pesan, semua aktor mengirim dan menerima pesan kepada akun yang dituju.

16. Activity Diagram Logout



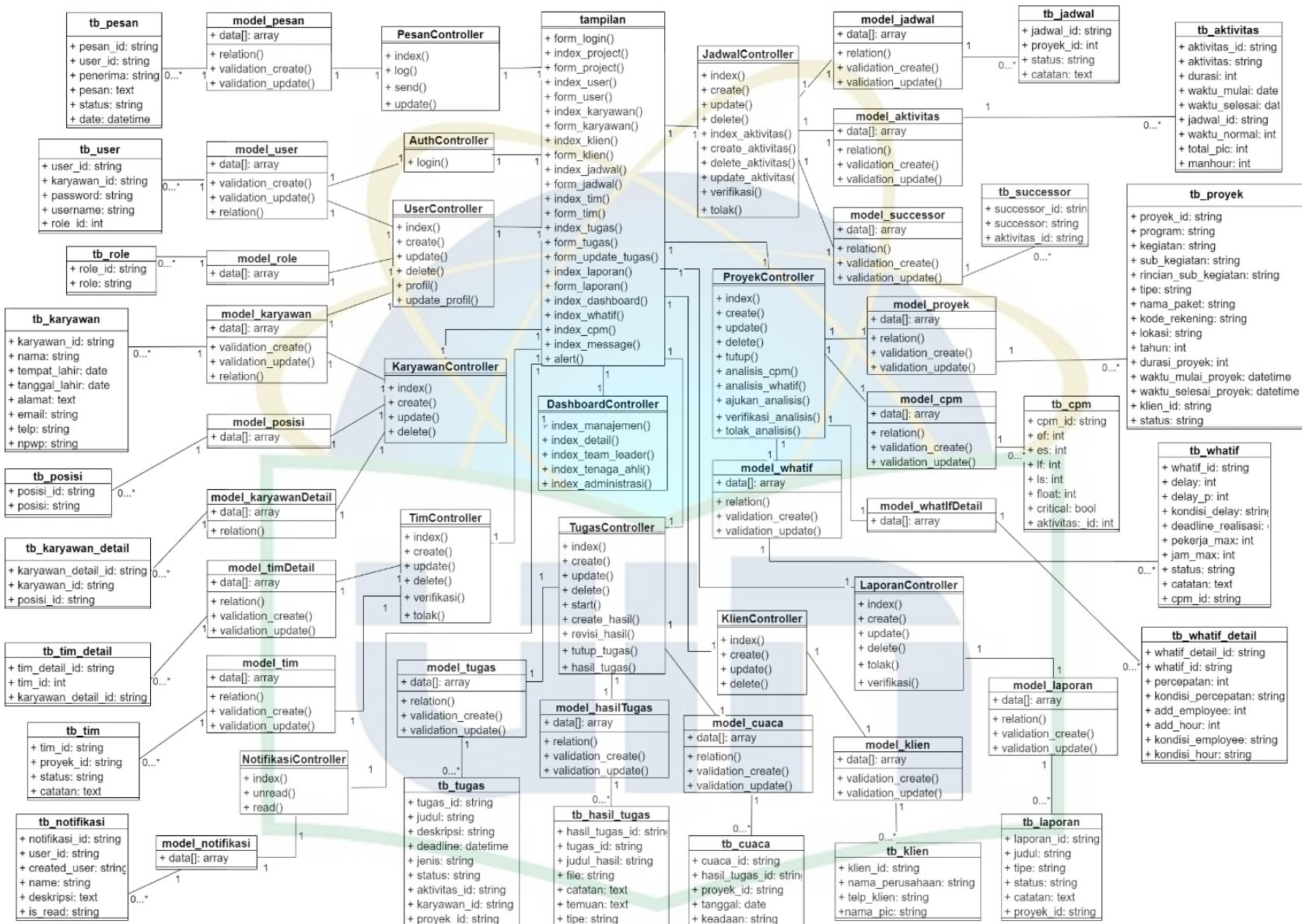
Gambar 4.21 Activity Diagram Logout

Pada aktivitas *logout*, seluruh aktor keluar dari sistem.

4.2.2.3 Class Diagram

Pada *class diagram* digambarkan struktur kelas-kelas yang berisi objek-objek dan hubungan antar objeknya serta metode yang dimiliki objek yang terdapat pada Gambar 4.22.



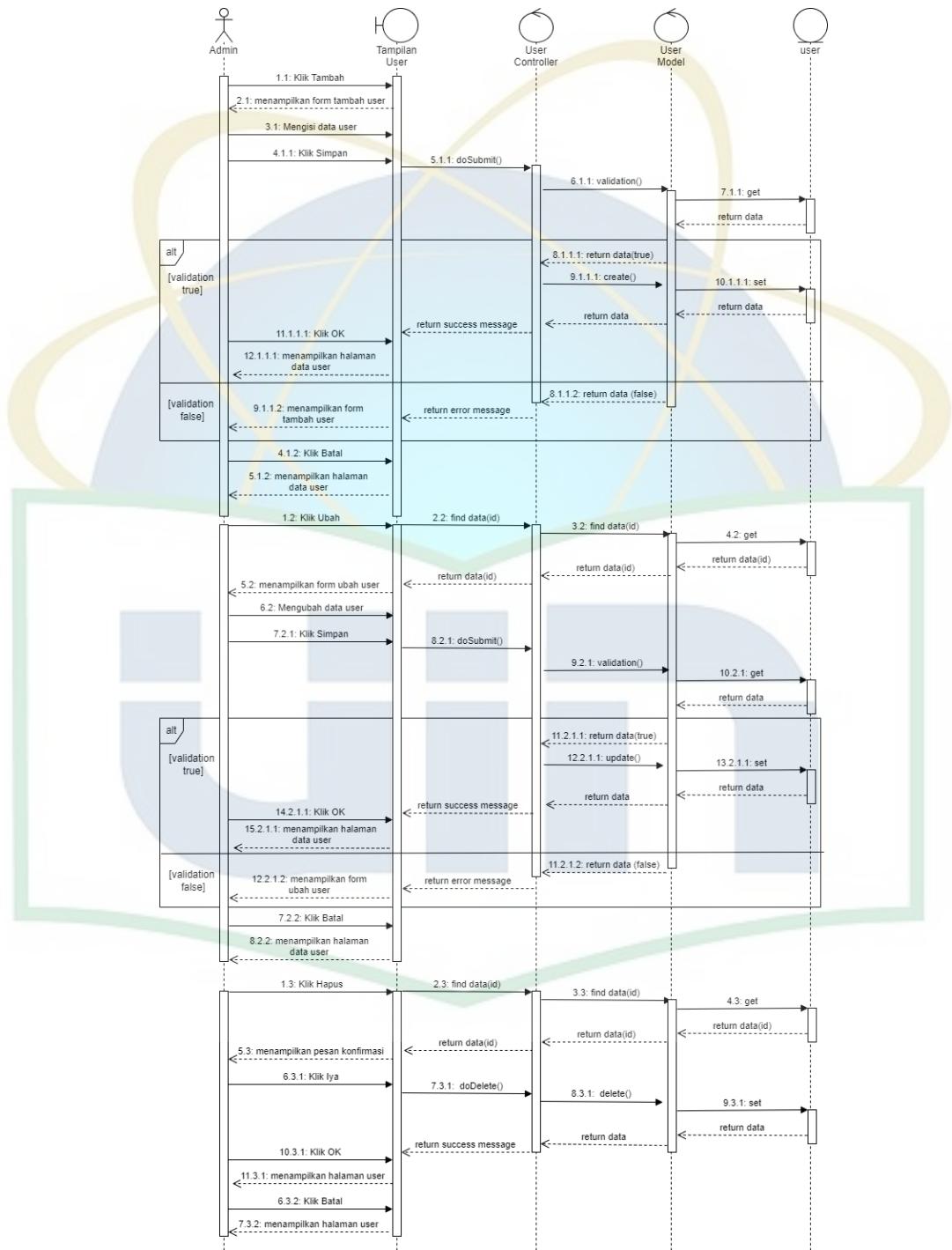


Gambar 4.22 *Class Diagram* Sistem Informasi Manajemen Proyek

4.2.2.4 Sequence Diagram

Pada *Sequence Diagram*, digambarkan interaksi antar objek.

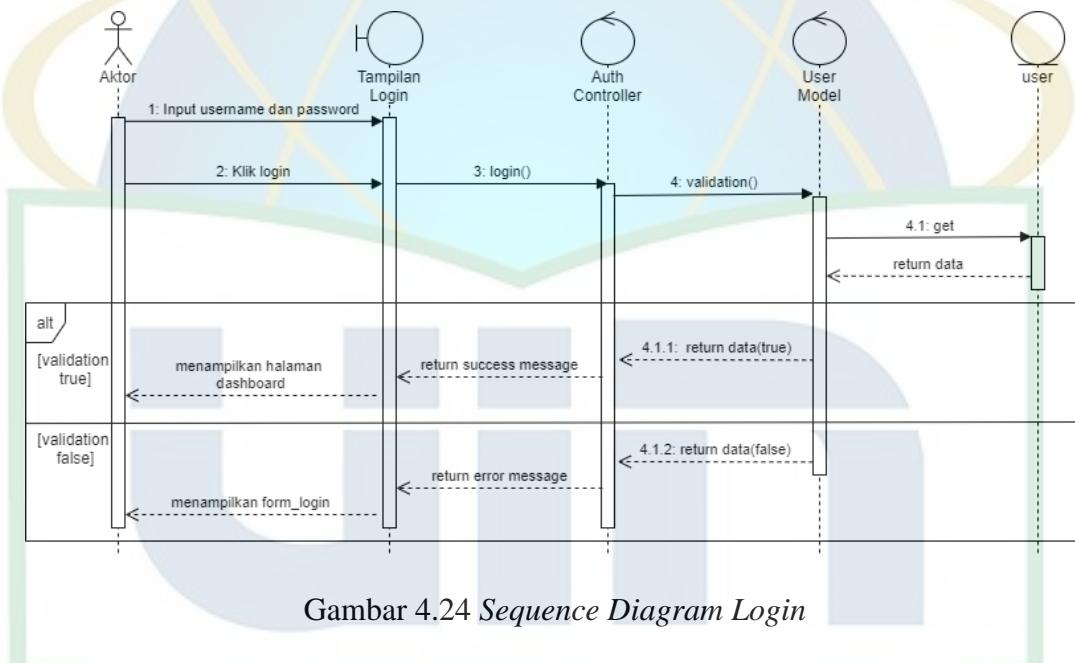
1. Sequence Diagram Kelola Data User



Gambar 4.23 Sequence Diagram Kelola Data User

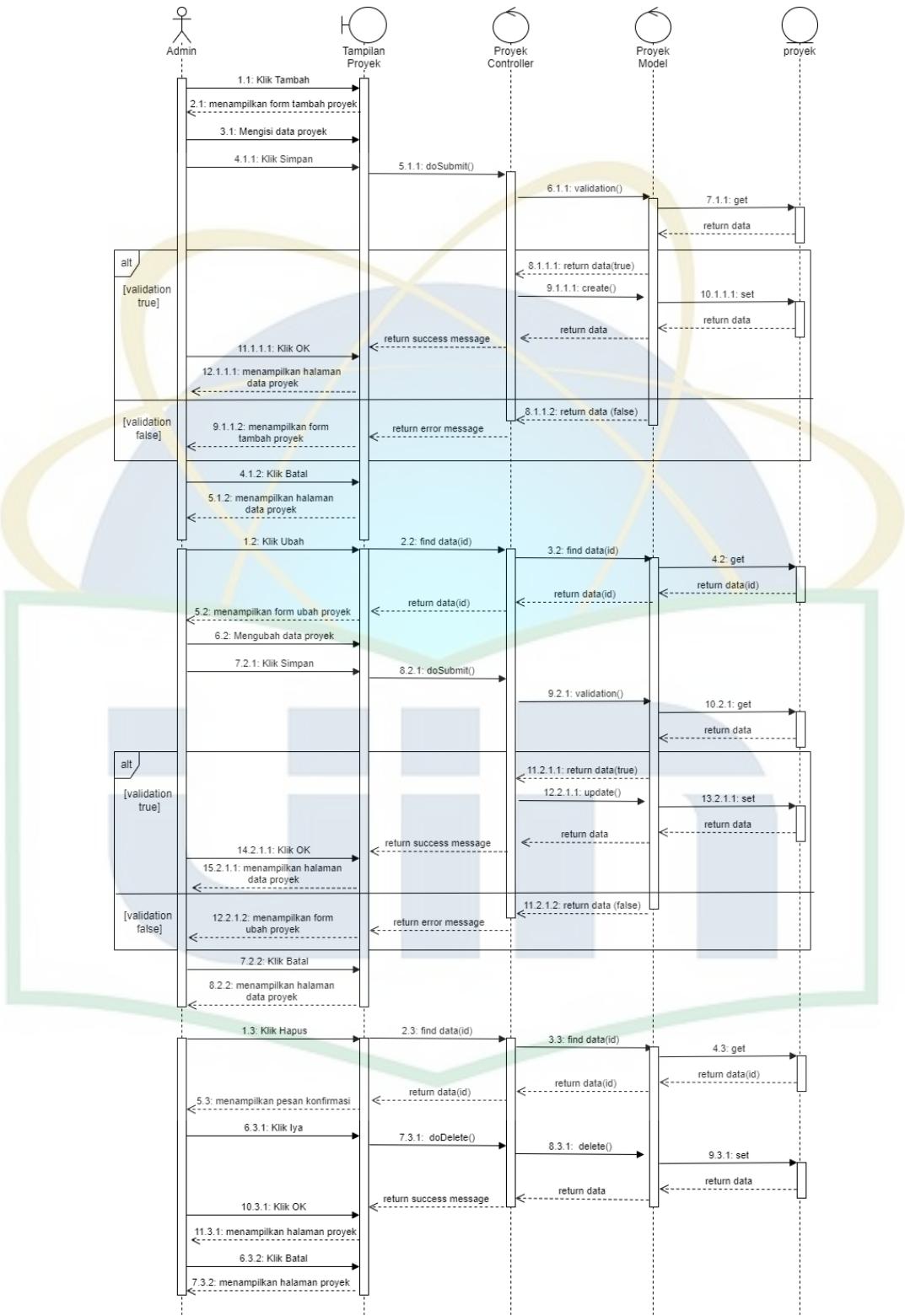
Pada *sequence diagram* Gambar 4.23, admin melakukan pengelolaan data *user* dengan tambah, ubah, dan hapus data-data *user*. Admin dapat klik tambah atau ubah data lalu sistem akan mevalidasi data. Jika sudah sesuai maka sistem menyimpan menambah atau mengubah data dan menampilkan data tersebut. Jika validasi gagal maka akan kembali ke halaman data user. Admin juga dapat menghapus data dengan klik hapus lalu sistem akan menghapus data dari *database*.

2. Sequence Diagram Login



Pada *sequence diagram* Gambar 4.24, seluruh aktor yaitu admin, direktur teknis, kepala divisi proyek, *team leader*, tenaga ahli/pendukung, administrasi proyek mengisi *username* dan *password* lalu sistem akan memvalidasi jika validasi sesuai maka masuk ke *dashboard* tapi jika tidak akan kembali ke halaman *login*.

3. Sequence Diagram Kelola Data Proyek

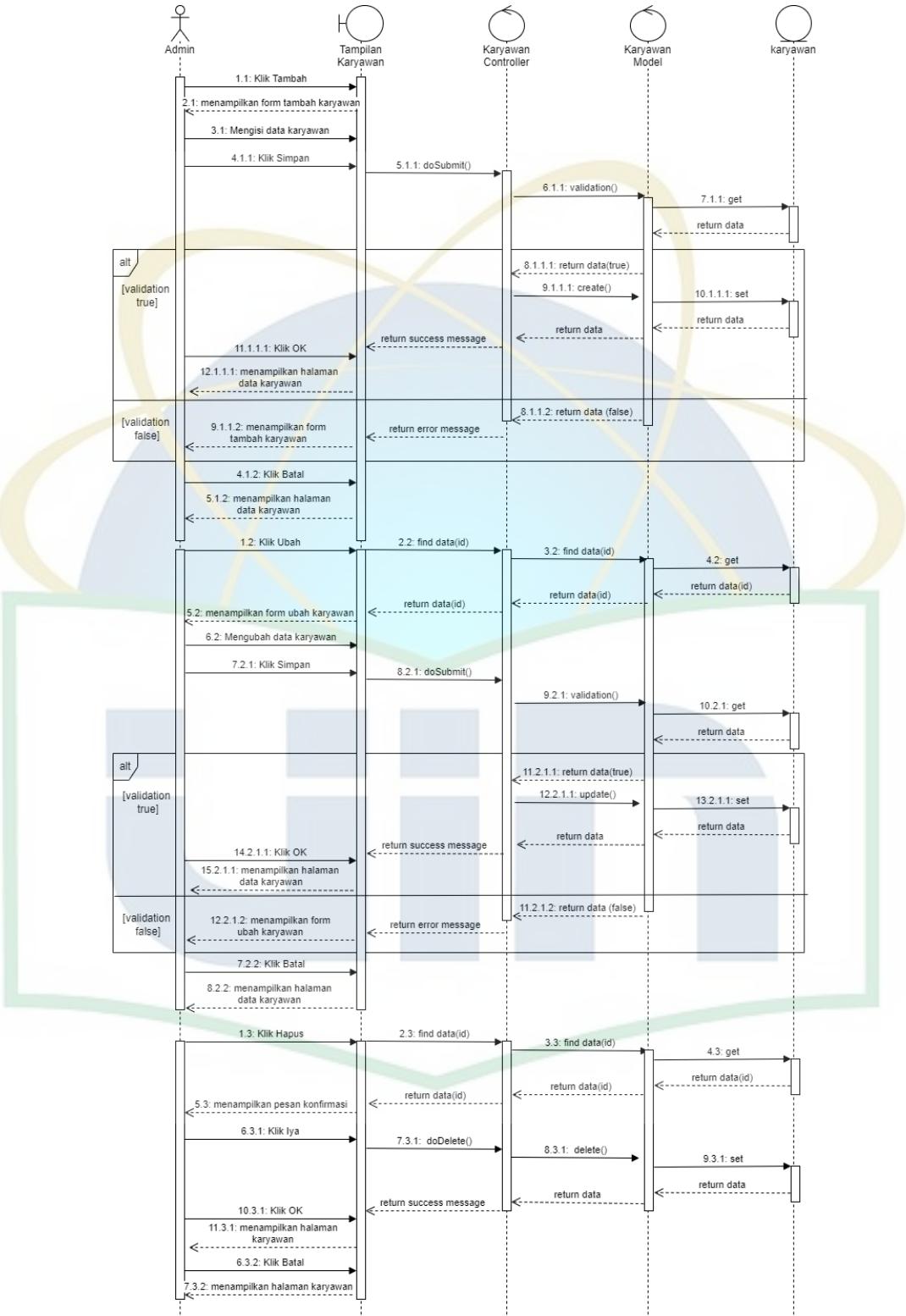


Gambar 4.25 Sequence Diagram Kelola Data Proyek

Pada *sequence diagram* Gambar 4.25, admin melakukan pengelolaan data proyek dengan tambah, ubah, dan hapus data-data proyek. Admin dapat klik tambah atau ubah data lalu sistem akan mevalidasi data. Jika sudah sesuai maka sistem menyimpan menambah atau mengubah data dan menampilkan data tersebut. Jika validasi gagal maka akan kembali ke halaman data proyek. Admin juga dapat menghapus data dengan klik hapus lalu sistem akan menghapus data dari *database*.



4. Sequence Diagram Kelola Data Karyawan

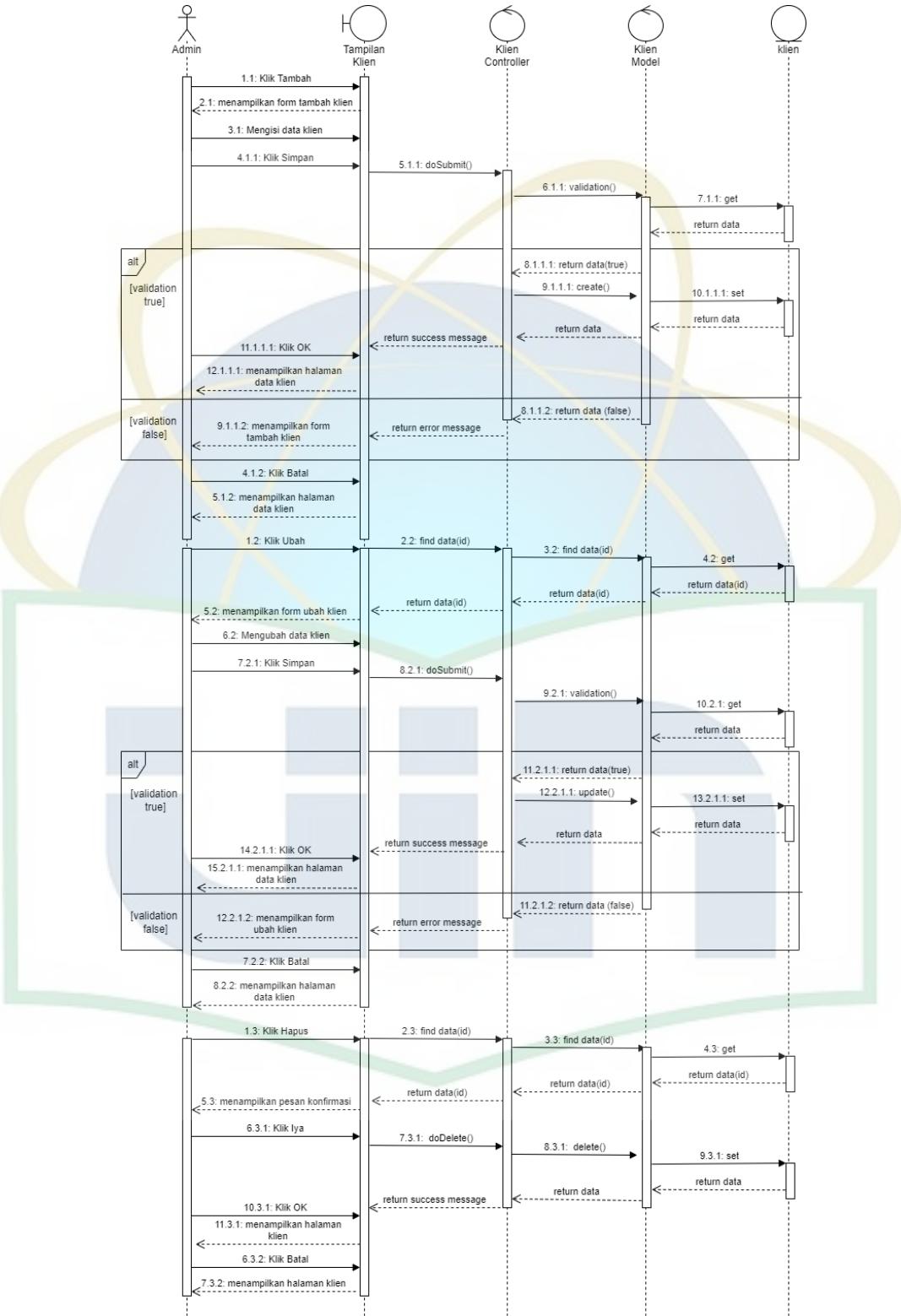


Gambar 4.26 Sequence Diagram Kelola Data Karyawan

Pada *sequence diagram* Gambar 4.26, admin melakukan pengelolaan data karyawan dengan tambah, ubah, dan hapus data-data karyawan. Admin dapat klik tambah atau ubah data lalu sistem akan mevalidasi data. Jika sudah sesuai maka sistem menyimpan menambah atau mengubah data dan menampilkan data tersebut. Jika validasi gagal maka akan kembali ke halaman data karyawan. Admin juga dapat menghapus data dengan klik hapus lalu sistem akan menghapus data dari *database*.



5. Sequence Diagram Kelola Data Klien

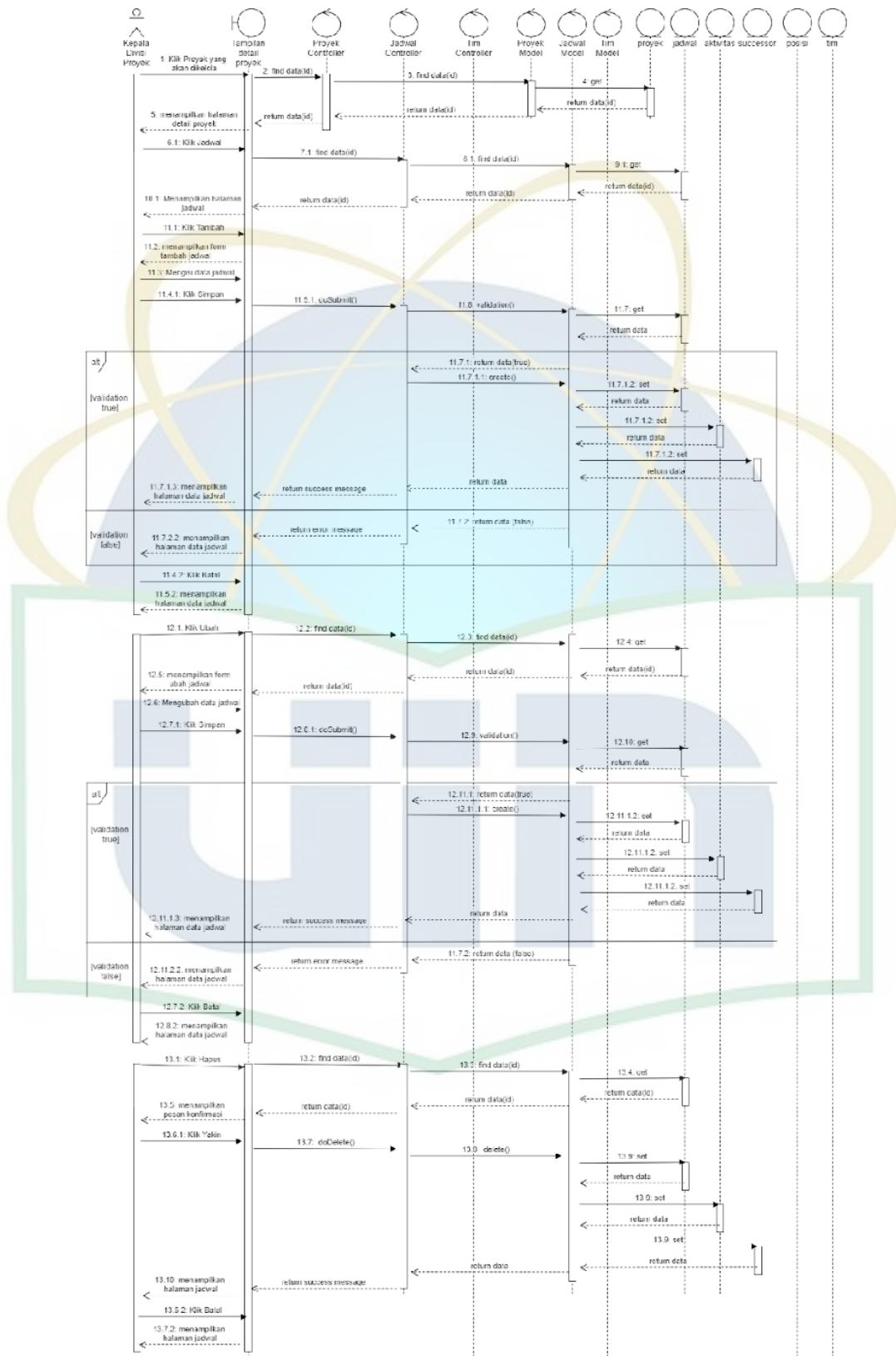


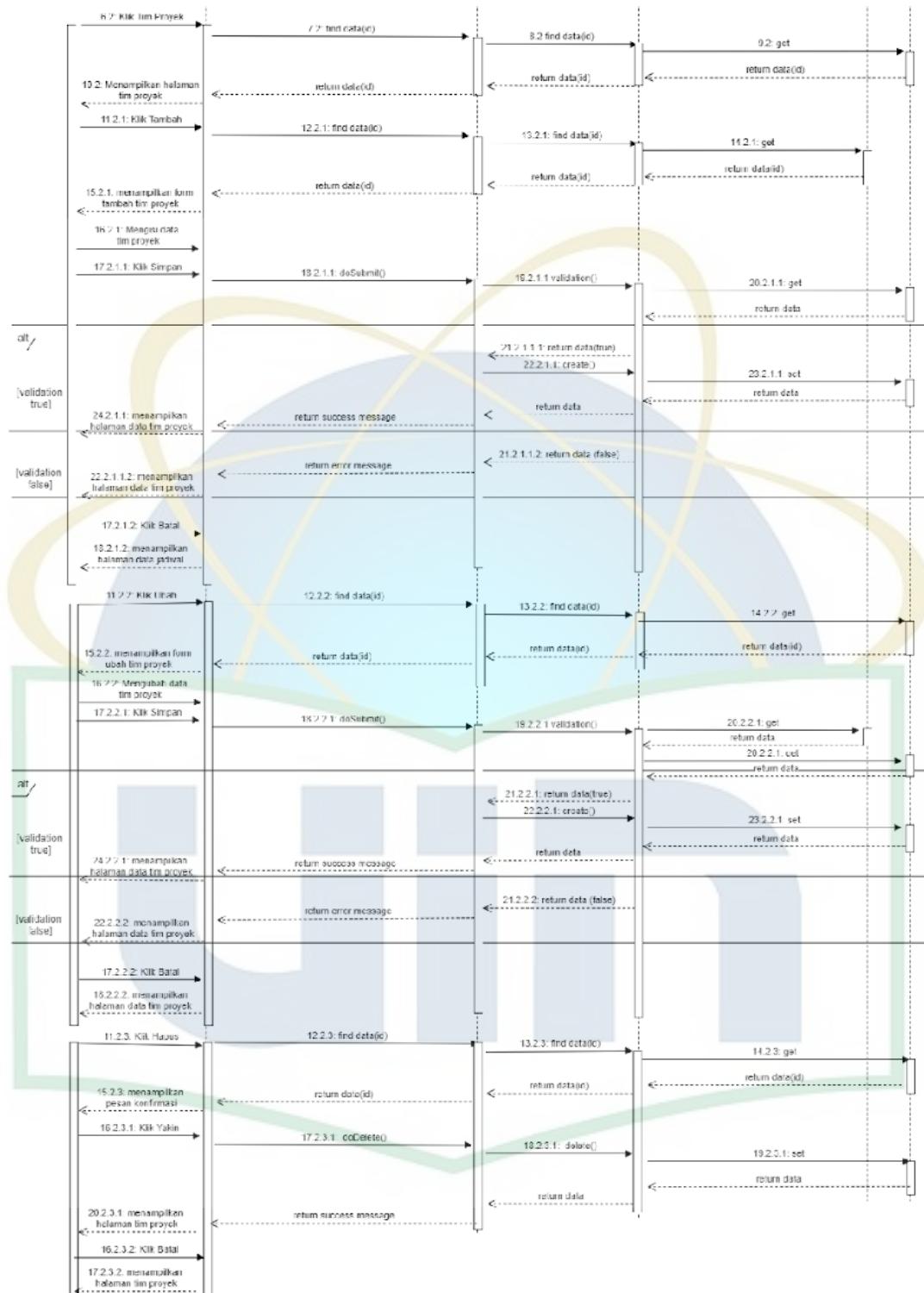
Gambar 4.27 Sequence Diagram Kelola Data Klien

Pada *sequence diagram* Gambar 4.27, admin melakukan pengelolaan data klien dengan tambah, ubah, dan hapus data-data klien. Admin dapat klik tambah atau ubah data lalu sistem akan mevalidasi data. Jika sudah sesuai maka sistem menyimpan menambah atau mengubah data dan menampilkan data tersebut. Jika validasi gagal maka akan kembali ke halaman data klien. Admin juga dapat menghapus data dengan klik hapus lalu sistem akan menghapus data dari *database*.



6. Sequence Diagram Kelola Detail Proyek



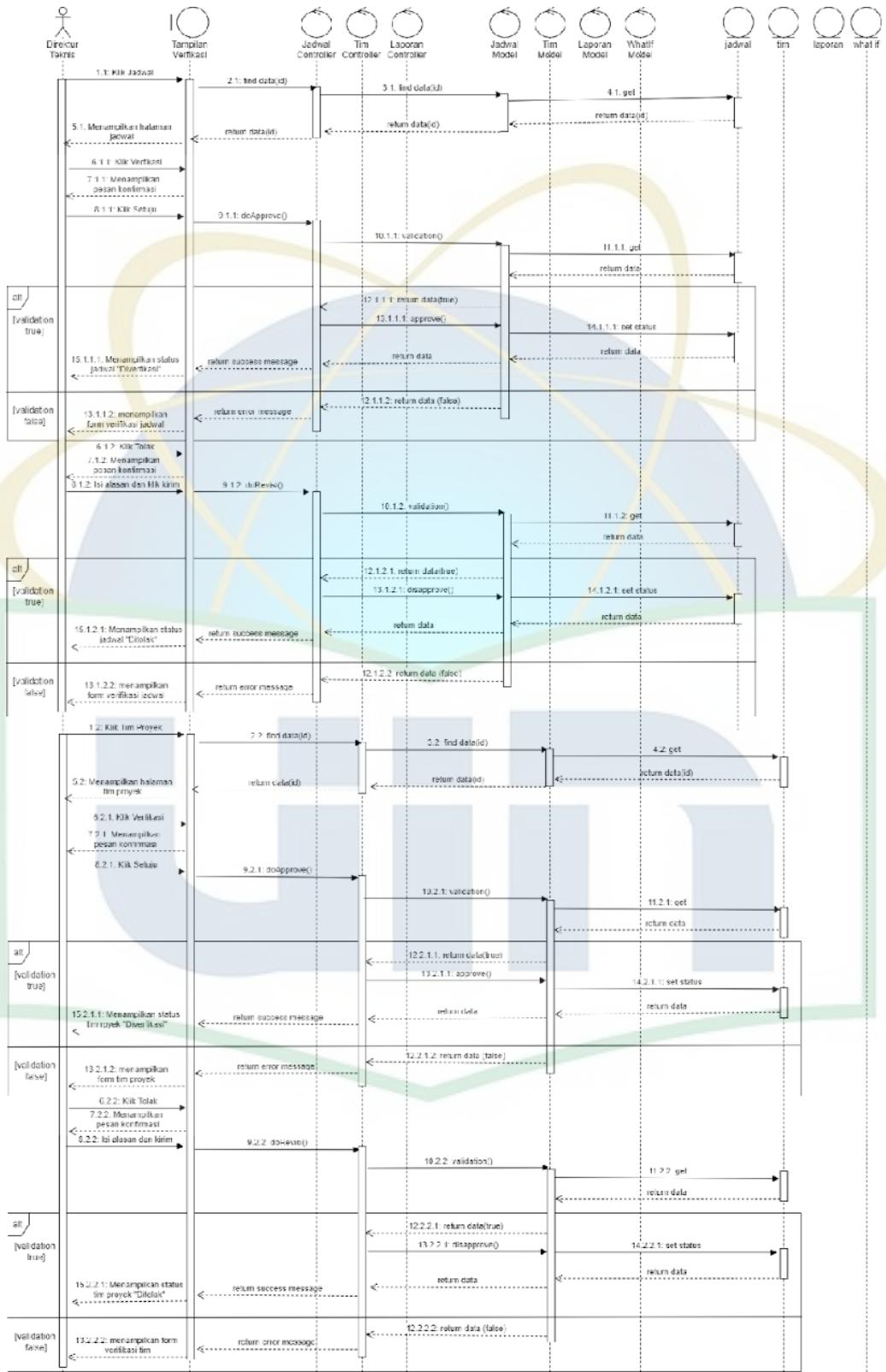


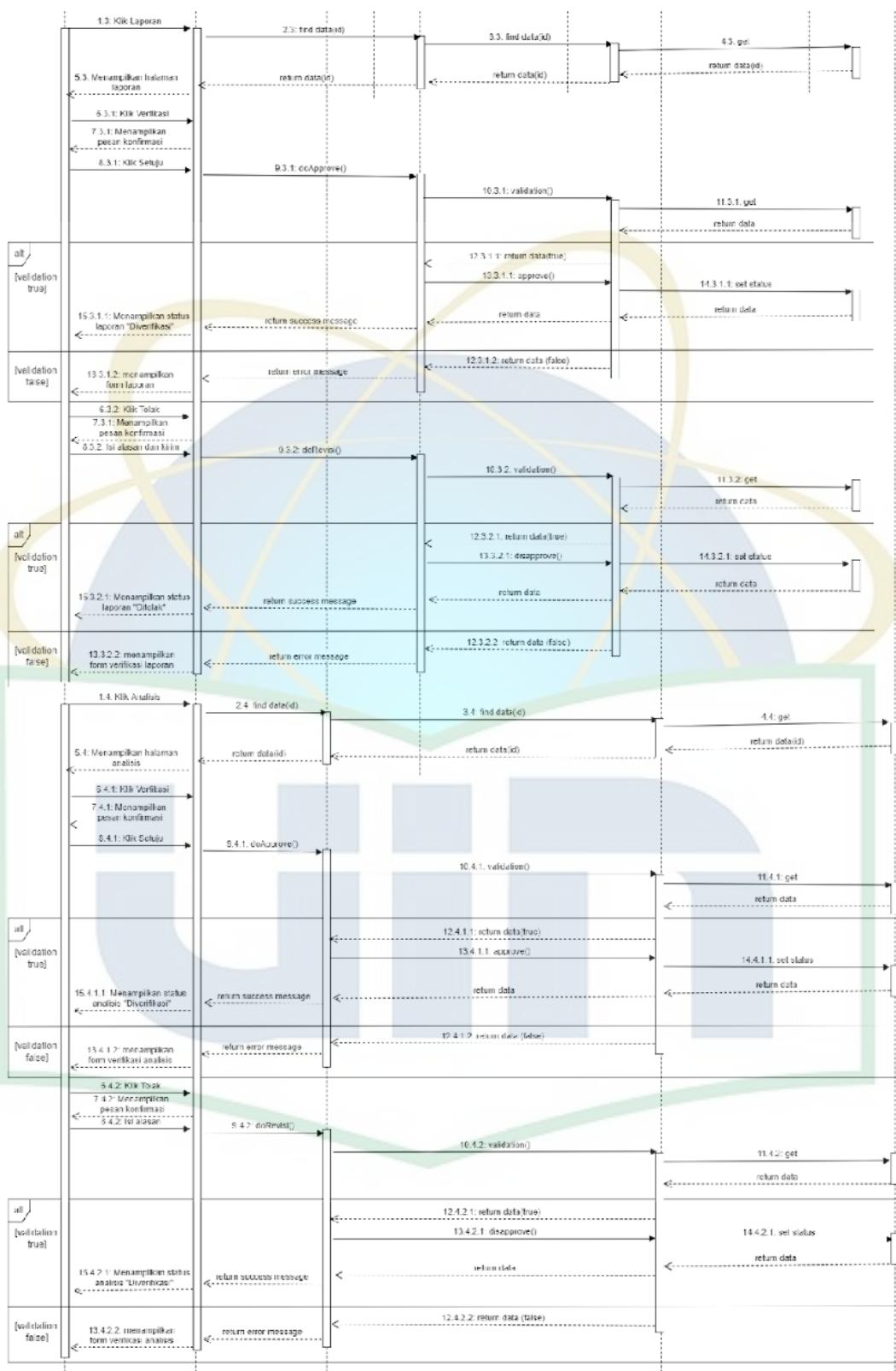
Gambar 4.28 Sequence Diagram Kelola Detail Proyek

Pada *sequence diagram* gamabr 4.28, kepala divisi proyek melakukan pengelolaan detail proyek dengan tambah, ubah, dan hapus data-data jadwal dan tim proyek. Kepala divisi dapat klik tambah atau ubah data lalu sistem akan mevalidasi data. Jika sudah sesuai maka sistem menyimpan menambah atau mengubah data dan menampilkan data tersebut. Jika validasi gagal maka akan kembali ke halaman data proyek. Admin juga dapat menghapus data dengan klik hapus lalu sistem akan menghapus data dari *database*.



7. Sequence Diagram Verifikasi

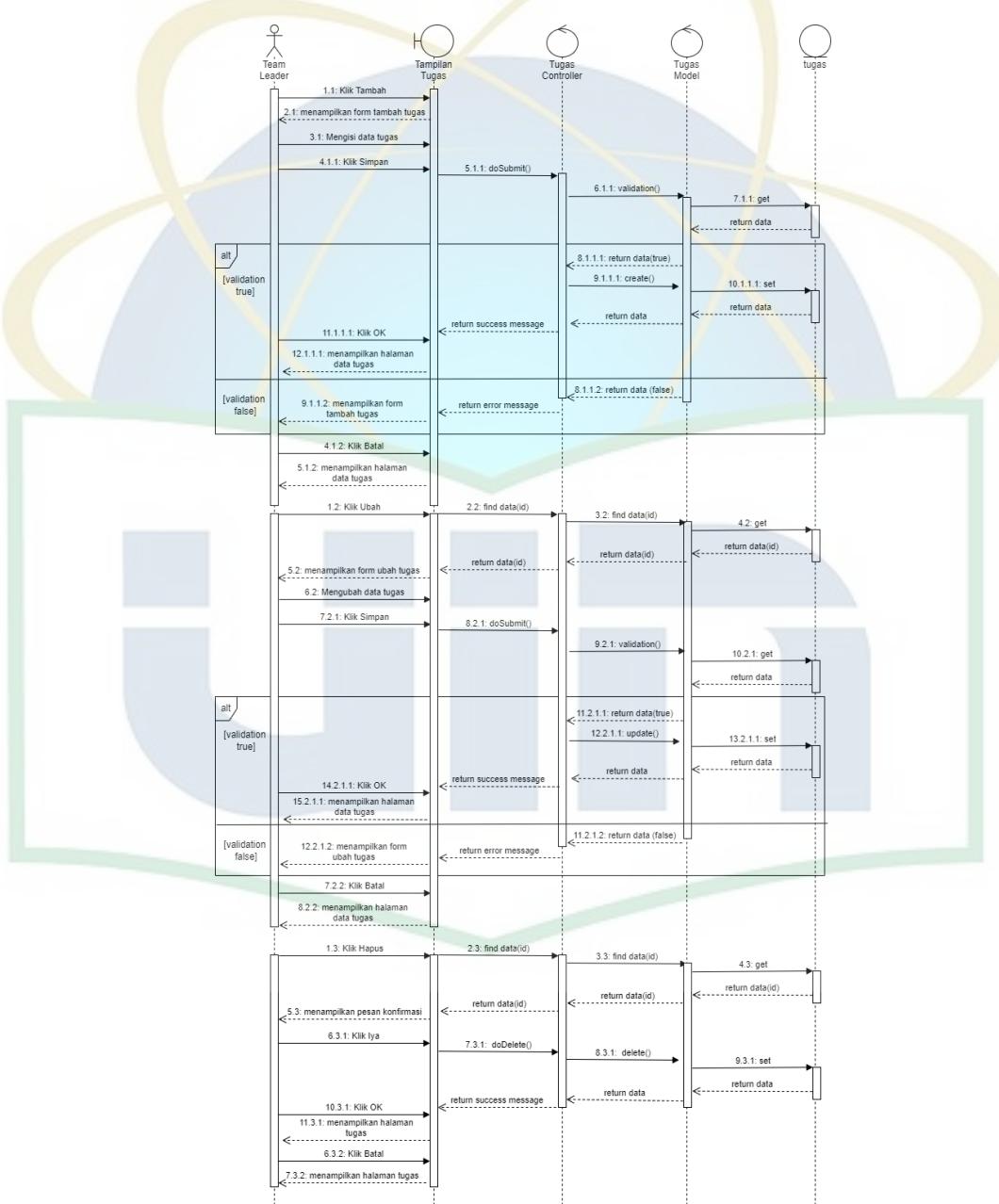




Gambar 4.29 Sequence Diagram Verifikasi

Pada *sequence diagram* Gambar 4.29, direktur teknis memverifikasi jadwal proyek, tim proyek, laporan, dan analisis keterlambatan proyek dengan klik verifikasi dan jika menolak klik tolak serta mengisi alasan menolak.

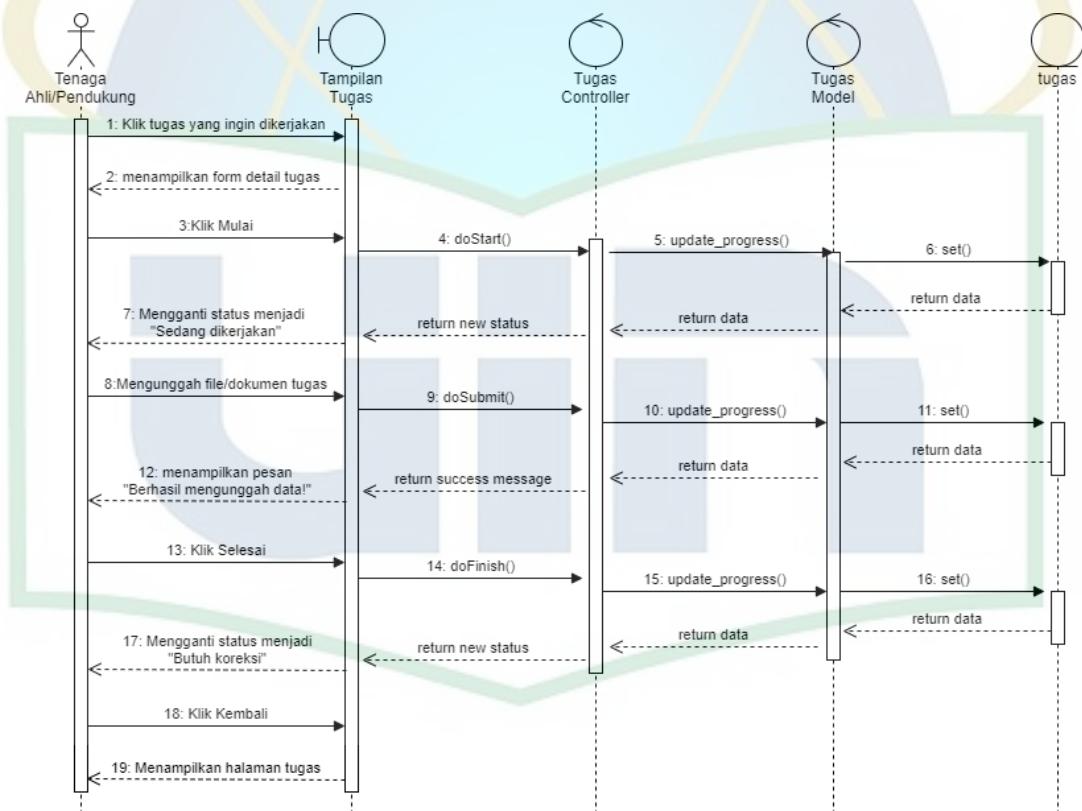
8. Sequence Diagram Kelola Tugas



Gambar 4.30 Sequence Diagram Kelola Tugas

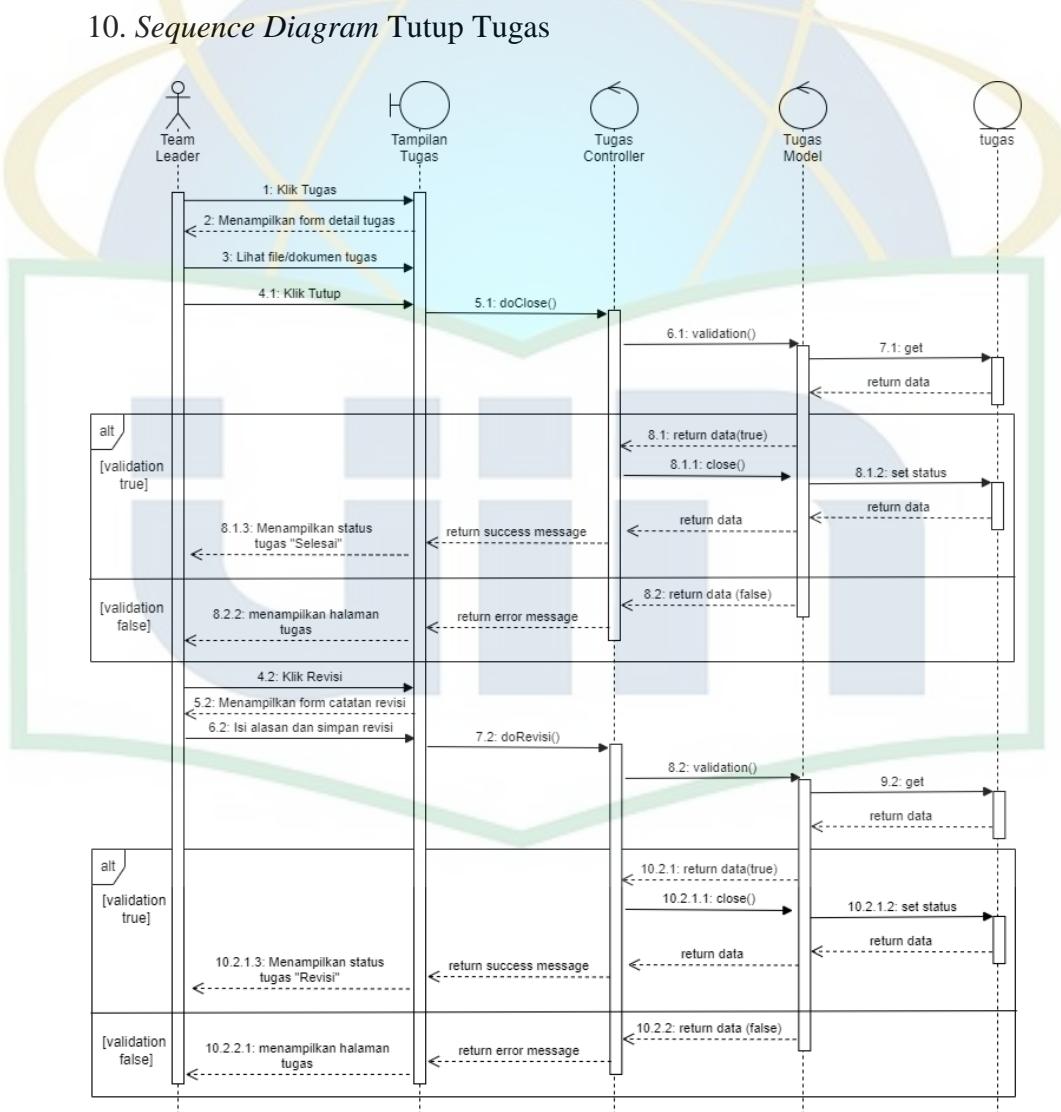
Pada *sequence diagram* 4.30, *team leader* melakukan pengelolaan data tugas dengan tambah, ubah, dan hapus data-data tugas. *Team leader* dapat klik tambah atau ubah data lalu sistem akan mevalidasi data. Jika sudah sesuai maka sistem menyimpan menambah atau mengubah data dan menampilkan data tersebut. Jika validasi gagal maka akan kembali ke halaman data tugas. *Team leader* juga dapat menghapus data dengan klik hapus lalu sistem akan menghapus data dari *database*.

9. Sequence Diagram Update Progress Tugas



Gambar 4.31 Sequence Diagram Update Progress Tugas

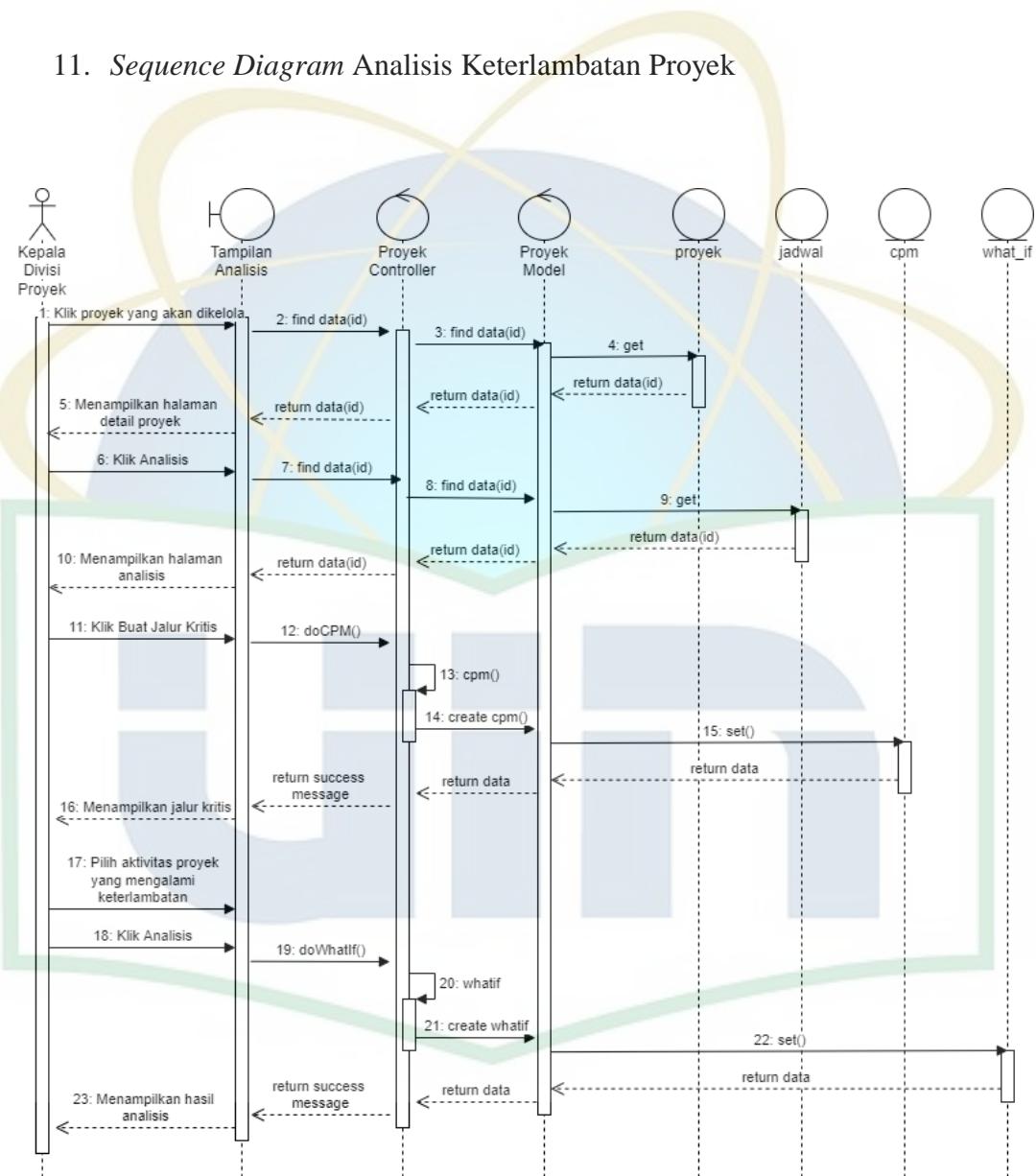
Pada *sequence diagram* Gamabr 4.31, tenaga ahli atau pendukung melakukan tugas dan melaporkannya dengan klik mulai dan sistem akan mengganti status menjadi “Sedang dikerjakan”, lalu *upload file/dokumen hasil dari tugas yang diberikan* dan sistem akan menyimpan file ke *database* untuk dicek oleh *team leader*. Lalu klik selesai dan sistem akan mengganti status “Butuh Koreksi” agar dapat dikoreksi oleh *team leader*.



Gambar 4.32 *Sequence Diagram Tutup Tugas*

Pada *sequence diagram* Gambar 4.32, *team leader* menutup tugas yang telah diselesaikan oleh tenaga ahli/pendukung dengan cara klik tutup dan jika masih terdapat kekurangan klik revisi dan alasan.

11. Sequence Diagram Analisis Keterlambatan Proyek

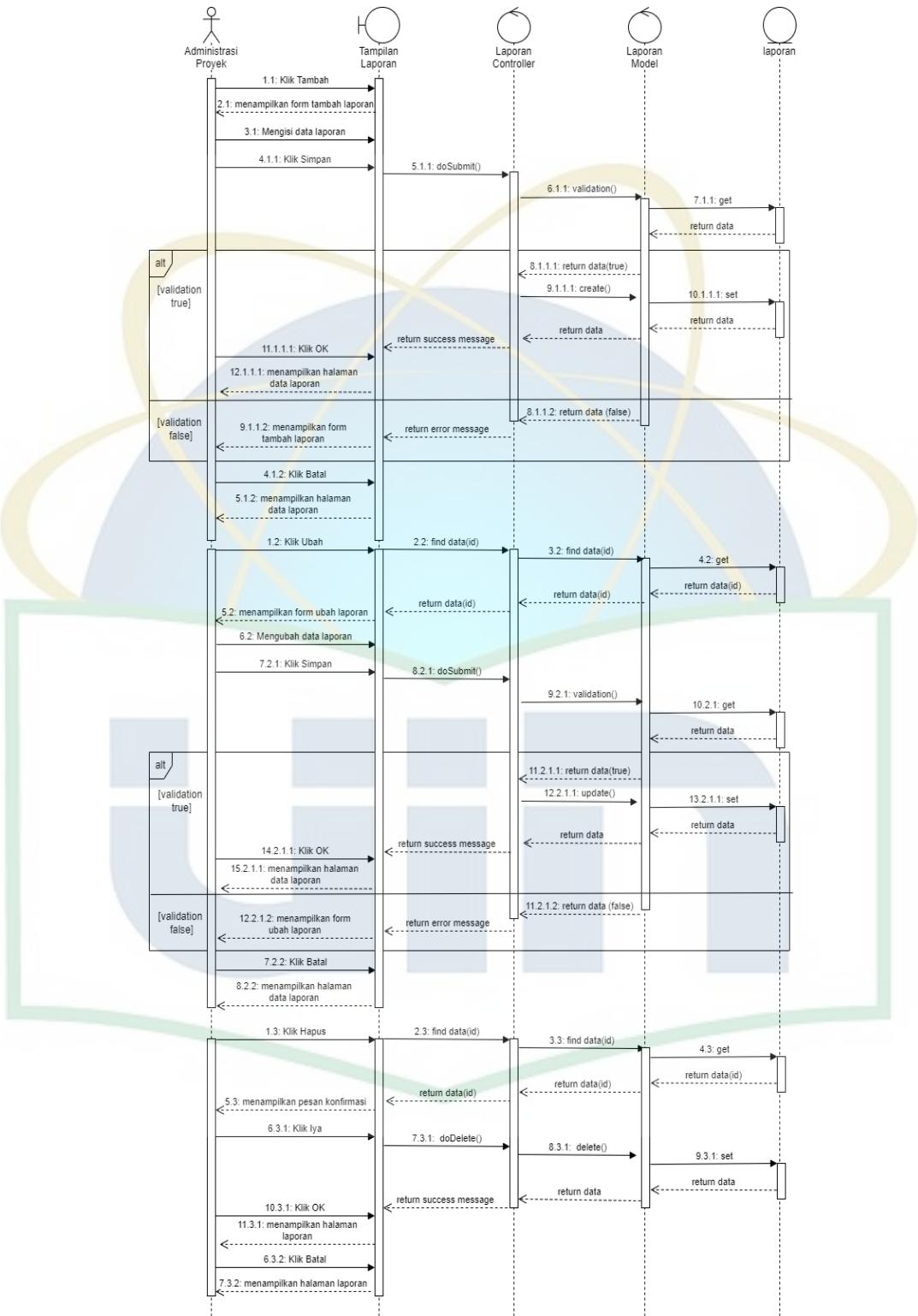


Gambar 4.33 Sequence Diagram Analisis Keterlambatan Proyek

Pada *sequence diagram* Gambar 4.33, kepala divisi proyek menganalisis keterlambatan proyek dengan menggunakan 2 metode yaitu *critical path method* dan *what if analysis*. Pertama sistem membuat jalur kritis setelah kepala divisi proyek klik Buat Jalur Kritis lalu sistem akan otomatis menghitung dengan *critical path method* untuk menentukan jalur kritis. Selanjutnya klik analisis pada aktivitas proyek yang terlambat dan sistem akan otomatis menganalisis jumlah pekerja dan waktu tambahan dengan *what if analysis*.



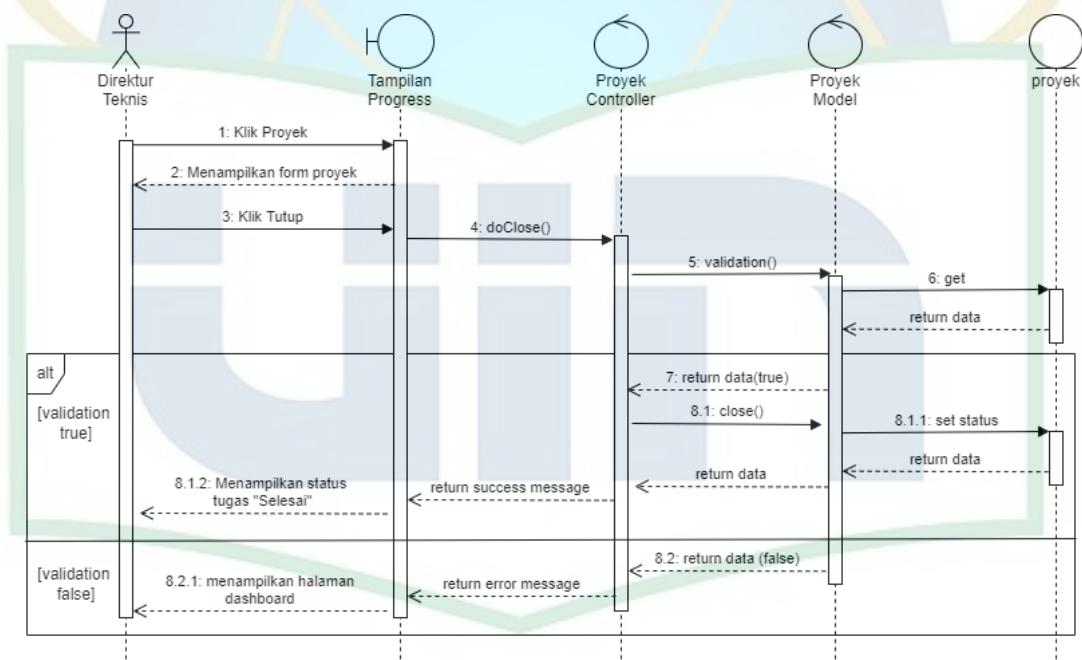
12. Sequence Diagram Kelola Laporan



Gambar 4.34 Sequence Diagram Kelola Laporan

Pada *sequence diagram* Gambar 4.34, administrasi proyek melakukan pengelolaan data laporan dengan tambah, ubah, dan hapus data-data laporan. Administrasi proyek dapat klik tambah atau ubah data lalu sistem akan mevalidasi data. Jika sudah sesuai maka sistem menyimpan menambah atau mengubah data dan menampilkan data tersebut. Jika validasi gagal maka akan kembali ke halaman data laporan. Admin juga dapat menghapus data dengan klik hapus lalu sistem akan menghapus data dari *database*.

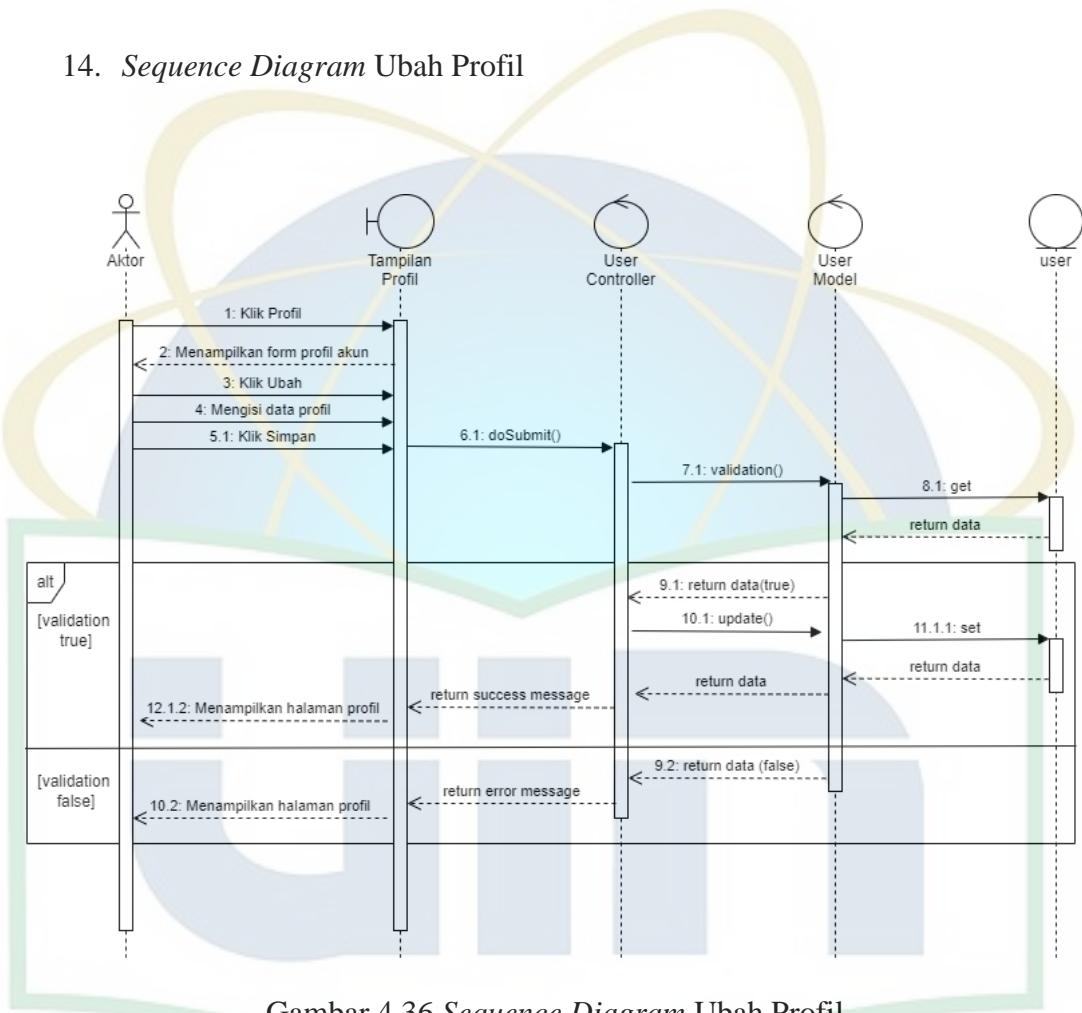
13. Sequence Diagram Tutup Proyek



Gambar 4.35 *Sequence Diagram Tutup Proyek*

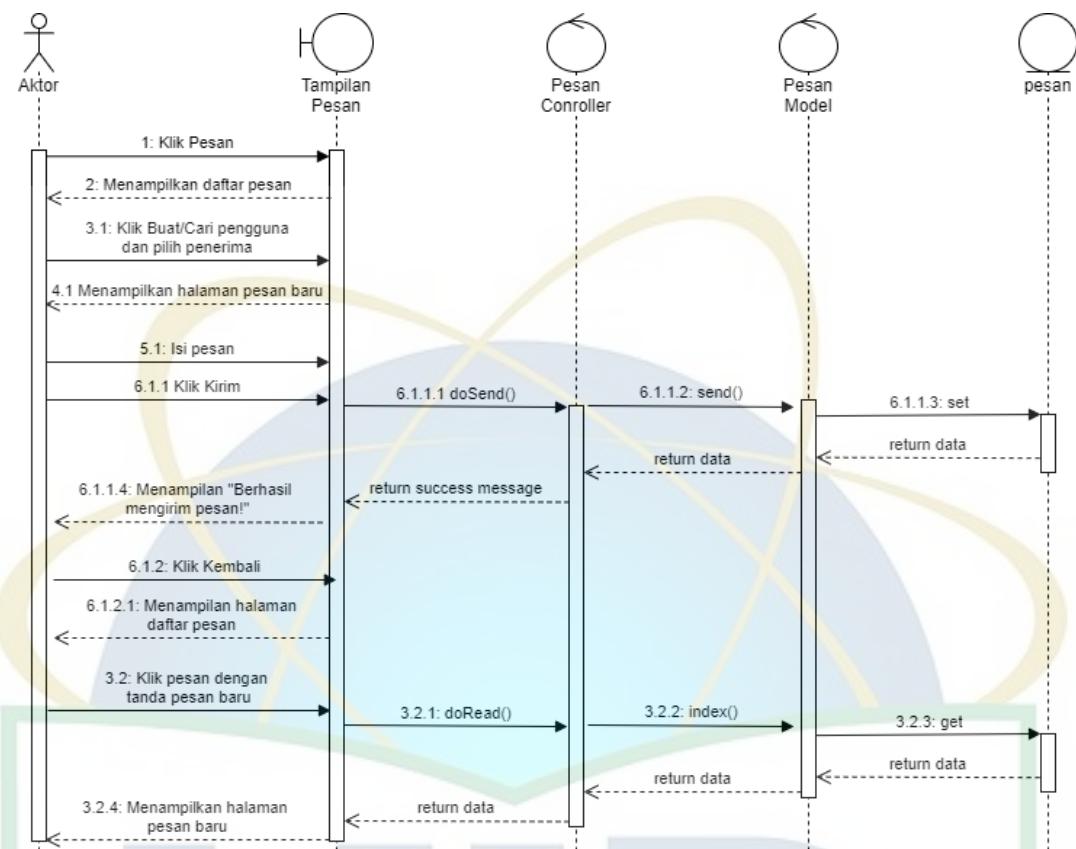
Pada *sequence diagram* Gambar 4.35, direktur teknis menutup proyek yang telah selesai dengan cara klik tutup dan jika masih terdapat kekurangan klik revisi dan alasan.

14. Sequence Diagram Ubah Profil



Pada *sequence diagram* Gambar 4.36, semua aktor mengubah profil akunnya dengan cara mengisi dengan data baru dan klik simpan lalu sistem akan mengubah data dari *database*.

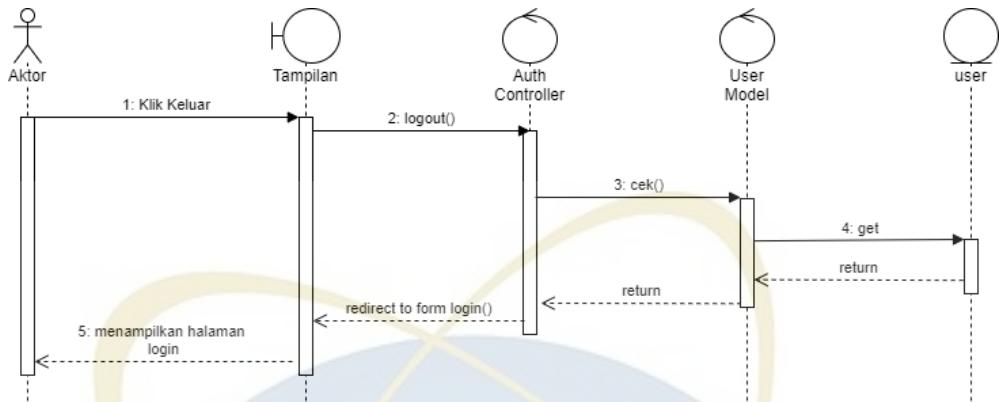
15. Sequence Diagram Kelola Pesan



Gambar 4.37 Sequence Diagram Kelola Pesan

Pada *sequence diagram* Gambar 4.37, semua aktor mengirim dan menerima pesan kepada akun yang dituju.

16. Sequence Diagram Logout



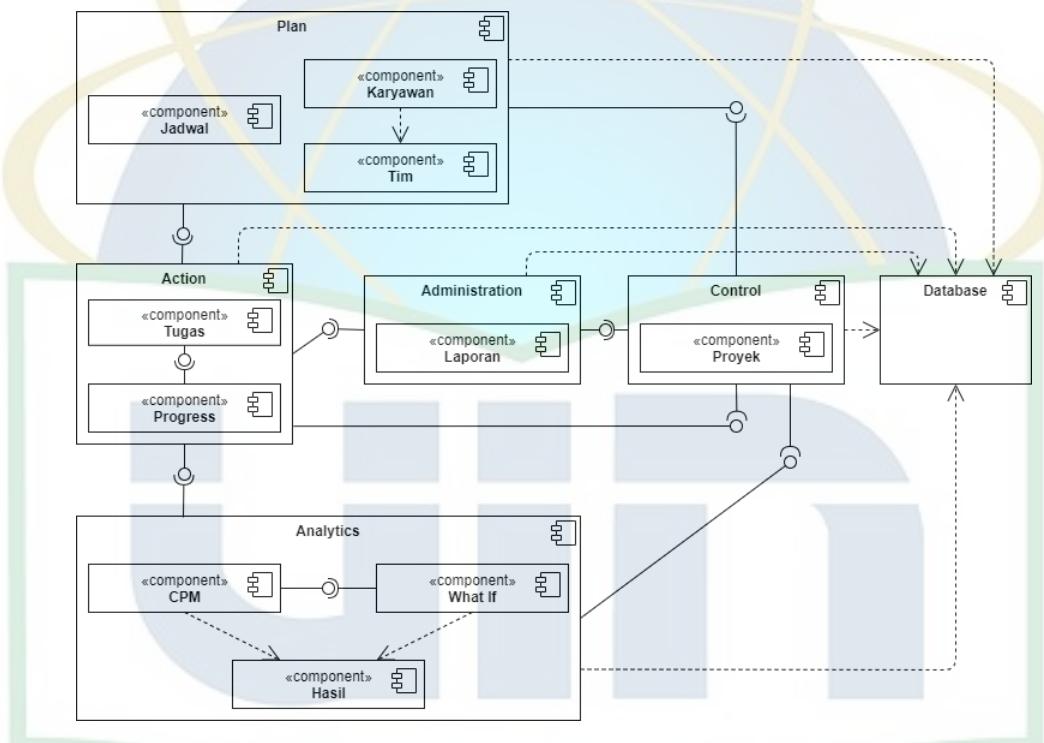
Gambar 4.38 Sequence Diagram Logout

Pada *sequence diagram* Gambar 4.38, seluruh aktor keluar dari sistem dengan klik keluar lalu sistem akan menampilkan halaman *login* kembali.

4.2.2.5 Component Diagram

Pada *component diagram* Gambar 4.39 menjelaskan komponen-komponen dan hubungannya. Terdapat komponen *plan*, *action*, *administration*, *control*, *analytics* dan *database*. Dalam komponen *plan* terdapat komponen jadwal dan komponen tim yang bergantung pada komponen karyawan. Komponen *plan* menyediakan *interface* untuk komponen *control* dan *action* serta bergantung pada *database*. Dalam komponen *action* terdapat komponen tugas yang menyediakan *interface* untuk komponen *progress*. Komponen *action* menyediakan *interface* untuk komponen *administration*, *control* dan *analytics* dan bergantung pada *database*. Dalam komponen *analytics* terdapat komponen CPM yang menyediakan *interface* untuk komponen *whatif*. Kedua komponen tersebut memiliki

ketergantungan dengan komponen hasil. Komponen *analytics* menyediakan interface untuk komponen *control* dan bergantung pada *database*. Dalam komponen *administration* terdapat komponen laporan. Komponen *administration* menyediakan *interface* untuk komponen *control* dan bergantung pada *database*. Dalam komponen *control* terdapat komponen proyek. Komponen *control* membutuhkan *interface* dari seluruh komponen lainnya.

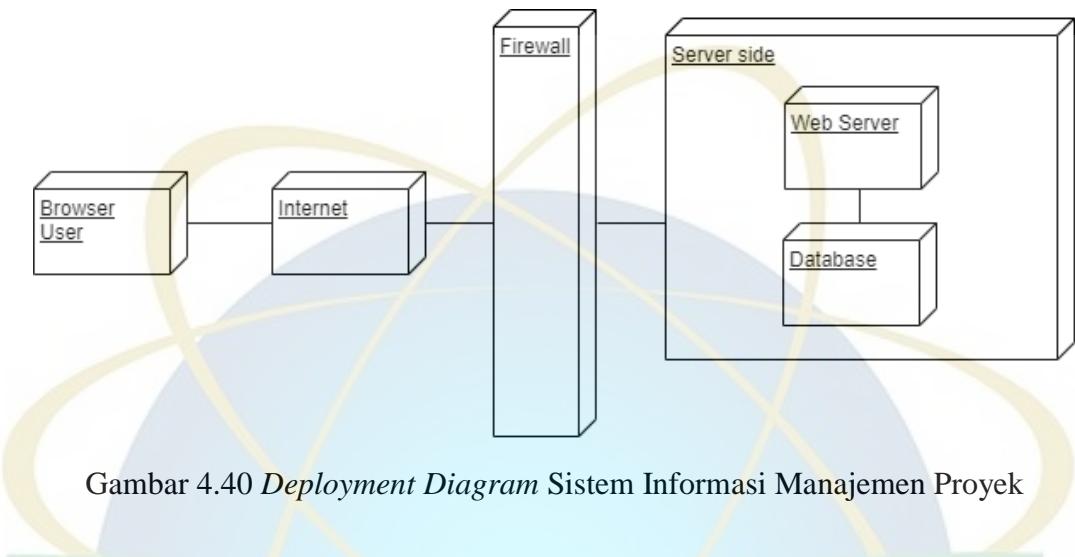


Gambar 4.39 *Component Diagram* Sistem Informasi Manajemen Proyek

4.2.2.6 Deployment Diagram

Pada diagram menggambarkan bentuk fisik dari sistem. Untuk mengakses sistem, *user* perlu membuka *browser* yang sudah terkoneksi dengan internet dan

dilindungi *firewall*. Terdapat pula perangkat *server* yang terdiri dari *web server* (Apache) dan *database* (MySQL).



Gambar 4.40 *Deployment Diagram* Sistem Informasi Manajemen Proyek

4.2.3 Perancangan *Database*

4.2.3.1 Normalisasi *Database*

Dalam perancangan *database* peneliti menggunakan normalisasi *database* untuk meminimalkan redundansi data. Dalam normalisasi *database*, peneliti menggunakan dokumen-dokumen perusahaan dan *class diagram* sebagai sumber data untuk diolah kedalam tiga bentuk normal form yaitu *first normal form* (1NF), *second normal form* (2NF), *third normal form* (3NF). Sebelum sumber data dibentuk kedalam 1NF, semua data yang berada pada sumber data dipindahkan kedalam satu format tabel atau *unnormalized form* (UNF).

Tabel 4.24 *Unnnormalized Form*

Proyek id	program	kegiatan	sub kegiatan	rincian sub kegiatan	tipe	Nama paket	Kode rekening	lokasi	tahun	Durasi proyek
1	Pengelolaan Keanekaragaman Hayati Kabupaten/Kota	Pengelolaan Keanekaragaman Hayati	Pengelolaan Ruang Terbuka Hijau (RTH)	Pembangunan RTH Taman di Wilayah Jakarta Barat	Pengawasan	Pembangunan RTH Jl. Duri Kosambi (Pulo Indah Raya)	Belanja Modal Taman	Jakarta Barat	2021	90 hari

Waktu mulai proyek	Waktu selesai proyek	Tgl cuaca	Keadaan cuaca	Total pic	Normal time	man hour	aktivitas	Durasi aktivitas	successor	Nama karyawan	Tempat lahir karyawan
24 Juni 2019	31 Desember 2019	24 Juni 2019	Cerah	3 orang	8 jam	2160 mh	Pekerjaan Lapangan Olahraga	16 hari	Pekerjaan Pendahuluan	Samal	Riau
		25 Juni 2019	Mendung				Pekerjaan Area Feature Wall	7 hari	Pekerjaan lapangan olahraga	Safira	Jakarta
		26 Juni 2019	Hujan						Pekerjaan Perkerasan Inlet	Fahira	

Tanggal lahir karyawan	Alamat karyawan	Email karyawan	Telp karyawan	Posisi karyawan	npwp	Judul tugas	Deskripsi tugas	Status tugas
30 Mei 1966	Blok A3, Jakarta selatan	samal@gmail.com	08123231232	Team Leader	2487483 4798324	Mengikuti rapat	Menghadiri rapat dan mengisi absen	Sedang dikerjakan
12 Oktober 1995	Perum Asri, Tangerang selatan	safira@gmail.com fahira@gmail.com	08222322345 4	Inspektor	3242154 3535324	Mealukan pengawasan ke lokasi	Dokumentasikan proyek yang sedang dikerjakan	Belum dikerjakan
2 Mei 2000	Pamulang Estate, Tangerang Selatan		08567892131 3	Computer operator	4564989 5344547	Membuat desain pedestrian	Buat model 3D pedestrian pada bagian selatan proyek	Revisi

Nama perusahaan klien	Nama pic klien	Telp klien	Judul laporan	Tipe laporan	Status laporan	Jalur kritis	<i>Additional employee</i>	<i>Additional hour</i>
BNI	Agus	083421342	Laporan Bulanan 1 Proyek	Bulanan	Belum diverifikasi	A-B-C	2 orang	1 jam
			Laporan mingguan 2	Mingguan	Revisi		3 orang	
			Laporan Akhir	Akhir	Sudah diverifikasi		1 orang	2 jam

<i>message</i>	pengirim	penerima	<i>Username</i>	<i>Password</i>	<i>Role</i>
Tolong periksa	Fahira	Safira	Safira	12345	Admin
Baik pak	Safira	Samal	Samal	54321	Direktur teknis
Sudah saya periksa		Fahira	fahira	00000	Administrasi proyek

Dalam bentuk UNF, masih terdapat data yang berulang atau *multivalue*. Seperti cuaca, aktivitas, successor, laporan, tugas. Maka untuk membentuk 1NF, semua data yang multivalue dihilangkan.



Tabel 4.25 First Normal Form

Proyek id	program	kegiatan	sub kegiatan	rincian sub kegiatan	tipe	Nama paket	Kode rekening	lokasi	tahun	Durasi proyek
PRY0 62100 01	Pengelolaan Keanekaraga man Hayati	Pengelolaan Keanekaragaman Hayati Kabupaten/Kota	Pengelolaan Ruang Terbuka Hijau (RTH)	Pembangunan RTH Taman di Wilayah Jakarta Barat	Penga wasan	Pembangunan RTH Jl. Duri Kosambi (Pulo Indah Raya)	Belanja Modal Taman	Jakarta Barat	2021	90 hari
PRY0 62100 01	Pengelolaan Keanekaraga man Hayati	Pengelolaan Keanekaragaman Hayati Kabupaten/Kota	Pengelolaan Ruang Terbuka Hijau (RTH)	Pembangunan RTH Taman di Wilayah Jakarta Barat	Penga wasan	Pembangunan RTH Jl. Duri Kosambi (Pulo Indah Raya)	Belanja Modal Taman	Jakarta Barat	2021	90 hari
PRY0 62100 01	Pengelolaan Keanekaraga man Hayati	Pengelolaan Keanekaragaman Hayati Kabupaten/Kota	Pengelolaan Ruang Terbuka Hijau (RTH)	Pembangunan RTH Taman di Wilayah Jakarta Barat	Penga wasan	Pembangunan RTH Jl. Duri Kosambi (Pulo Indah Raya)	Belanja Modal Taman	Jakarta Barat	2021	90 hari

Waktu mulai proyek	Waktu selesai proyek	Status Proyek	Cuaca id	Tgl cuaca	Keadaan cuaca	Jadwal id	Total pic	Normal time	manhour	Aktivitas id	aktivitas	Durasi aktivitas
24 Juni 2019	31 Desember 2019	Berlang sung	CUA0621001	24 Juni 2019	Cerah	JWL06210001	3 orang	8 jam	2160 mh	AKT0621001	Pekerjaan Lapangan Olahraga	16 hari
24 Juni 2019	31 Desember 2019	Berlang sung	CUA0621002	25 Juni 2019	Mendung	JWL06210001	3 orang	8 jam	2160 mh	AKT0621002	Pekerjaan Area Feature Wall	7 hari
24 Juni 2019	31 Desember 2019	Berlang sung	CUA0621003	26 Juni 2019	Hujan	JWL06210001	3 orang	8 jam	2160 mh	AKT0621002	Pekerjaan Area Feature Wall	7 hari

Succ esso r id	successor	Tim id	Tim detail id	Karya wan id	Nama karyawan	Tempat lahir karyawan	Tanggal lahir karyawan	Alamat karyawan	Email karyawan	Telp karyawan	Npwp
SUC 0621 0001	Pekerjaan Pendahulun	TIM06 210001	TMD0 62100 01	KYW 06210 001	Samal	Riau	30 Mei 1966	Blok A3, Jakarta selatan	samal@gmail.co	081232312 1234	223123
SUC 0621 0002	Pekerjaan lapangan olahraga	TIM06 210001	TMD0 62100 02	KYW 06210 002	Safira	Jakarta	12 Oktober 1995	PerumTangerang selatan	safira@gmail.co	082223223 4	533422 3423
SUC 0621 0003	Pekerjaan Perkerasan Inlet	TIM06 210001	TMD0 62100 03	KYW 06210 003	Fahira	Jakarta	2 Mei 2000	Pamulang Estate, Tangerang Selatan	fahira@gmail.co	085678921 3	423432 4234

Posisi id	Posisi karyawan	Tugas id	Judul tugas	Deskripsi tugas	deadline	Status tugas	Klien id	Nama perusahaan klien	Nama pic klien	Telp klien
PSS062 10001	Team Leader	TGS06 210001	Mengikuti rapat	Menghadiri rapat dan mengisi absen	1 September 2019	Sedang dikerjakan	KLN06 210001	BNI	Agus	0834 2134 2
PSS062 10001	Inspektor	TGS06 210002	Mealukan pengawasan ke lokasi	Dokumentasikan proyek yang sedang dikerjakan	5 September 2019	Belum dikerjakan	KLN06 210001	BNI	Agus	0834 2134 2
PSS062 10001	Computer Operator	TGS06 210003	Membuat desain pedestrian	Buat model 3D pedestrian pada bagian selatan proyek	8 Oktober 2019	Revisi	KLN06 210001	BNI	Agus	0834 2134 2

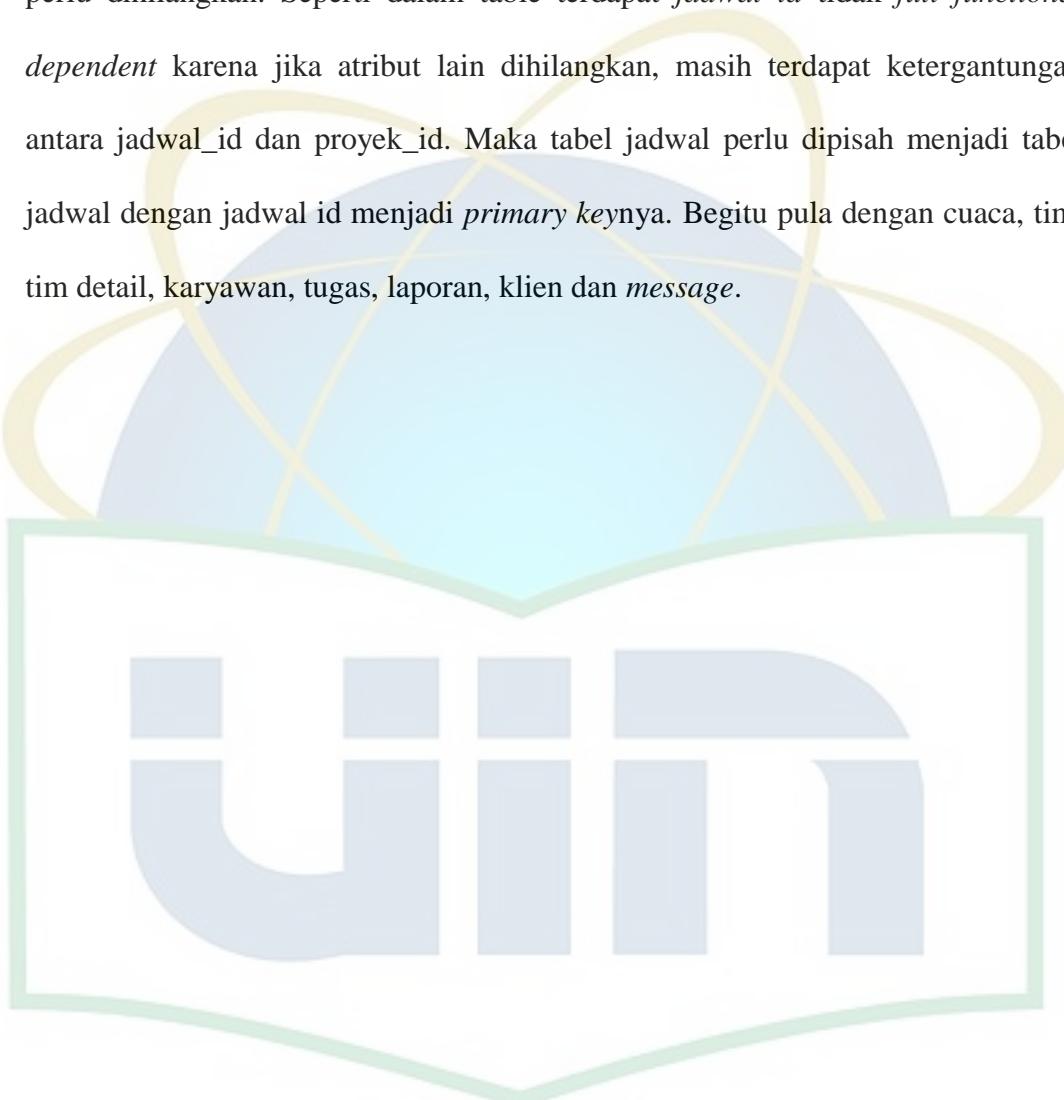
Cpm id	es	ef	ls	lf	float	critical	What if id	delay	Delay percent	Delay p	acceleration	condition	Additional employee	Addition al hour
CPM06 210001	0	10	0	10	0	T	WIF0621 0001	7	8	3.5	8.5	Yes	2 orang	1 jam
CPM06 210002	10	33	0	33	0	T	WIF 06210001	7	3	3.5	0.5	Yes	3 orang	1 jam
CPM06 210003	33	142	33	142	0	T	WIF 06210001	7	3	3.5	4.5	yes	1 orang	2 jam

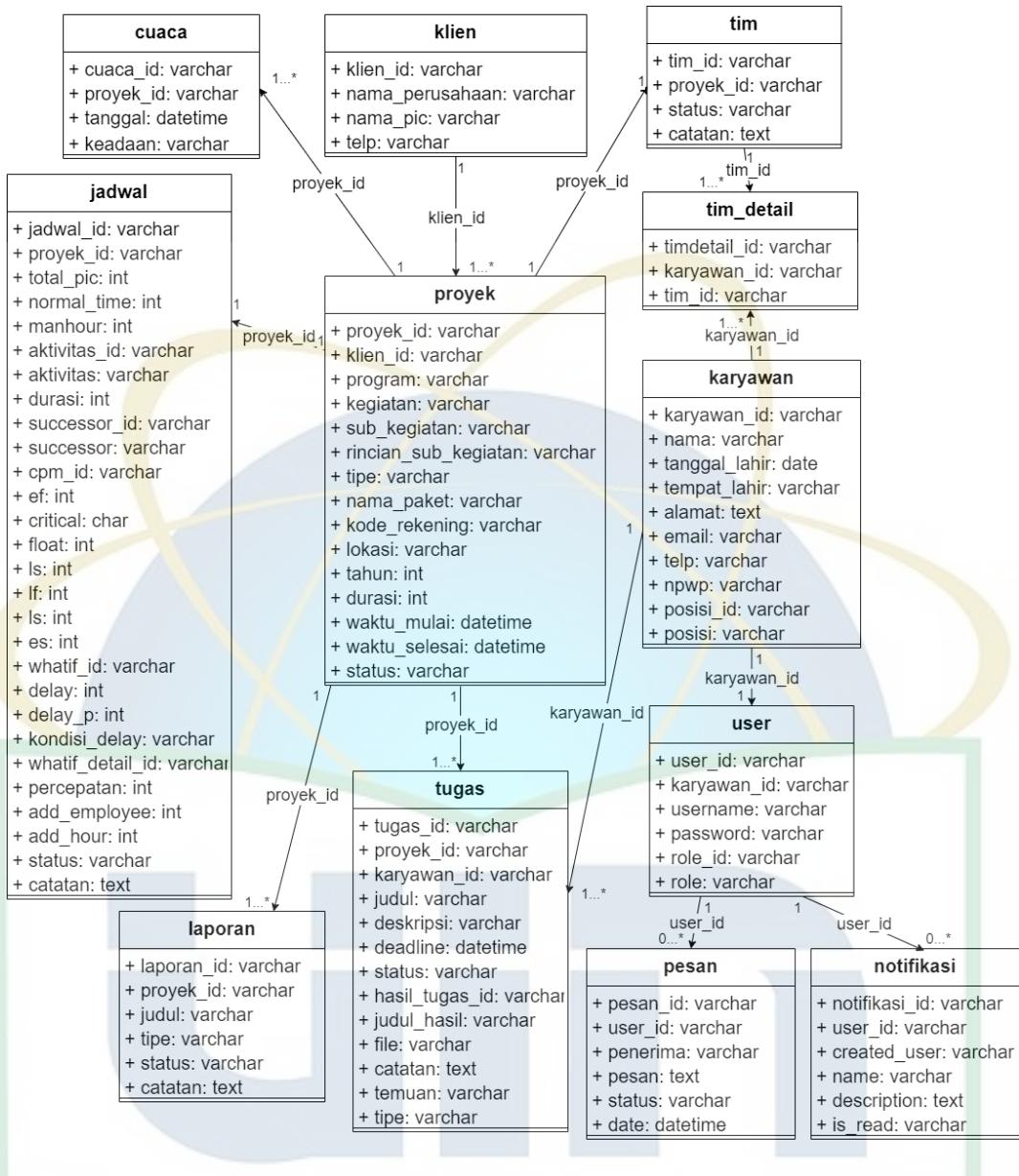
Laporan id	Judul laporan	Tipe laporan	Status laporan	Message id	message	pengirim	penerima	status
LAP06210001	Laporan Bulanan 1	Bulanan	Belum diverifikasi	MSG06210001	Tolong periksa	Fahira	Safira	Terkirim
LAP06210002	Laporan mingguan 2	Mingguan	Revisi	MSG06210002	Baik pak	Safira	Samal	Dibaca
LAP06210003	Laporan Akhir	Akhir	Sudah diverifikasi	MSG06210003	Sudah saya periksa	Safira	Fahira	Terikirm

User id	Username	Password	Role id	Role
USR06210001	Safira	12345	ROL06210001	Admin
USR06210002	Samal	54321	ROL06210002	Direktur teknis
USR06210003	fahira	00000	ROL06210003	Administrasi proyek



Dalam bentuk 1NF, masih terdapat data yang *partially dependent* atau tidak *full functional dependent* yaitu masih terdapat atribut yang tidak bergantung secara penuh ke *primary key*. Untuk membentuk 2NF, data yang masih *partially dependent* perlu dihilangkan. Seperti dalam table terdapat *jadwal id* tidak *full functional dependent* karena jika atribut lain dihilangkan, masih terdapat ketergantungan antara *jadwal_id* dan *proyek_id*. Maka tabel *jadwal* perlu dipisah menjadi tabel *jadwal* dengan *jadwal id* menjadi *primary key*nya. Begitu pula dengan cuaca, tim, tim detail, karyawan, tugas, laporan, klien dan *message*.

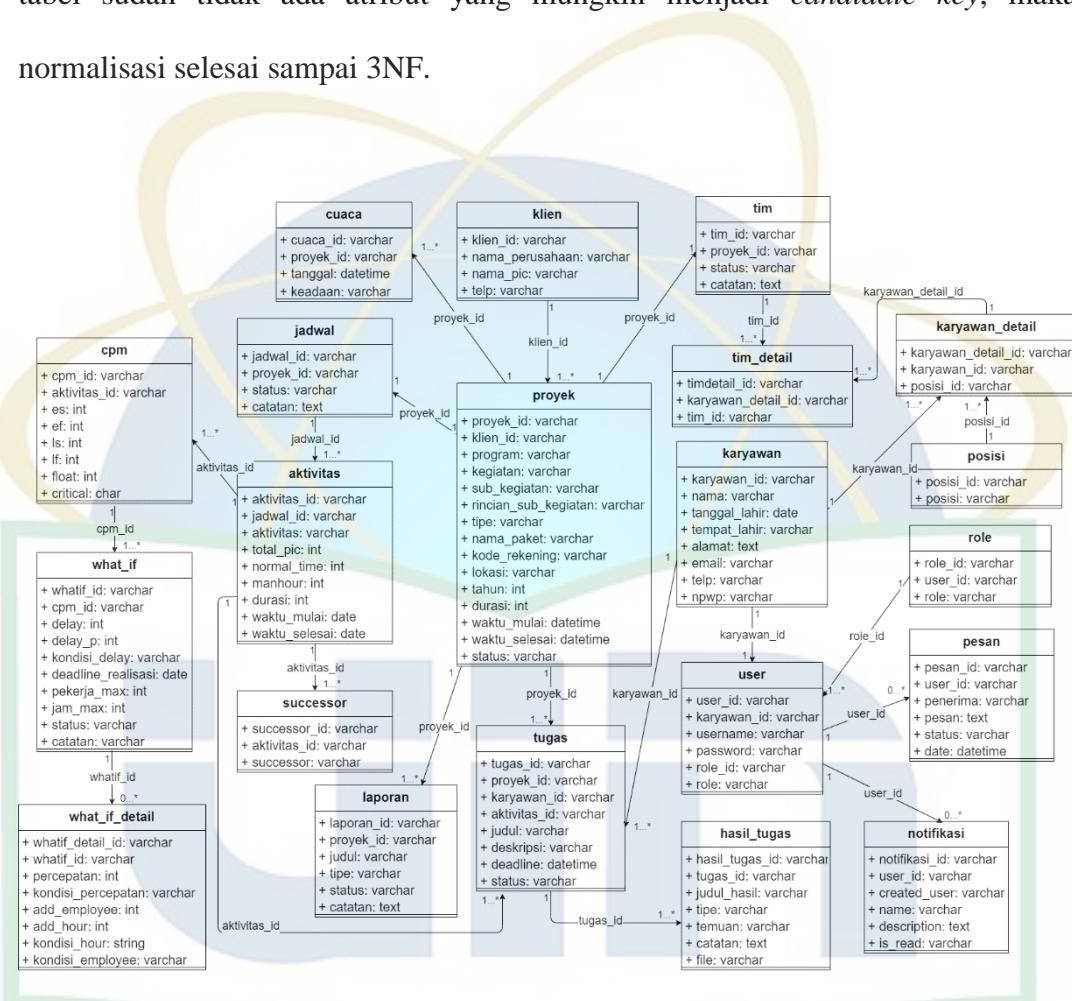




Gambar 4.41 Second Normal Form

Dalam bentuk 2NF masih terdapat data yang mempunyai *transitive dependency* yaitu atribut yang masih mempunyai ketergantungan dengan atribut lain dalam sebuah tabel. Untuk membentuk 3NF, hilangkan data yang mempunyai *transitive dependency*. Seperti aktivitas pada table jadwal. Atribut aktivitas dan durasi mempunyai ketergantungan terhadap aktivitas *id*, sehingga aktivitas perlu

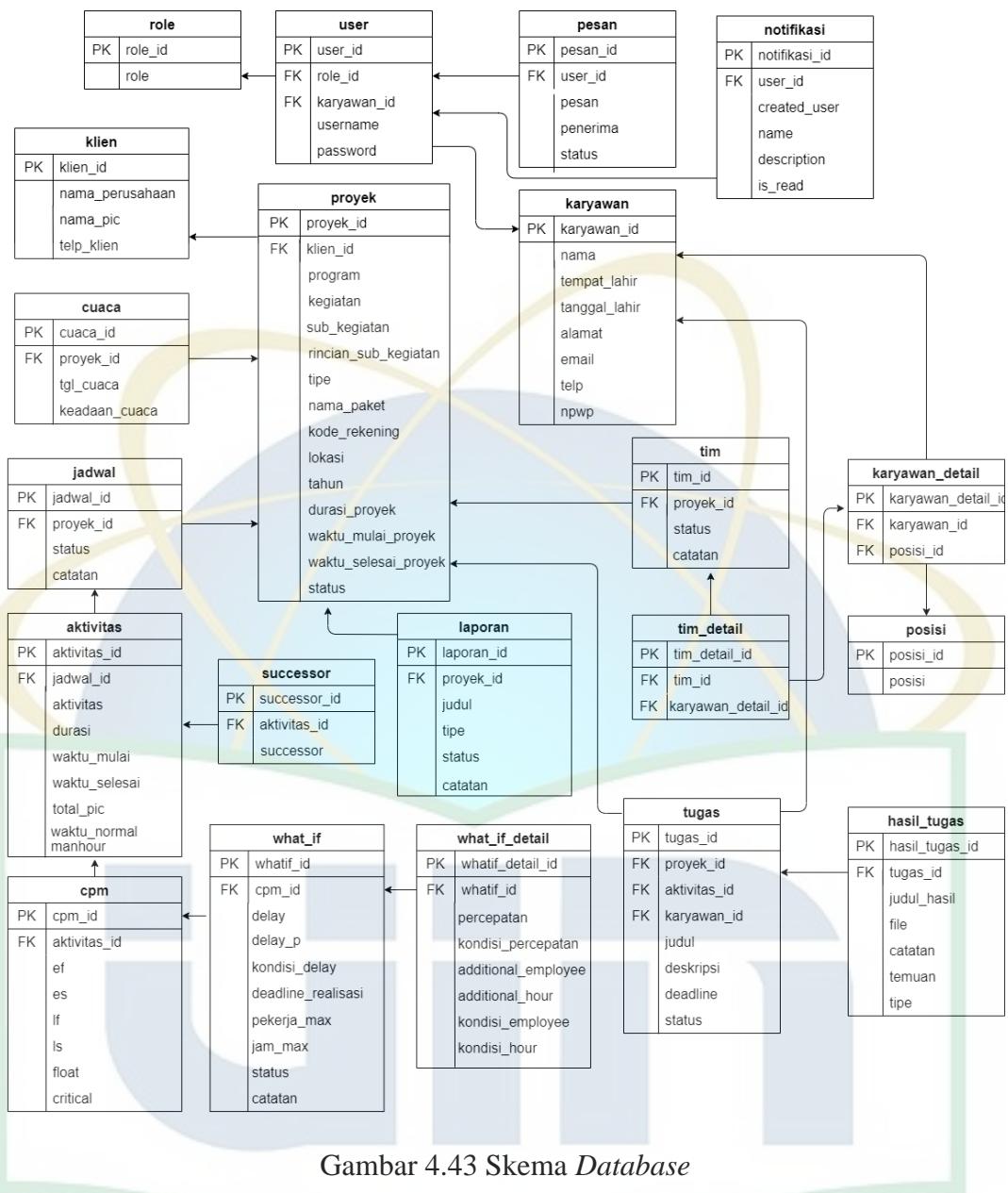
dibuatkan table sendiri dengan aktivitas id menjadi *primary key*nya dan aktivitas dan durasi menjadi atributnya, serta jadwal id menjadi *foreign key*nya. Begitu pula dengan *successor*, *cpm*, *whatif*, *role*, posisi dan *user*, karena pada masing-masing tabel sudah tidak ada atribut yang mungkin menjadi *candidate key*, maka normalisasi selesai sampai 3NF.



Gambar 4.42 Third Normal Form

4.2.3.2 Skema Database

Peneliti membuat skema database untuk menggambarkan hubungan *primary key* dan *foreign key* antar table.



Gambar 4.43 Skema Database

4.2.3.3 Spesifikasi Database

Berikut spesifikasi dari masing-masing tabel dalam sistem *database* sistem informasi manajemen proyek.

1. Tabel *User*

Nama : *User*

Tipe : Master

Primary Key : user_id

Foreign Key : karyawan_id, role_id

Tabel 4.26 Tabel *User*

No.	Field	Type	Size	Description
1	<i>user_id</i>	varchar	14	PK <i>User</i> Kode USR <u>00 00 0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
2	karyawan_id	varchar	14	FK <i>User</i> Kode KYW <u>00 00 0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
3	<i>role_id</i>	varchar	14	FK <i>User</i> Kode ROL <u>00 00 0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
4	<i>username</i>	varchar	16	<i>Username</i> pengguna
5	<i>password</i>	varchar	16	<i>Password</i> pengguna

2. Tabel Role

Nama : *Role*

Tipe : Master

Primary Key : role_id

Foreign Key : -

Tabel 4.27 Tabel *Role*

No.	Field	Type	Size	Description
1	<i>role_id</i>	varchar	14	PK <i>Role</i> Kode ROL <u>00</u> <u>00</u> <u>0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
2	<i>role</i>	varchar	20	Nama <i>role</i>

3. Tabel Karyawan

Nama : Karyawan

Tipe : Master

Primary Key : karyawan_id

Foreign Key : -

Tabel 4.28 Tabel Karyawan

No.	Field	Type	Size	Description
1	karyawan_id	varchar	14	PK karyawan Kode

				KYW <u>00</u> <u>00</u> <u>0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
2	nama	varchar	26	Nama karyawan
3	tanggal_lahir	date		Tanggal lahir karyawan
4	tempat_lahir	varchar	35	Tempat lahir karyawan
5	alamat	text		Alamat karyawan
6	<i>email</i>	varchar	80	<i>Email</i> karyawan
7	telp	varchar	15	Nomor telepon karyawan
8	npwp	varchar	15	Nomor NPWP karyawan

4. Tabel Posisi

Nama : Posisi
 Tipe : Master
 Primary Key : posisi_id
 Foreign Key : -

Tabel 4.29 Tabel Posisi

No.	Field	Type	Size	Description
1	posisi_id	varchar	14	PK posisi Kode POS <u>00</u> <u>00</u> <u>0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
2	posisi	varchar	80	Nama posisi

5. Tabel Karyawan Detail

Nama : Karyawan detail

Tipe : Transaksi

Primary Key : karyawan_detail_id

Foreign Key : karyawan_id, posisi_id

Tabel 4.30 Tabel Karyawan Detail

No.	Field	Type	Size	Description
1	karyawan_detail_id	varchar	14	PK karyawan detail Kode <u>KYD 00 00 0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
2	karyawan_id	varchar	14	FK karyawan Kode <u>KYW 00 00 0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
3	posisi_id	varchar	14	FK posisi Kode <u>POS 00 00 0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data

6. Tabel Pesan

Nama : Pesan

Tipe : Transaksi

Primary Key : message_id

Foreign Key : user_id

Tabel 4.31 Tabel Pesan

No.	Field	Type	Size	Description
1	pesan_id	varchar	14	PK pesan Kode MSG <u>00 00 0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
2	user_id	varchar	14	FK message Kode USR <u>00 00 0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
3	penerima	varchar	14	Penerima pesan
4	pesan	text		Isi pesan
5	status	varchar	1	Status pesan 0: Terkirim 1: Dibaca
6	date	datetime		Tanggal dan waktu pesan dibuat

7. Tabel Klien

Nama : Klien

Tipe : Master

Primary Key : klien_id

Foreign Key : -

Tabel 4.32 Tabel Klien

No.	Field	Type	Size	Description
1	klien_id	varchar	14	PK klien Kode KLN <u>00</u> <u>00</u> <u>0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
2	nama_perusahaan	varchar	50	Nama perusahaan klien
3	nama_pic	varchar	26	Nama <i>person in charge</i> dari perusahaan klien
4	telp_klien	varchar	15	Nomor telepon klien

8. Tabel Proyek

Nama : Proyek

Tipe : Transaksi

Primary Key : proyek_id

Foreign Key : klien_id

Tabel 4.33 Tabel Proyek

No.	Field	Type	Size	Description
1	proyek_id	varchar	14	PK proyek Kode PRY <u>00 00 0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
2	klien_id	varchar	14	FK proyek Kode KLN <u>00 00 0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
3	program	varchar	80	Nama program proyek
4	kegiatan	varchar	80	Nama kegiatan proyek
5	sub_kegiatan	varchar	80	Nama sub kegiatan proyek
6	rincian_sub_kegiatan	varchar	80	Nama rincian sub kegiatan
7	tipe	varchar	15	Tipe proyek
8	nama_paket	varchar	80	Nama paket proyek
9	kode_rekening	varchar	80	Kode rekening proyek
10	lokasi	varchar	35	Lokasi kota proyek
11	tahun	int	4	Tahun pelaksanaan proyek
12	durasi	int	4	Durasi waktu proyek dalam hari

13	waktu_mulai	datetime		Waktu proyek dimulai
14	waktu_selesai	datetime		Waktu selesai proyek
15	status	varchar	1	Status proyek 0: Berlangsung 1: Selesai

9. Tabel Cuaca

Nama : Cuaca

Tipe : Transaksi

Primary Key : cuaca_id

Foreign Key : proyek_id

Tabel 4.34 Tabel Cuaca

No.	Field	Type	Size	Description
1	cuaca_id	varchar	14	PK cuaca Kode CUA <u>00 00 0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
2	proyek_id	varchar	14	FK cuaca Kode PRY <u>00 00 0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
3	tanggal	datetime		Tanggal saat pelaporan cuaca dilakukan

4	keadaan	varchar	1	Keadaan pelaporan dilakukan 0: Hujan 1: Cerah 2: Berawan	cuaca	saat cuaca
---	---------	---------	---	---	-------	------------

10. Tabel Jadwal

Nama : Jadwal

Tipe : Transaksi

Primary Key : jadwal_id

Foreign Key : proyek_id

Tabel 4.35 Tabel Jadwal

No.	Field	Type	Size	Description
1	jadwal_id	varchar	14	PK jadwal Kode JWL <u>00 00 0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
2	proyek_id	varchar	14	FK jadwal Kode PRY <u>00 00 0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
3	status	varchar	1	status jadwal 0: Belum Diverifikasi

				1: Diverifikasi 2: Ditolak
4	catatan	text		Catatan jika menolak jadwal

11. Tabel Aktivitas

Nama : Aktivitas

Tipe : Transaksi

Primary Key : aktivitas_id

Foreign Key : jadwal_id

Tabel 4.36 Tabel Aktivitas

No.	Field	Type	Size	Description
1	aktivitas_id	varchar	14	PK aktivitas Kode <u>AKT 00 00 0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
2	jadwal_id	varchar	14	FK jadwal Kode <u>JWL 00 00 0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
3	aktivitas	varchar	80	Nama aktivitas dalam jadwal proyek

4	durasi	int	3	Jumlah hari untuk pelaksanaan aktivitas
5	waktu_mulai	date		Waktu aktivitas dimulai
5	waktu_selesai	date		Waktu selesai aktivitas
6	<i>total_pic</i>	Int	3	Jumlah orang yang bekerja dalam aktivitas
7	waktu_normal	int	3	Jumlah jam kerja normal dalam sehari
8	<i>manhour</i>	int	3	Jumlah manhour dalam aktivitas

12. Tabel Successor

Nama : *Successor*

Tipe : Transaksi

Primary Key : *successor_id*

Foreign Key : *aktivitas_id*

Tabel 4.37 Tabel Successor

No.	Field	Type	Size	Description
1	<i>successor_id</i>	varchar	14	PK <i>successor</i> Kode SUC <u>00</u> <u>00</u> <u>0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
2	<i>aktivitas_id</i>	varchar	14	FK <i>successor</i> Kode AKT <u>00</u> <u>00</u> <u>0000</u>

				1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
3	<i>successor</i>	varchar	80	Nama aktivitas pengikut dari aktivitas_id

13. Tabel CPM

Nama : cpm

Tipe : Transaksi

Primary Key : cpm_id

Foreign Key : aktivitas_id

Tabel 4.38 Tabel CPM

No.	Field	Type	Size	Description
1	cpm_id	varchar	14	PK cpm Kode CPM <u>00 00 0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
2	aktivitas_id	varchar	14	FK cpm Kode AKT <u>00 00 0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
3	es	int	3	Jumlah waktu mulai paling awal dari aktivitas

4	ef	int	3	Jumlah waktu selesai paling awal dari aktivitas
5	ls	int	3	Jumlah waktu mulai paling terlambat dari aktivitas
6	lf	int	3	Jumlah waktu selesai paling terlambat dari aktivitas
7	<i>float</i>	int	3	Jumlah selisih waktu mulai dan selesai
8	<i>critical</i>	boolean	1	Kondisi aktivitas proyek kritis atau tidak

14. Tabel What If

Nama : what if

Tipe : Transaksi

Primary Key : whatif_id

Foreign Key : aktivitas_id

Tabel 4.39 Tabel *What If*

No.	Field	Type	Size	Description
1	<i>whatif_id</i>	varchar	14	PK <i>what if</i> Kode WIF <u>00 00 0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
2	cpm_id	varchar	14	FK <i>what if</i>

				Kode CPM <u>00</u> <u>00</u> <u>0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
3	<i>delay</i>	int	3	Jumlah hari terlambat dalam aktivitas
4	<i>delay_p</i>	int	3	Jumlah hari terlambat dalam proyek
5	<i>kondisi_delay</i>	char	1	Kondisi jika aktivitas proyek terlambat akan mempengaruhi keterlambatan proyek atau tidak
6	<i>deadline_realisasi</i>	date		Tanggal sebenarnya atau perkiraan aktivitas selesai
7	<i>pekerja_max</i>	int	3	Jumlah orang maksimum yang bisa ditambahkan jika aktivitas terlambat
8	<i>jam_max</i>	int	3	Jumlah orang maksimum yang bisa ditambahkan jika aktivitas terlambat
9	<i>status</i>	varchar	1	status analisis 0: Belum mengajukan 1: Belum Diverifikasi 2: Diverifikasi 3: Ditolak
10	<i>catatan</i>	text		Catatan jika menolak analisis

15. Tabel *WhatIf Detail*

Nama : what if detail

Tipe : Transaksi

Primary Key : whatif_detail_id

Foreign Key : whatif_id

Tabel 4.40 Tabel *What If Detail*

No.	Field	Type	Size	Description
1	<i>whatif_detail_id</i>	varchar	14	PK <i>what if detail</i> Kode <u>WFD 00 00 0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
2	<i>whatif_id</i>	varchar	14	FK <i>what if</i> Kode <u>WIF 00 00 0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
3	percepatan	int	3	Jumlah hari yang dibutuhkan untuk aktivitas
4	kondisi_percepatan	char	1	Kondisi untuk memeriksa hasil percepatan aktivitas pengikut memenuhi kedua batasan atau tidak

5	<i>additonal_employee</i>	int	3	Jumlah orang yang dibutuhkan untuk percepatan proyek
7	<i>additonal_hour</i>	int	3	Jumlah jam yang dibutuhkan untuk percepatan proyek
8	<i>kondisi_employee</i>	char	1	Kondisi untuk memeriksa hasil penambahan karyawan tidak melebih pekerja maksimum
9	<i>kondisi_hour</i>	char	1	Kondisi untuk memeriksa hasil penambahan jam tidak melebih jam maksimum

16. Tabel Tim

Nama : Tim

Tipe : Transaksi

Primary Key : tim_id

Foreign Key : proyek_id

Tabel 4.41 Tabel Tim

No.	Field	Type	Size	Description
1	tim_id	varchar	14	PK tim Kode TIM <u>00 00 0000</u>

				1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
2	proyek_id	varchar	14	FK tim Kode PRY <u>00 00 0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
3	status	varchar	1	Status tim 0: Belum Diverifikasi 1: Diverifikasi 2: Ditolak
4	catatan	text		Catatan jika menolak tim

17. Tabel Tim Detail

Nama : Tim Detail

Tipe : Transaksi

Primary Key : tim_detail_id

Foreign Key : tim_id, karyawan_detail_id

Tabel 4.42 Tabel Tim Detail

No.	Field	Type	Size	Description
1.	tim_detail_id	varchar	14	PK Tim detail Kode TMD <u>00 00 0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data

2	tim_id	varchar	14	FK tim detail Kode TIM 00 00 0000 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
3	karyawan_detail_id	varchar	14	FK tim detail Kode KYD 00 00 0000 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data

18. Tabel Tugas

Nama : Tugas
 Tipe : Transaksi
 Primary Key : tugas_id
 Foreign Key : proyek_id, karyawan_id, aktivitas_id

Tabel 4.43 Tabel Tugas

No.	Field	Type	Size	Description
1	tugas_id	varchar	14	PK tugas Kode TGS 00 00 0000 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
2	proyek_id	varchar	14	FK tugas Kode

				PRY <u>00 00 0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
4	karyawan_id	varchar	14	FK tugas atau penanggung jawab tugas Kode KYW <u>00 00 0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
5	aktivitas_id	varchar	14	FK tugas Kode AKT <u>00 00 0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
6	judul	varchar	80	Judul tugas
7	deskripsi	text		Deskripsi tugas
8	deadline	datetime		Waktu tenggat tugas harus selesai
9	status	varchar	1	Status tugas 0: Sedang Dikerjakan 1: Butuh Koreksi 2: Revisi 3: Selesai

19. Tabel Hasil Tugas

Nama : Hasil Tugas

Tipe : Transaksi

Primary Key : hasil_tugas_id

Foreign Key : tugas_id

Tabel 4.44 Tabel Hasil Tugas

No.	Field	Type	Size	Description
1	hasil_tugas_id	varchar	14	PK hasil tugas Kode <u>HSL 00 00 0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
2	tugas_id	varchar	14	FK hasil tugas Kode <u>TGS 00 00 0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
3	judul_hasil	varchar	80	Judul hasil tugas
4	file	varchar		File hasil tugas
5	catatan	text		Catatan tambahan seperti hasil rapat
6	temuan	text	1	Temuan dari proyek pengawasan
7	tipe	varchar	80	Tipe tugas yang diberikan

20. Tabel Laporan

Nama : Laporan

Tipe : Transaksi

Primary Key : laporan_id

Foreign Key : proyek_id

Tabel 4.45 Tabel Laporan

No.	Field	Type	Size	Description
1	laporan_id	varchar	14	PK laporan Kode LAP <u>00</u> <u>00</u> <u>0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
2	proyek_id	varchar	14	FK laporan Kode PRY <u>00</u> <u>00</u> <u>0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
3	judul	varchar	80	Judul laporan
4	tipe	varchar	10	Tipe laporan
5	status	varchar	1	Status laporan 0: Belum Diverifikasi 1: Diverifikasi 2: Ditolak

6	catatan	text		Catatan jika menolak laporan
---	---------	------	--	------------------------------

21. Tabel Notifikasi

Nama : Notifikasi

Tipe : Transaksi

Primary Key : notifikasi_id

Foreign Key : user_id

Tabel 4.46 Tabel Notifikasi

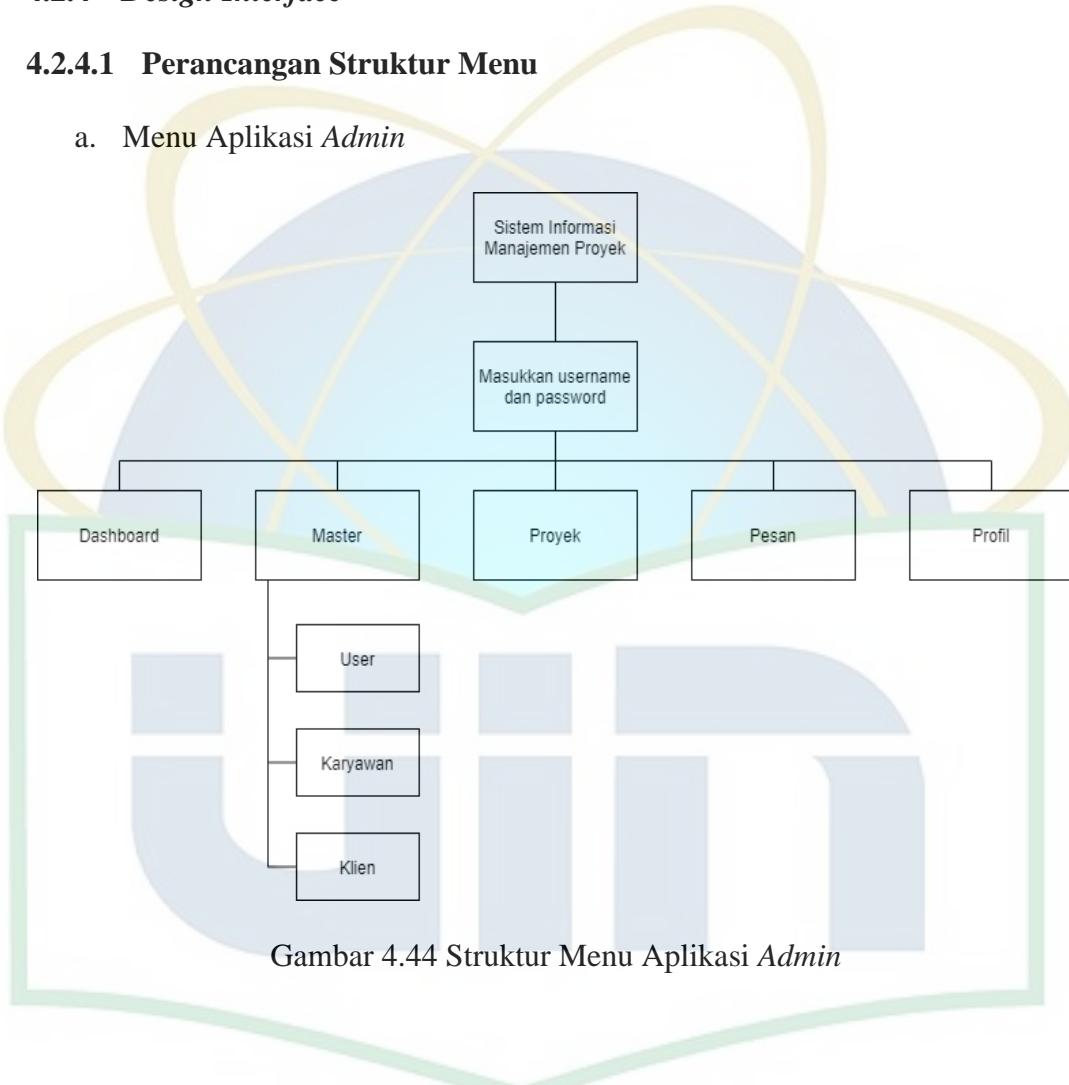
No.	Field	Type	Size	Description
1	notifikasi_id	varchar	14	PK notifikasi Kode NTF <u>00</u> <u>00</u> <u>0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
2	user_id	varchar	14	FK notifikasi Kode USR <u>00</u> <u>00</u> <u>0000</u> 1: Bulan data dibuat 2: Tahun data dibuat 3: No. urut data
3	created_user	varchar	14	Pembuat aktivitas
4	name	varchar	80	Nama modul
5	desciptiion	text		Deskripsi notifikasi dari aktivitas yang dilakukan
6	is_read	varchar	1	Status notifikasi

				0: Belum dibaca
				1: Sudah dibaca

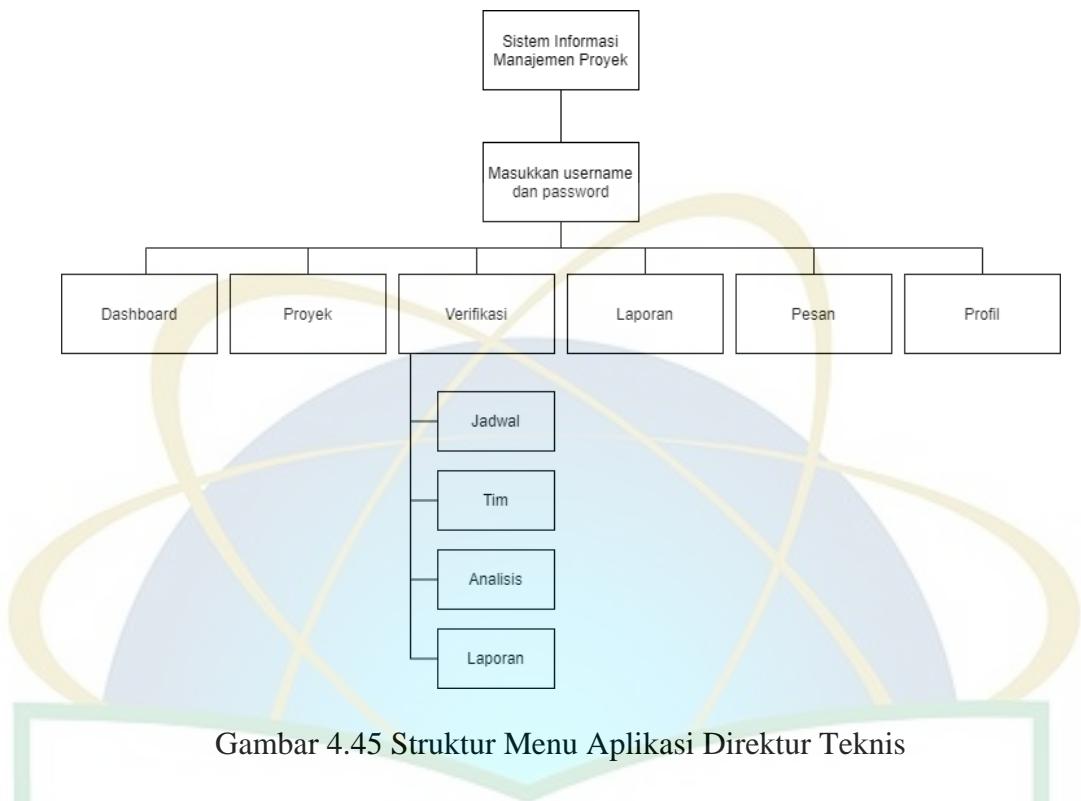
4.2.4 *Design Interface*

4.2.4.1 Perancangan Struktur Menu

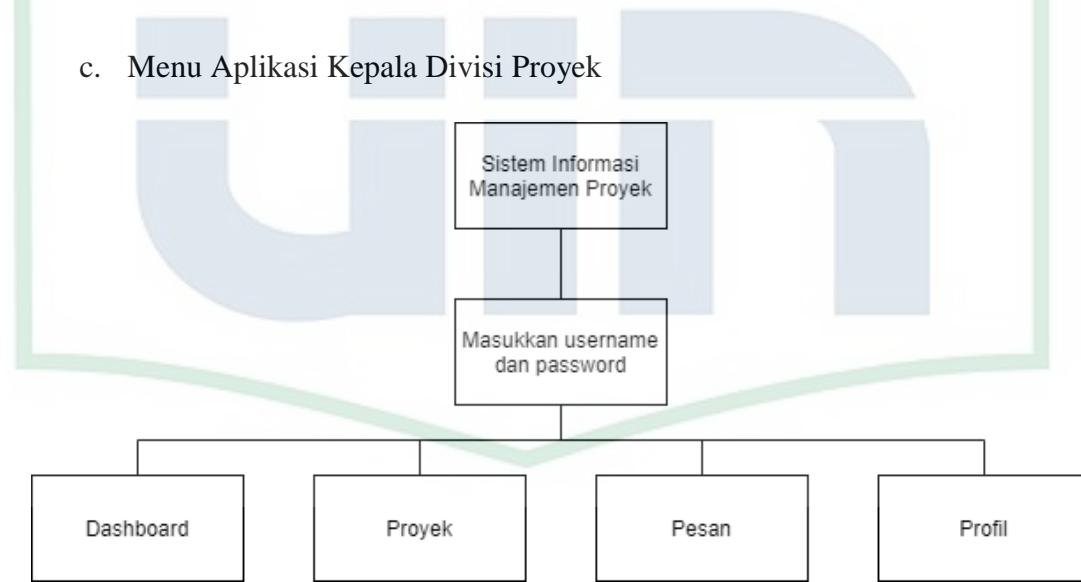
a. Menu Aplikasi Admin



b. Menu Aplikasi Direktur Teknis



c. Menu Aplikasi Kepala Divisi Proyek

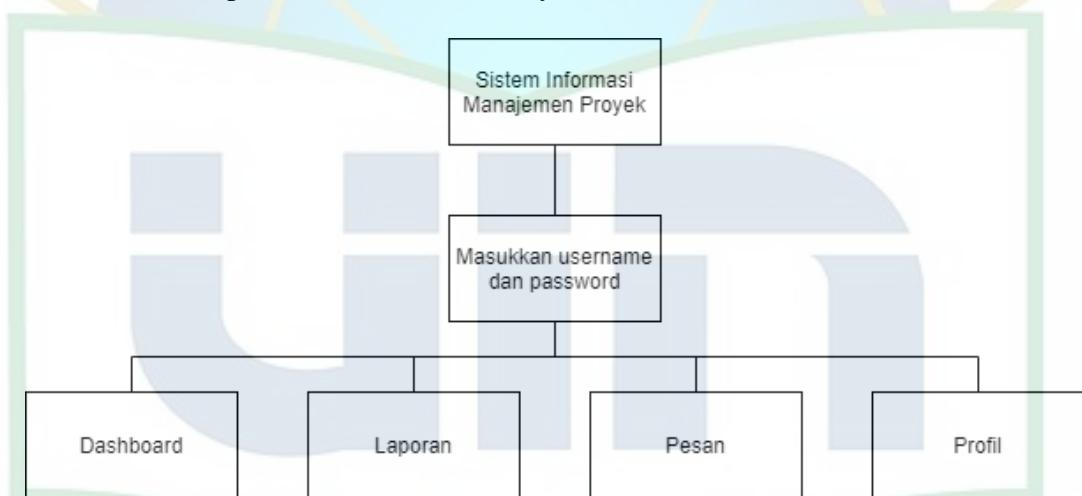


d. Menu Aplikasi *Team Leader* dan Tenaga Ahli



Gambar 4.47 Struktur Menu Aplikasi *Team Leader* dan Tenaga Ahli

e. Menu Aplikasi Administrasi Proyek

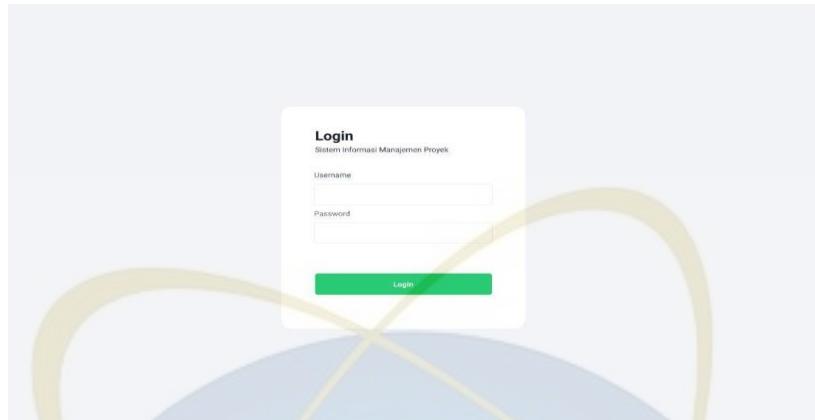


Gambar 4.48 Struktur Menu Aplikasi Administrasi Proyek

4.2.4.2 Perancangan *Interface*

Pada perancangan *interface*, peneliti membagi perancangan *interface* berdasarkan aktor-aktor pada *use case*. Berikut hasil perancangan *interface* sistem informasi manajemen proyek.

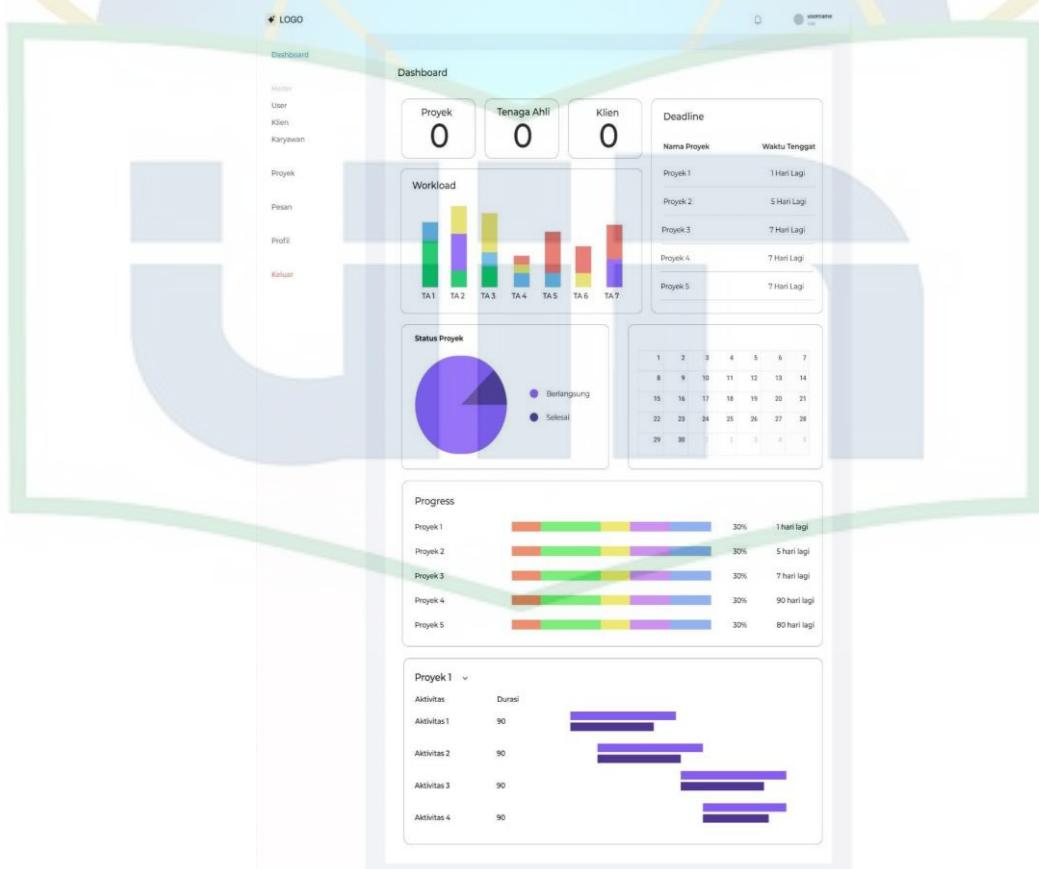
a. Login Interface



Gambar 4.49 *Login Interface*

b. Admin Interface

1) *Dashboard admin*



Gambar 4.50 *Dashboard Admin Interface*

2) User Interface

The screenshot shows a user interface for managing users. At the top right, there is a search bar labeled "Caril..." and a user profile icon. On the left, a sidebar menu lists "Dashboard", "Master", "User", "Klien", "Karyawan", "Proyek", "Pesan", "Profil", and "Keluar". The main content area is titled "User" and contains a table with the following columns: Kode, Nama, Username, Role, and Aksi. The "Aksi" column includes buttons for "Ubah" (Change) and "Hapus" (Delete). The table currently has no data rows.

Gambar 4.51 User Table Interface

The screenshot shows a form for adding a new user. The title of the form is "Form User". It contains four input fields: "Nama", "Username", "Password", and "Role". Each field has a dropdown arrow icon to its right. At the bottom of the form are two buttons: "Batal" (Cancel) on the left and "Simpan" (Save) on the right. The form is set against a background featuring a large, stylized blue and yellow graphic element.

Gambar 4.52 Form Tambah User

Form User

Nama: Nama user

Username: username

Password: password

Role: Role

Batal Simpan

Gambar 4.53 Form Ubah User

3) Karyawan Interface

Karyawan

Kode	Nama	TTL	Alamat	Telp	Aksi
					Ubah Hapus

Tambah Caril...
Ubah Hapus

Gambar 4.54 Tabel Karyawan Interface

Form Karyawan

Nama Lengkap

Tempat Lahir
 Tanggal Lahir

Alamat

Telp

Batal **Simpan**

Karyawan

Gambar 4.55 Form Tambah Karyawan

Form Karyawan

Nama Lengkap
Nama Lengkap

Tempat Lahir
Tempat Lahir Tanggal Lahir
Tanggal Lahir

Alamat
Alamat

Telp
Telepon

Batal **Simpan**

Karyawan

Gambar 4.56 Form Ubah Karyawan

4) Klien Interface

The screenshot shows a web-based application interface. At the top left is a logo with a star icon. To its right are user authentication fields: 'username' and 'password'. Below the header is a navigation menu on the left side with the following items: Dashboard, Master, User, **Klien**, Karyawan, Proyek, Pesan, Profil, and Keluar. The 'Klien' item is highlighted with a blue background. On the right side, there is a main content area titled 'Klien'. It features a green 'Tambah' button and a search bar with the placeholder 'Cari...'. Below these are two buttons: 'Ubah' (blue) and 'Hapus' (red). A table lists client information with columns: Kode, Nama Perusahaan, PIC, Telp, and Aksi. The table has one visible row.

Gambar 4.57 Tabel Klien Interface

This screenshot shows the 'Form Klien' (Client Form) page. The left sidebar has the same navigation menu as the previous screenshot. The main form area contains three input fields: 'Nama Perusahaan' (Business Name), 'Nama PIC' (PIC Name), and 'Telp' (Phone). Below these fields are two buttons: 'Batal' (Cancel) and 'Simpan' (Save). The entire form area is highlighted with a green border.

Gambar 4.58 Form Tambah Klien

Form Klien

Nama Perusahaan
Nama Perusahaan

Nama PIC
Nama PIC

Telp
Telepon

Batal Simpan

Gambar 4.59 Form Ubah Klien

5) Proyek Interface

Proyek

Tambah Car...

Kode	Nama Paket	Tipe	Lokasi	Waktu Mulai	Waktu Selesai	Aksi
						Ubah Hapus

Gambar 4.60 Daftar Proyek Admin

Form Proyek

Program:

Kegiatan: Sub Kegiatan:

Rincian Sub Kegiatan:

Type:

Nama Paket:

Kode Rekening:

Lokasi: Tahun:

Durasi: Waktu Mulai: Waktu Selesai:

Batal **Simpan**

Gambar 4.61 *Form Tambah Proyek Admin*

Form Proyek

Program: Program

Kegiatan: Tempat Lahir | Sub Kegiatan: Tanggal Lahir

Rincian Sub Kegiatan: Alamat

Type: Type

Nama Paket: Nama Paket

Kode Rekening: Kode Rekening

Lokasi: Kota Lokasi | Tahun: Tahun

Durasi: Durasi | Waktu Mulai: Tanggal mulai | Waktu Selesai: Tanggal selesai

Batal | Simpan

Gambar 4.62 Form Ubah Proyek Admin

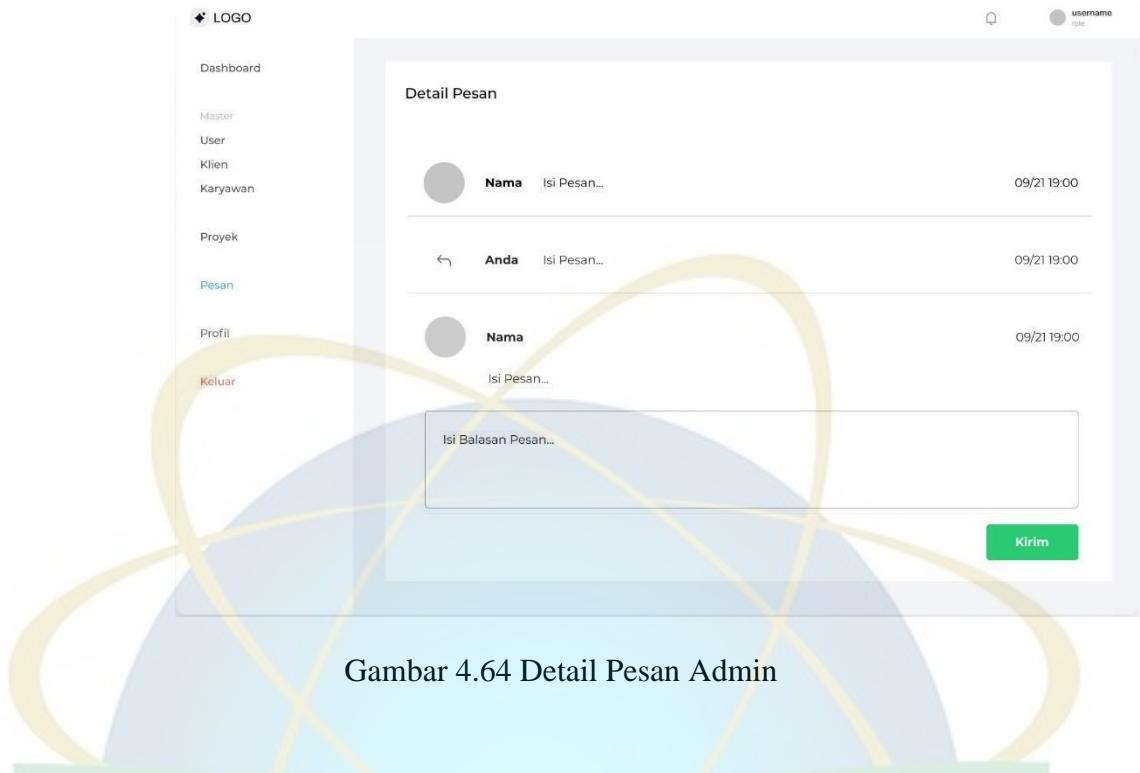
6) Pesan Interface

Pesan

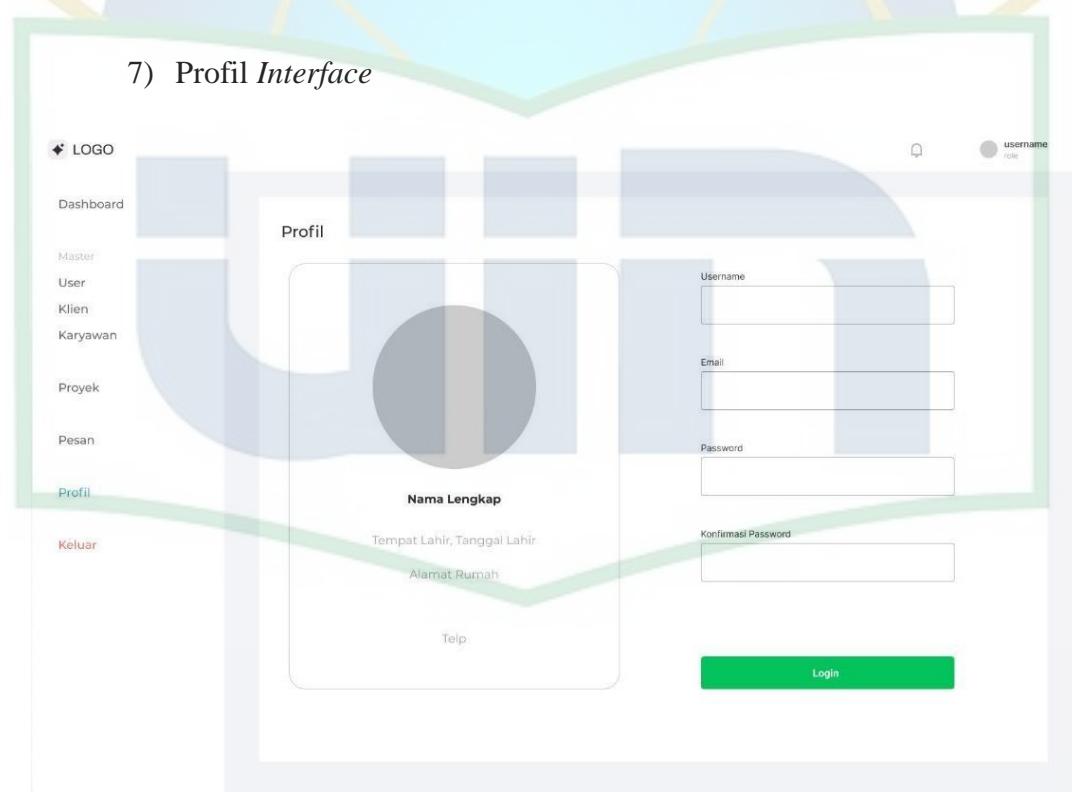
Nama	Tanggal	Aksi
Isi Pesan...	09/21 19:00	Pesan Baru
Isi Pesan...	09/21 19:00	Terkirim
Isi Pesan...	09/21 19:00	Terkirim
Isi Pesan...	09/21 19:00	Terkirim
Isi Pesan...	09/21 19:00	Terkirim

Cari Pengguna

Gambar 4.63 Daftar Pesan Admin



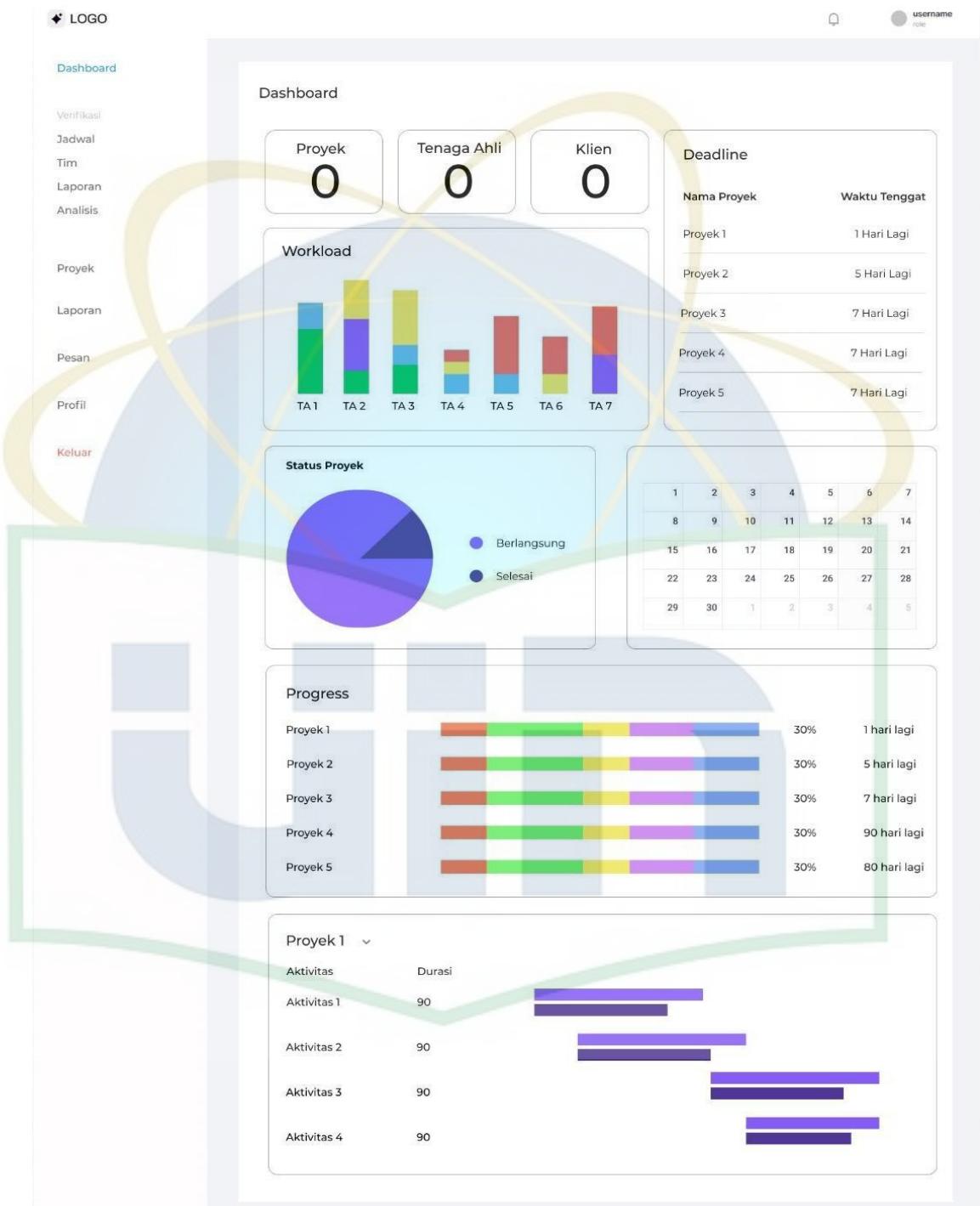
Gambar 4.64 Detail Pesan Admin



Gambar 4.65 Ubah Profil Admin

c. Direktur Teknis *Interface*

1) Dashboard Direktur Teknis



Gambar 4.66 Dashboard Direktur Teknis

2) Verifikasi Jadwal *Interface*

LOGO

Cari...

Kode	Nama Proyek	Total PIC	Normal Time	Manhour	Aksi
					<button>Verifikasi</button> <button>Tolak</button>

Dashboard

Verifikasi

Jadwal

Tim

Laporan

Analisis

Proyek

Laporan

Pesan

Profil

Keluar

Verifikasi Jadwal

Cari...

Gambar 4.67 Verifikasi Jadwal

3) Verifikasi Tim *Interface*

LOGO

Cari...

Kode	Nama Proyek	Aksi
		<button>Verifikasi</button> <button>Tolak</button>

Dashboard

Verifikasi

Tim

Jadwal

Tim

Laporan

Analisis

Proyek

Laporan

Pesan

Profil

Keluar

Verifikasi Tim

Cari...

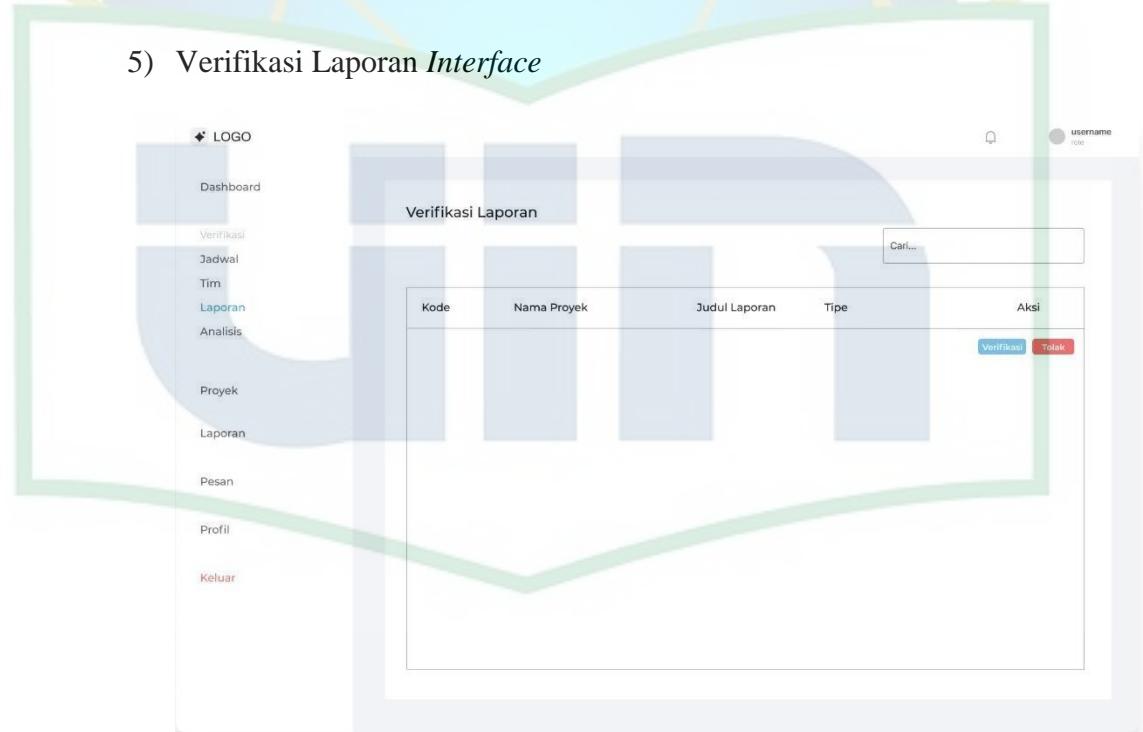
Gambar 4.68 Verifikasi Tim

4) Verifikasi Analisis *Interface*



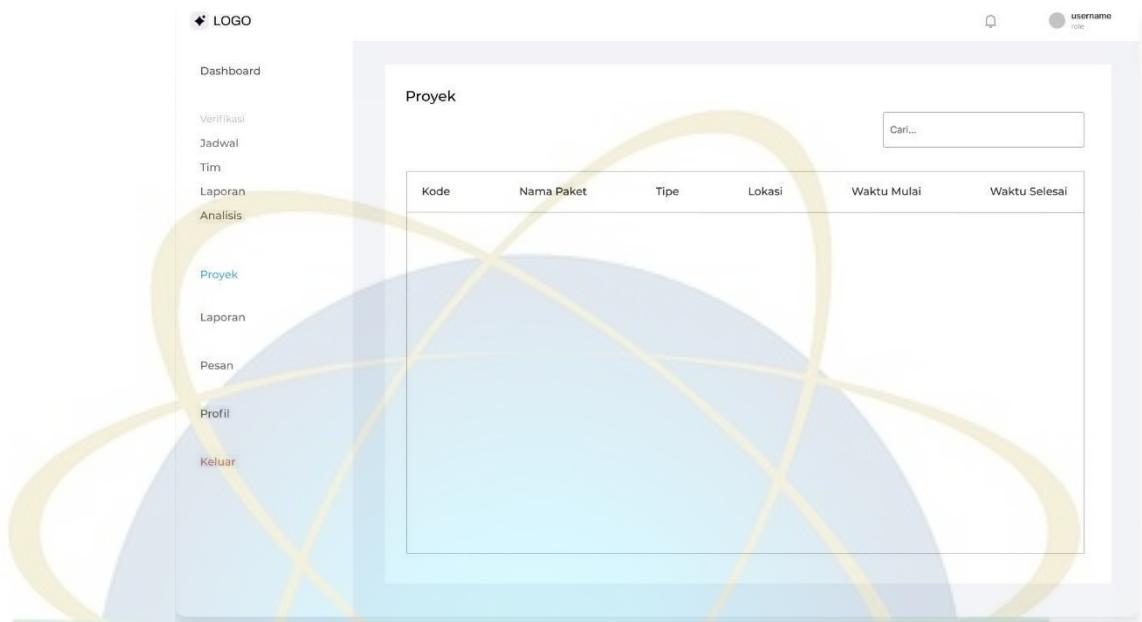
Gambar 4.69 Verifikasi Analisis

5) Verifikasi Laporan *Interface*



Gambar 4.70 Verifikasi Laporan

6) Proyek *Interface*



Gambar 4.71 Daftar Proyek Direktur Teknis



Gambar 4.72 *Form Detail Proyek Direktur Teknis*

Gambar 4.73 *Form Jadwal Proyek Direktur Teknis*

Form Tim Proyek

Posisi	Anggota
Team Leader	Karyawan 1
Posisi 1	Karyawan 2
Posisi 2	Karyawan 3

Gambar 4.74 Form Tim Proyek Direktur Teknis

Jalur Kritis

```

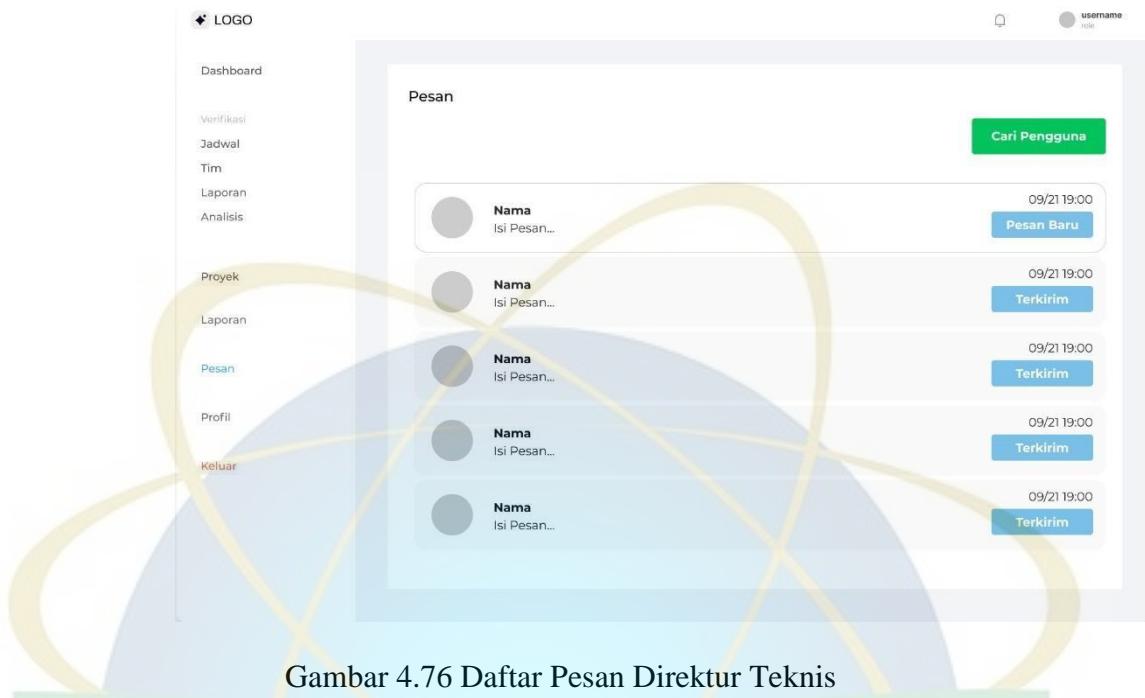
graph TD
    A[Aktivitas 1] --- B[Aktivitas 2]
    B --- C[Aktivitas 3]
  
```

Analisis What If

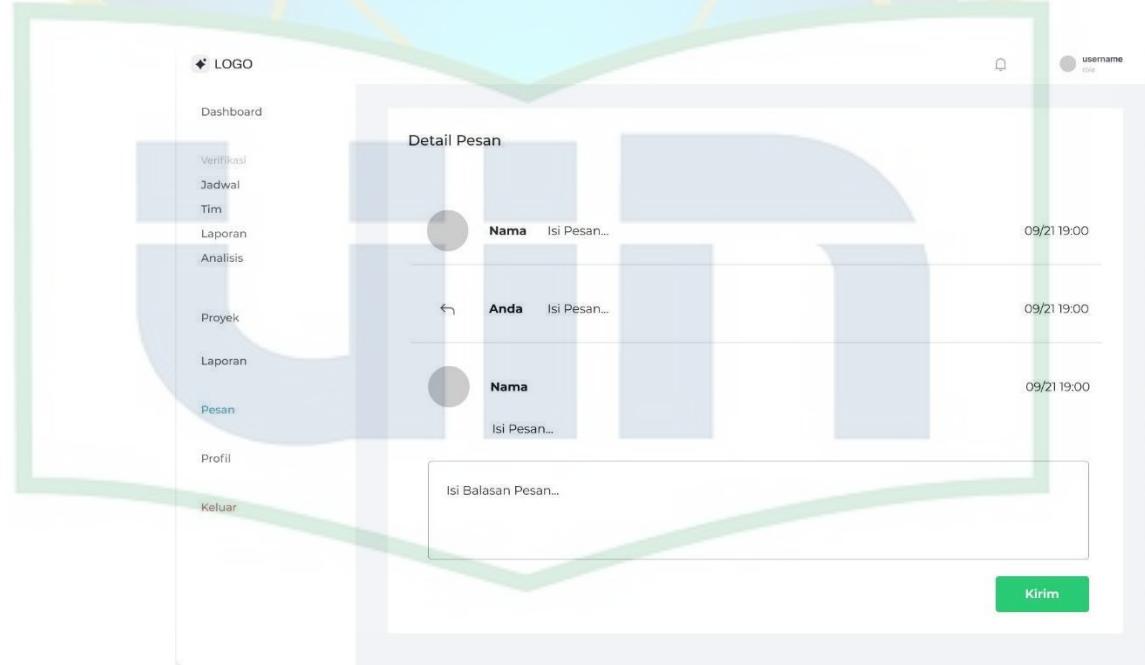
Aktivitas	Waktu Tersisa	Keterlambatan	Tambahan Jam	Tambahan Karyawan
Aktivitas 2	2 Hari	5 Hari	5 Jam	2 Orang
Aktivitas 3	1 Hari			

Gambar 4.75 Form Analisis Proyek Direktur Teknis

7) Pesan Interface

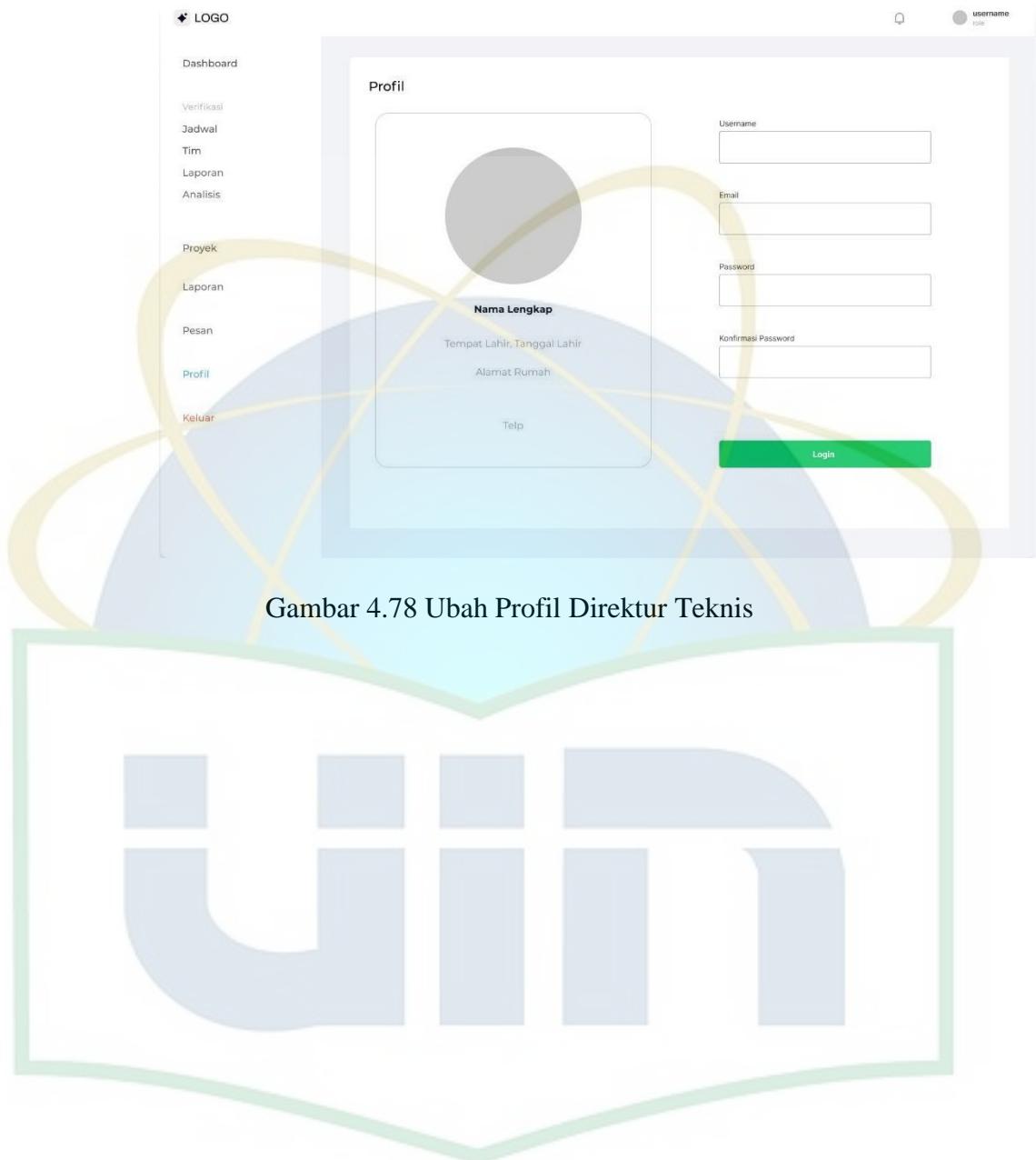


Gambar 4.76 Daftar Pesan Direktur Teknis



Gambar 4.77 Detail Pesan Direktur Teknis

8) Profil Interface



Gambar 4.78 Ubah Profil Direktur Teknis

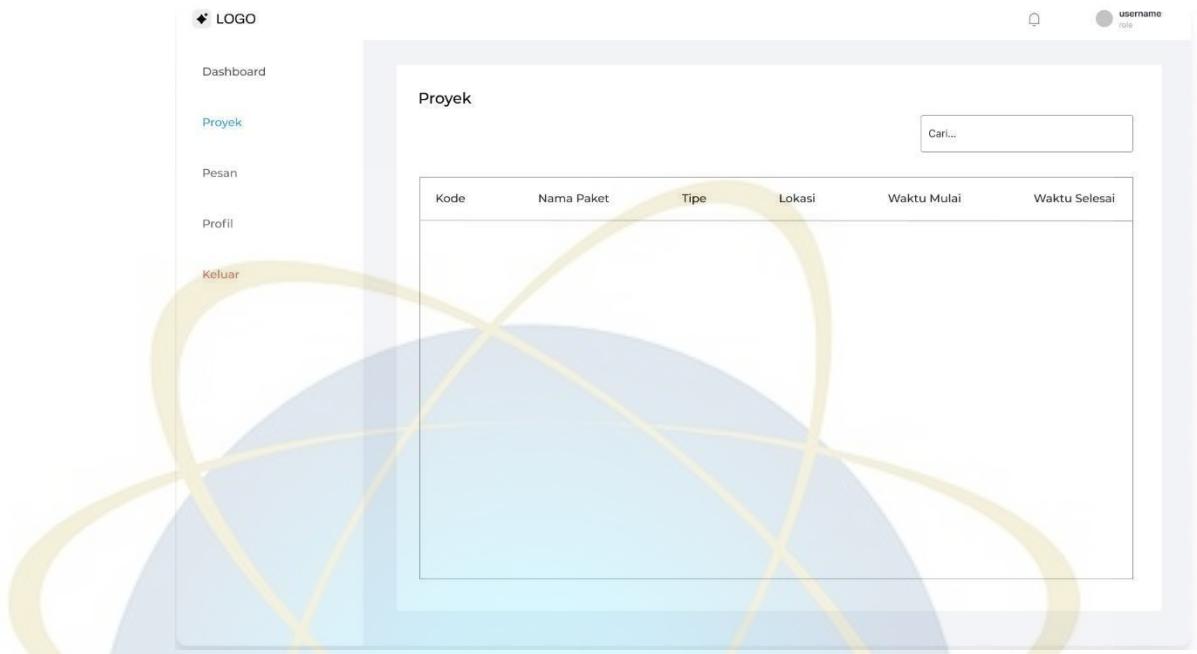
d. Kepala Divisi Proyek *Interface*

1) *Dashboard* Kepala Divisi Proyek



Gambar 4.79 *Dashboard* Kepala Divisi Proyek

2) Proyek *Interface*



Gambar 4.80 Daftar Proyek Kepala Divisi Proyek



Gambar 4.81 *Form Detail Proyek Kepala Divisi Proyek*

Gambar 4.82 *Form Jadwal Proyek Kepala Divisi Proyek*

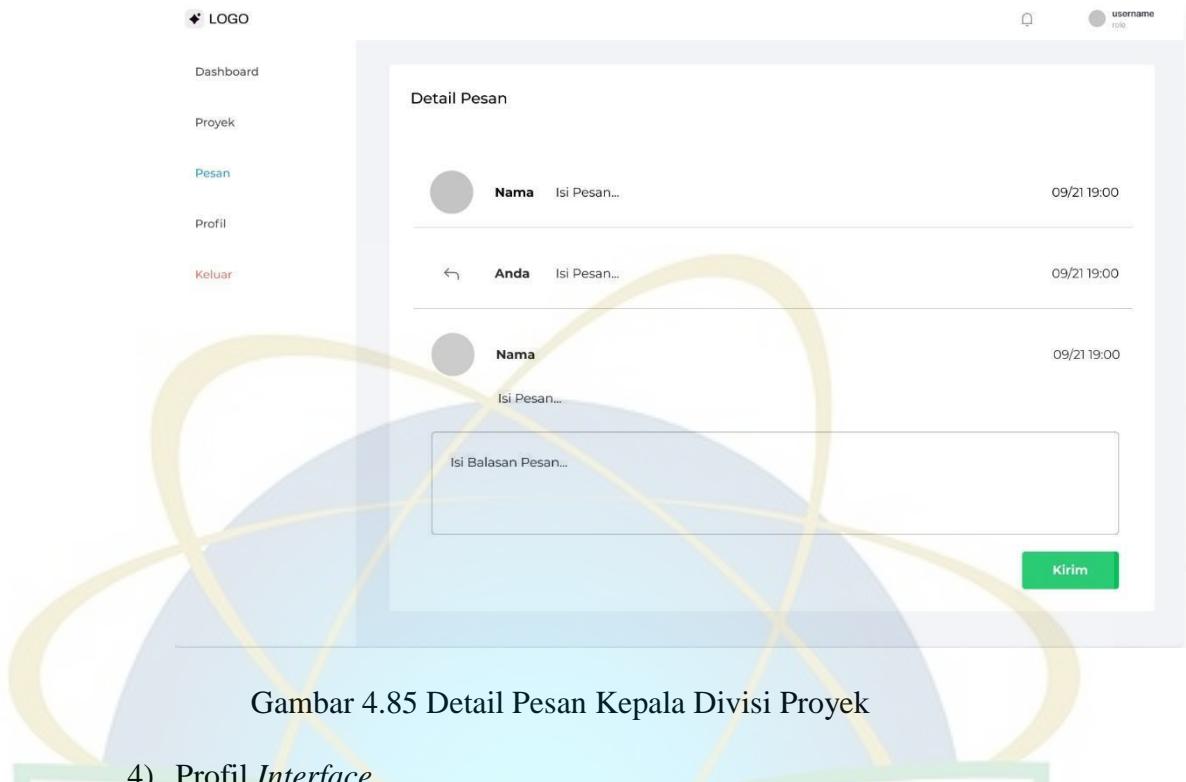
The screenshot shows a user interface for managing project teams. At the top, there's a navigation bar with a logo, a search bar, and user authentication information ('username' and 'role'). Below the navigation is a sidebar with links: Dashboard, Proyek, Pesan, Profil, and Keluar. The main content area has tabs for 'Form Projek', 'Jadwal', 'Tim', and 'Analisis'. Under the 'Tim' tab, there's a section for 'Posisi' (positions) with three tabs: 'Team Leader', 'Posisi 1', and 'Posisi 2'. A dropdown menu under 'Anggota' (members) lists 'Karyawan 1', 'Karyawan 2', and 'Karyawan 3'. At the bottom right is a green 'Simpan' (Save) button, and at the bottom center is a grey 'Batal' (Cancel) button.

Gambar 4.83 Form Tim Proyek Kepala Divisi Proyek

3) Pesan Interface

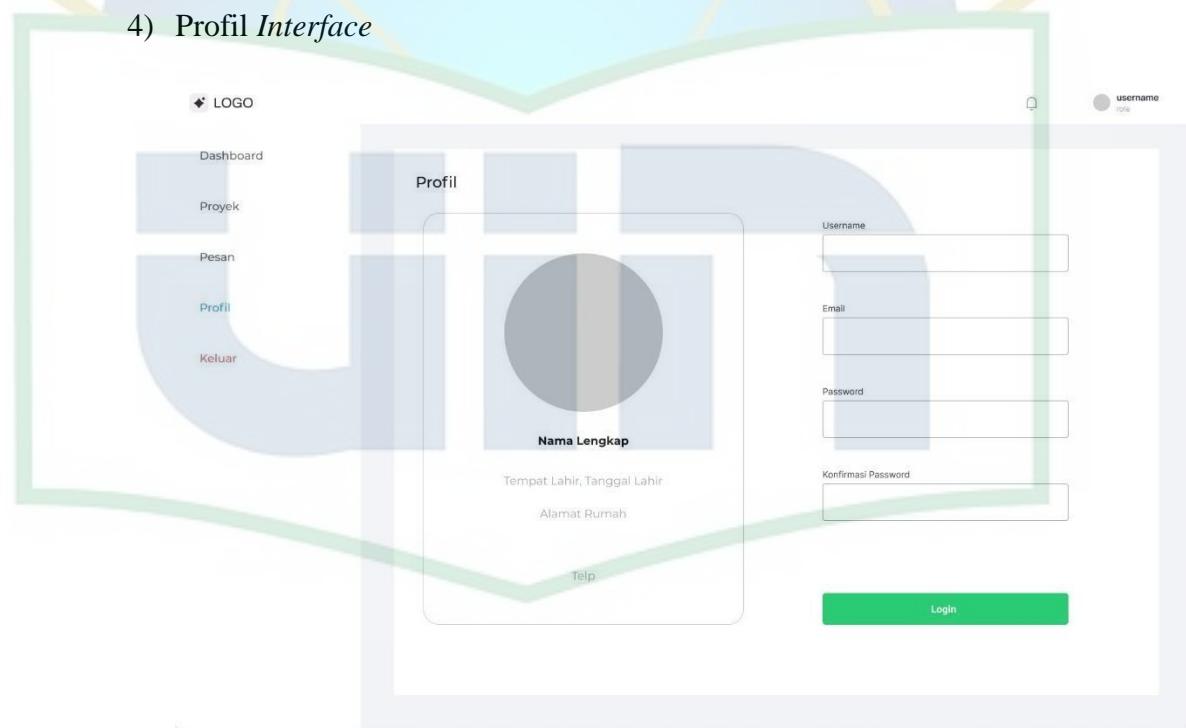
This screenshot displays a messaging interface. On the left is a sidebar with links: Dashboard, Proyek, Pesan, Profil, and Keluar. The main area features a 'Pesaj' (Messages) section with a search bar labeled 'Cari Pengguna'. Below it is a list of messages. Each message item includes a profile picture placeholder, the recipient's name ('Nama'), a placeholder for the message content ('Isi Pesan...'), and a timestamp ('09/21 19:00'). To the right of the messages is a blue button labeled 'Terkirim' (Sent). The entire interface is framed by a light green border.

Gambar 4.84 Daftar Pesan Kepala Divisi Proyek



Gambar 4.85 Detail Pesan Kepala Divisi Proyek

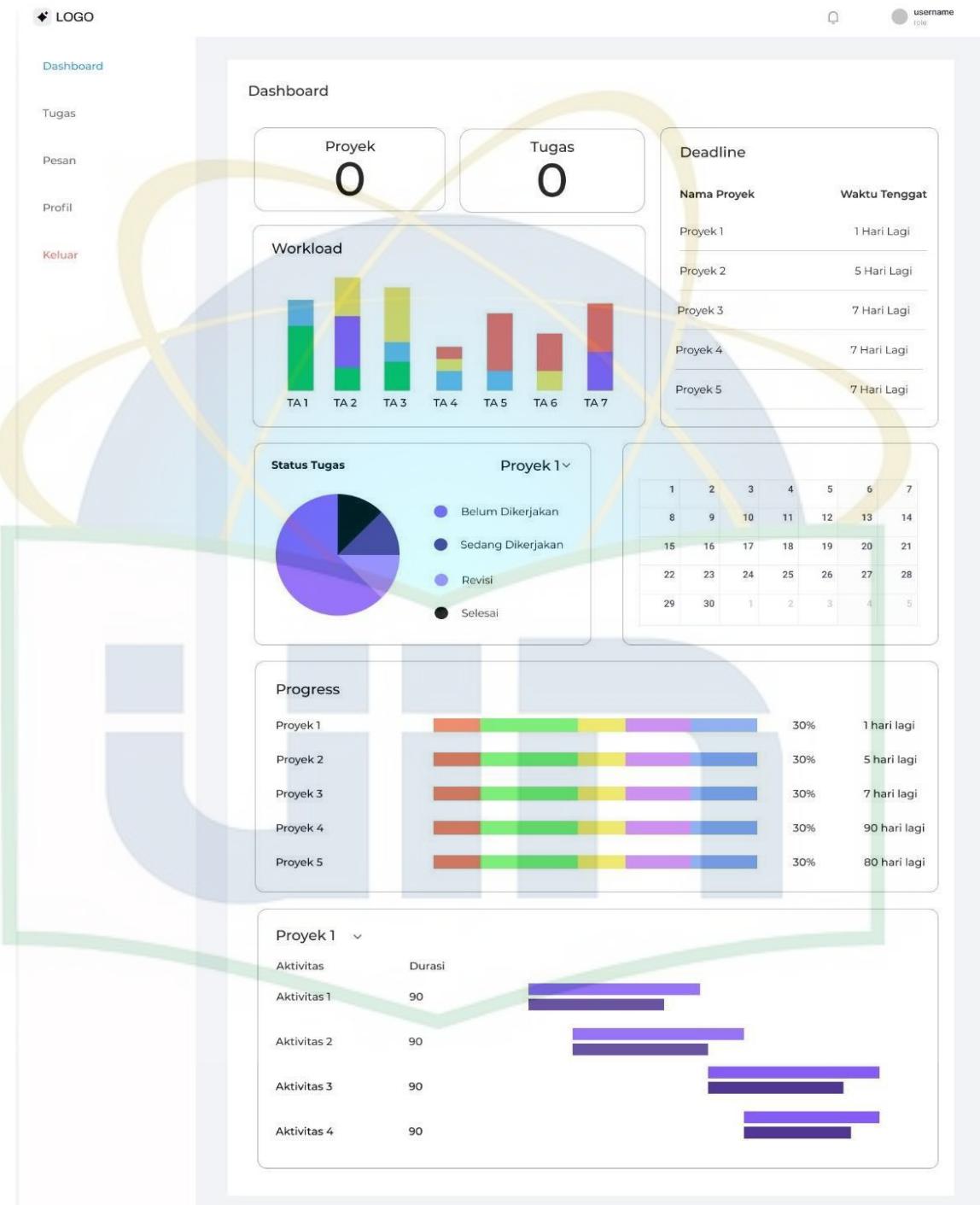
4) Profil Interface



Gambar 4.86 Ubah Profil Kepala Divisi Proyek

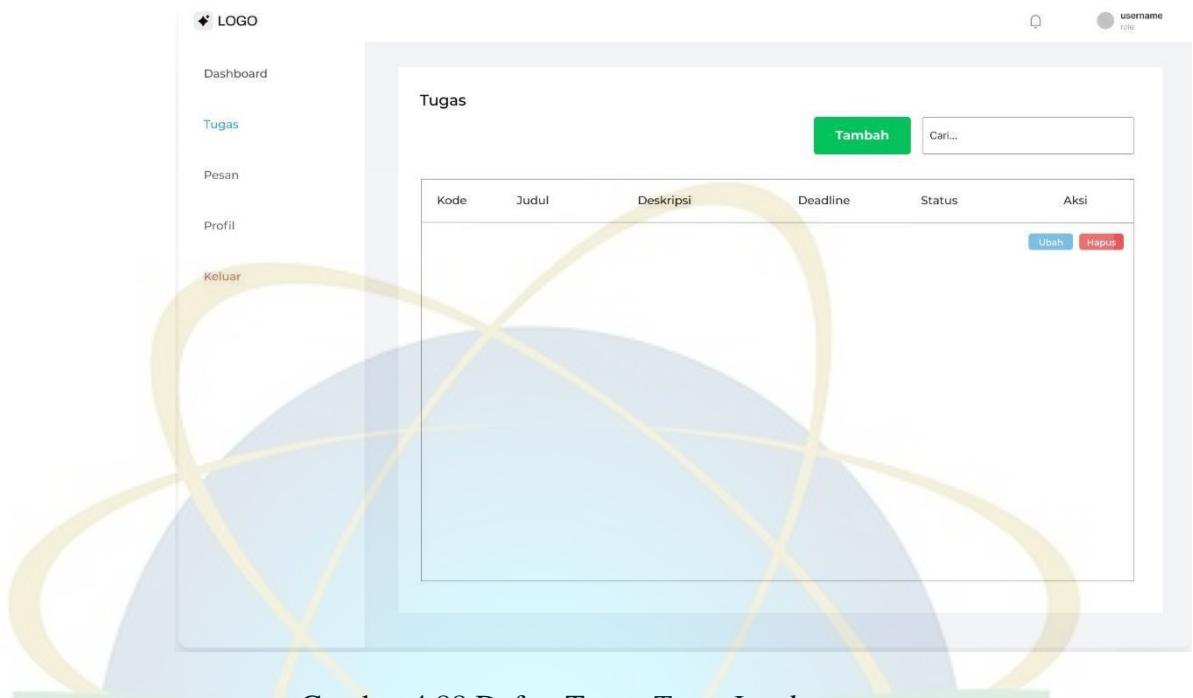
e. Team Leader Interface

1) *Dashboard Team Leader*



Gambar 4.87 *Dashboard Team Leader*

2) Tugas Interface



Gambar 4.88 Daftar Tugas *Team Leader*

The screenshot shows the 'Form Tugas' (Task Form) interface. At the top right, there is a logo, a search bar with placeholder 'Cari...', and user information including 'username' and 'role'. On the left, a vertical sidebar menu lists 'Dashboard', 'Tugas' (selected and highlighted in blue), 'Pesan', 'Profil', and 'Keluar'. The main content area is titled 'Form Tugas' and contains several input fields: 'Proyek' (Nama Proyek dropdown), 'Aktivitas' (Aktivitas dropdown), 'Judul' (Judul Tugas input field), 'Deskripsi' (Deskripsi input field), 'Deadline' (Tanggal Deadline input field), 'PIC' (Nama Karyawan dropdown), 'Jenis' (Jenis dropdown), and a 'Batal' (Cancel) button at the bottom left and a 'Simpan' (Save) button at the bottom right. The entire form is set against a light gray background with a green decorative border on the left side.

Gambar 4.89 Form Detail Tugas

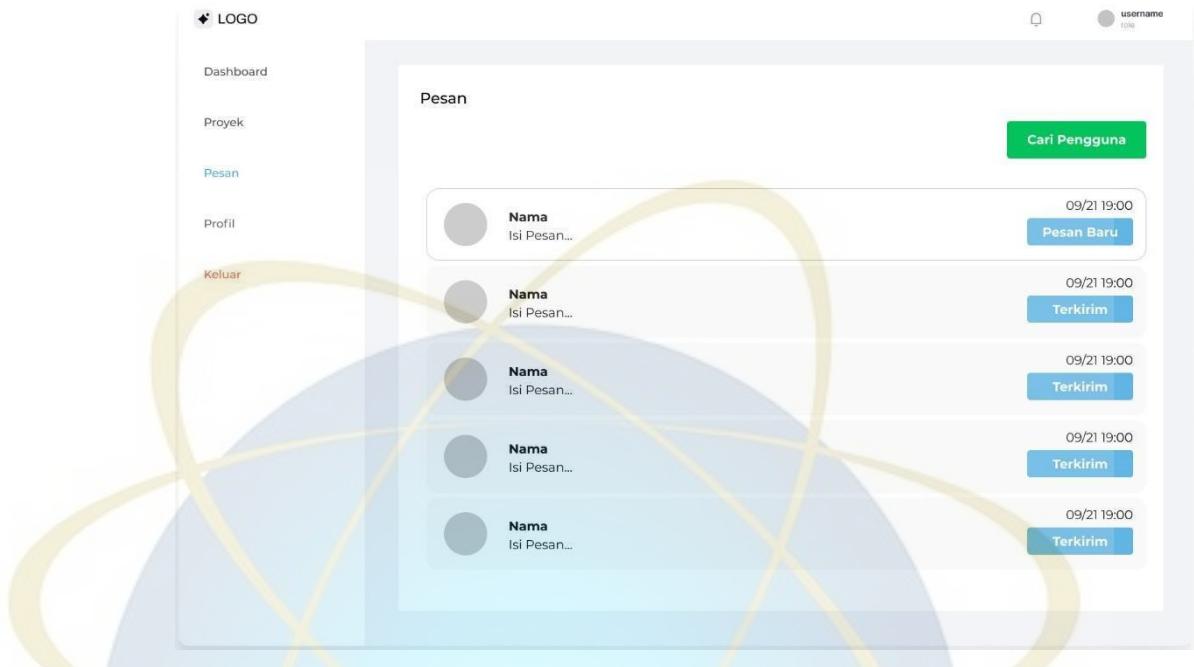
The screenshot shows a user interface for submitting a task. At the top left is a logo icon with a checkmark and the word 'LOGO'. On the top right are icons for a profile (user icon), 'username', and 'role'. The left sidebar has navigation links: 'Dashboard' (selected), 'Tugas' (highlighted in green), 'Pesan', 'Profil', and 'Keluar' (highlighted in red). The main content area is titled 'Form Tugas'. It contains several input fields:

- 'Proyek': A dropdown menu labeled 'Nama Proyek'.
- 'Aktivitas': A dropdown menu labeled 'Aktivitas'.
- 'Judul': An input field labeled 'Judul Tugas'.
- 'Deskripsi': An input field labeled 'Deskripsi'.
- 'Deadline': A date input field labeled 'Tanggal Deadline'.
- 'PIC': A dropdown menu labeled 'Nama Karyawan'.
- 'Jenis': A dropdown menu labeled 'Jenis'.
- 'Hasil': An input field labeled 'Hasil'.

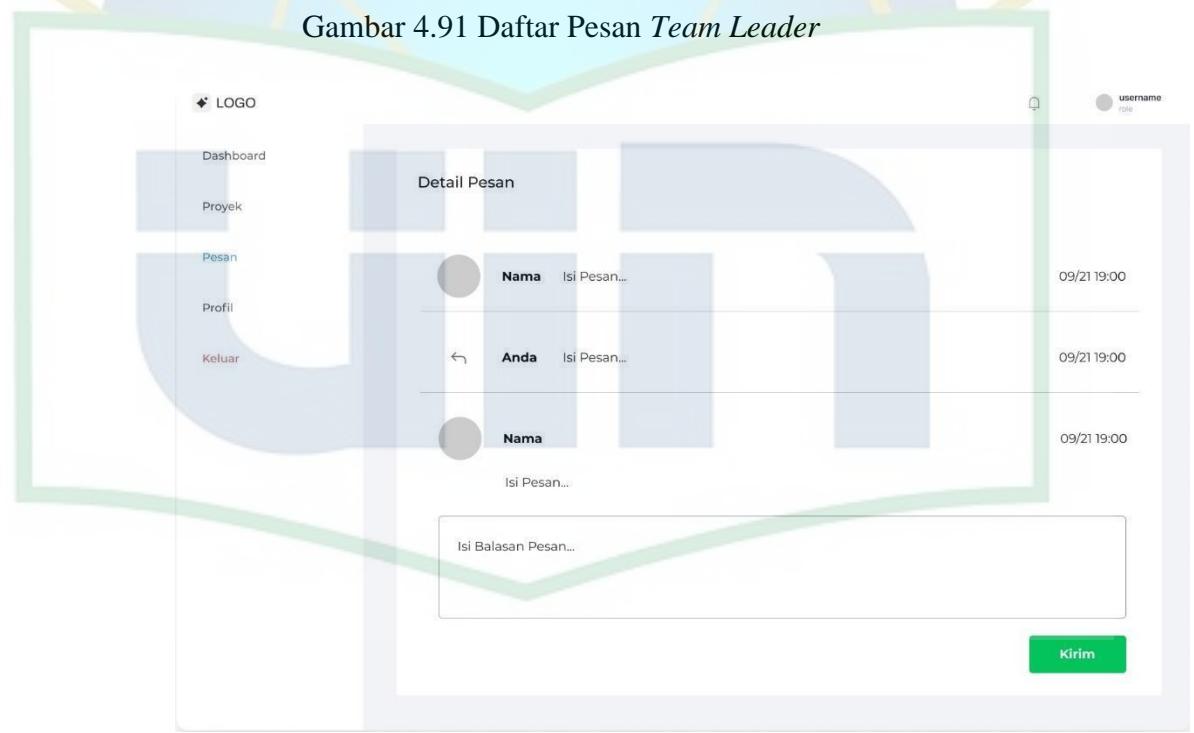
At the bottom are two buttons: a red 'Revisi' button on the left and a green 'Tutup' button on the right.

Gambar 4.90 Tutup Tugas *Interface*

3) Pesan Interface

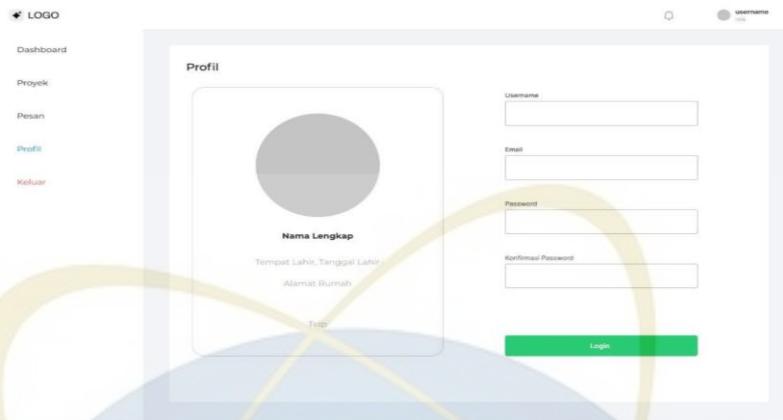


Gambar 4.91 Daftar Pesan Team Leader



Gambar 4.92 Detail Pesan Team Leader

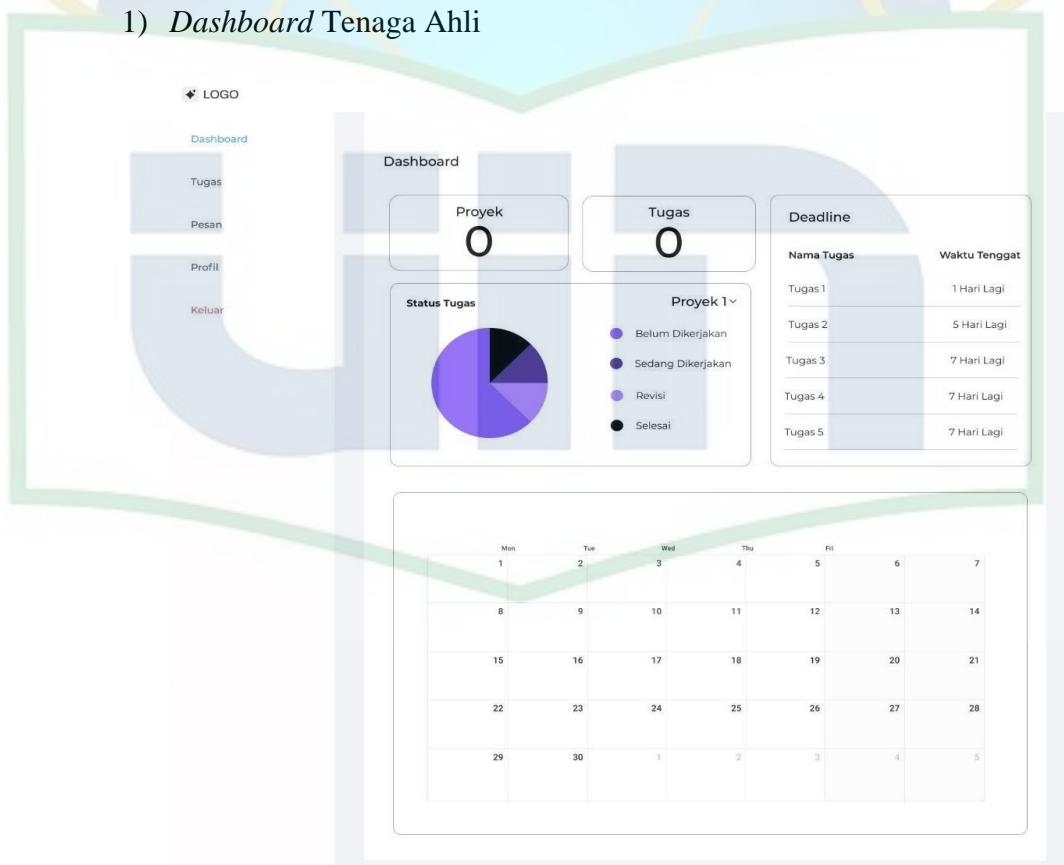
4) Profil Interface



Gambar 4.93 Ubah Profil Team Leader

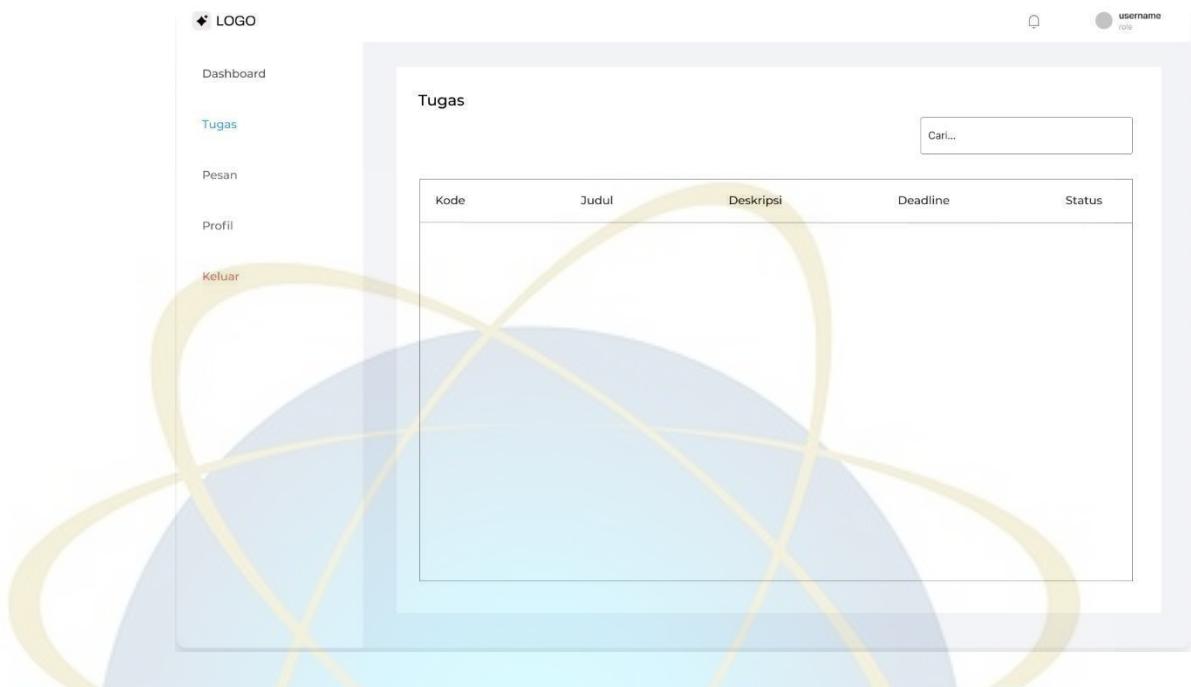
f. Tenaga Ahli Interface

1) Dashboard Tenaga Ahli



Gambar 4.94 Dashboard Tenaga Ahli

2) Tugas Interface



Gambar 4.95 Daftar Tugas Tenaga Ahli

The screenshot shows a task form interface. At the top right is a green button labeled 'Mulai'. On the left is a vertical navigation menu with options: Dashboard, Tugas (highlighted), Pesan, Profil, and Keluar. The main content area is titled 'Form Tugas' and contains several input fields:

- Proyek: Nama Proyek
- Aktivitas: Aktivitas
- Judul: Judul Tugas
- Deadline: Tanggal Deadline
- Deskripsi: Deskripsi
- Jenis: Jenis
- Hasil: A file upload input labeled 'Choose file'
- PIC: Nama Karyawan

The entire interface has a light gray background with a green border around the main form area.

Gambar 4.96 Form Detail Tugas Tenaga Ahli

3) Pesan Interface

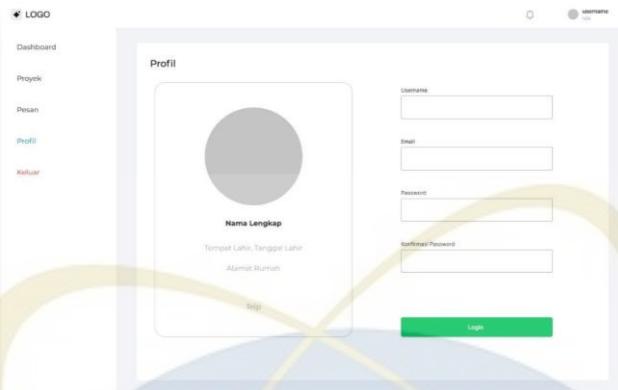


Gambar 4.97 Daftar Pesan Tenaga Ahli



Gambar 4.98 Detail Pesan Tenaga Ahli

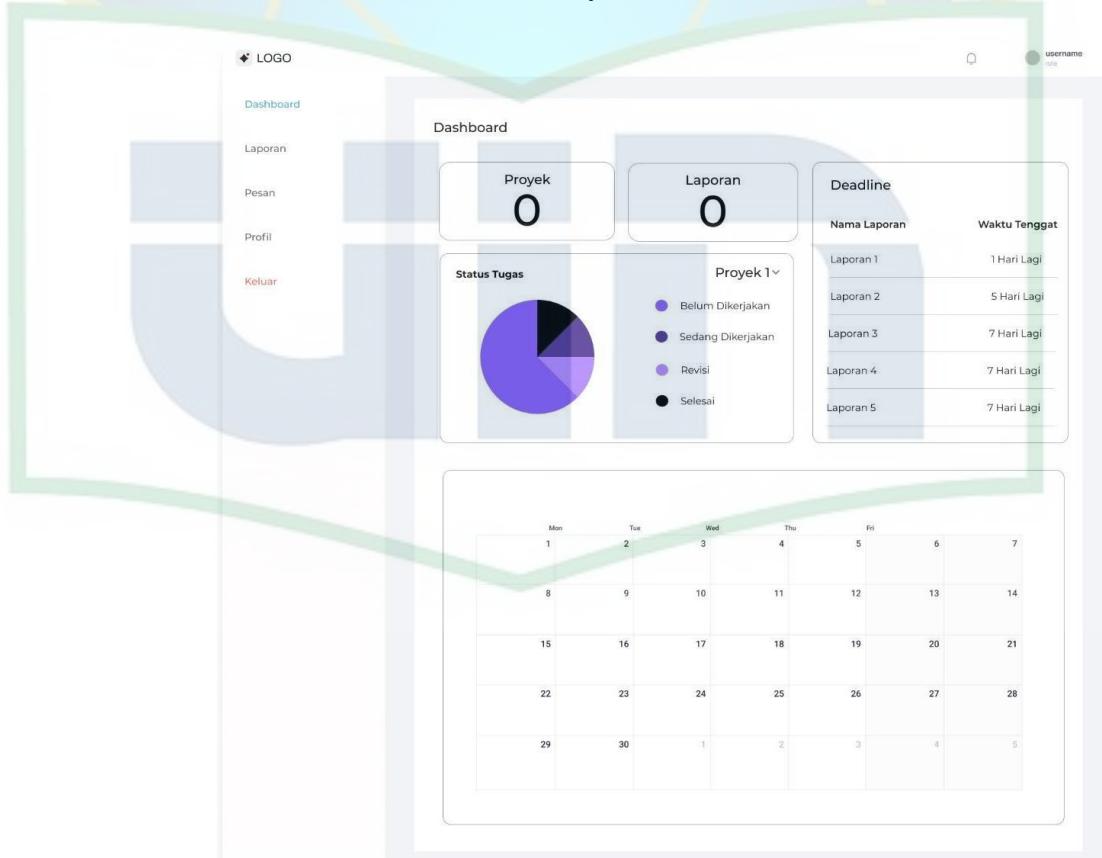
4) Profil Interface



Gambar 4.99 Ubah Profil Tenaga Ahli

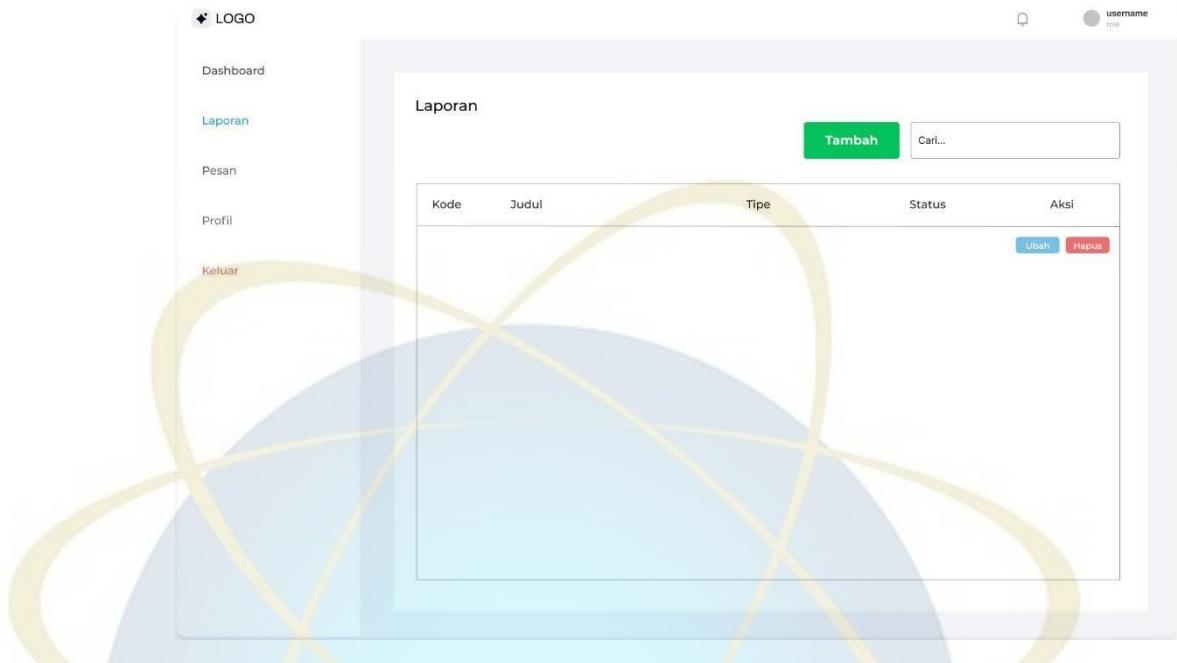
g. Administrasi Proyek Interface

1) Dashboard Administrasi Proyek

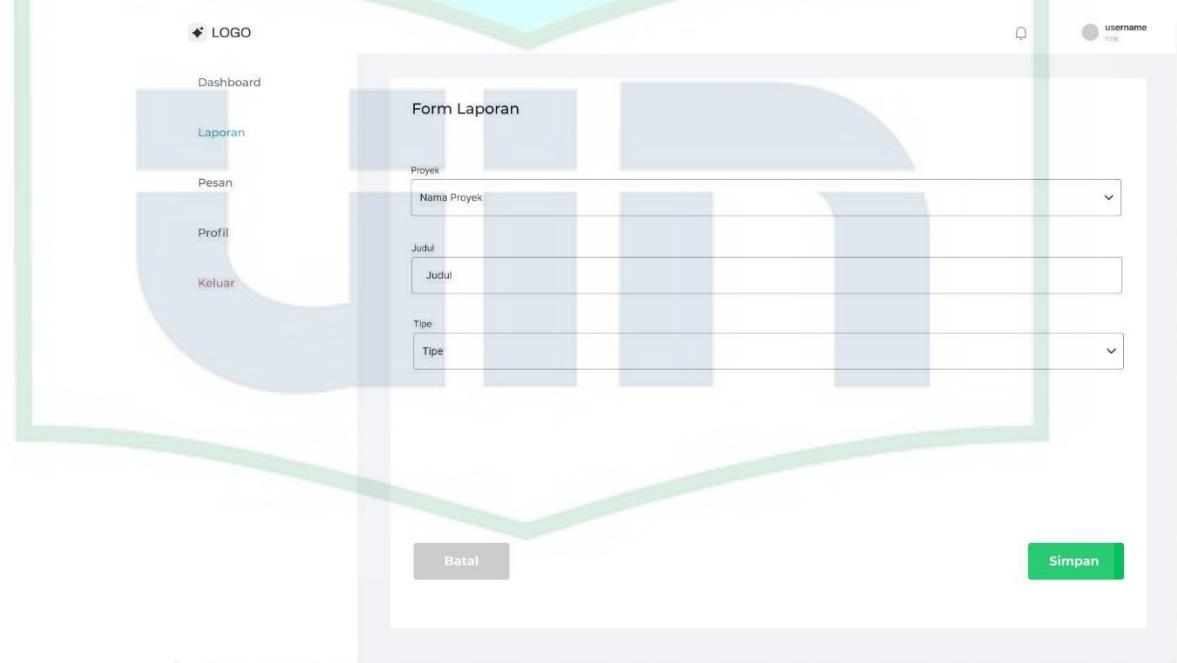


Gambar 4.100 Dashboard Administrasi Proyek

2) Laporan Interface

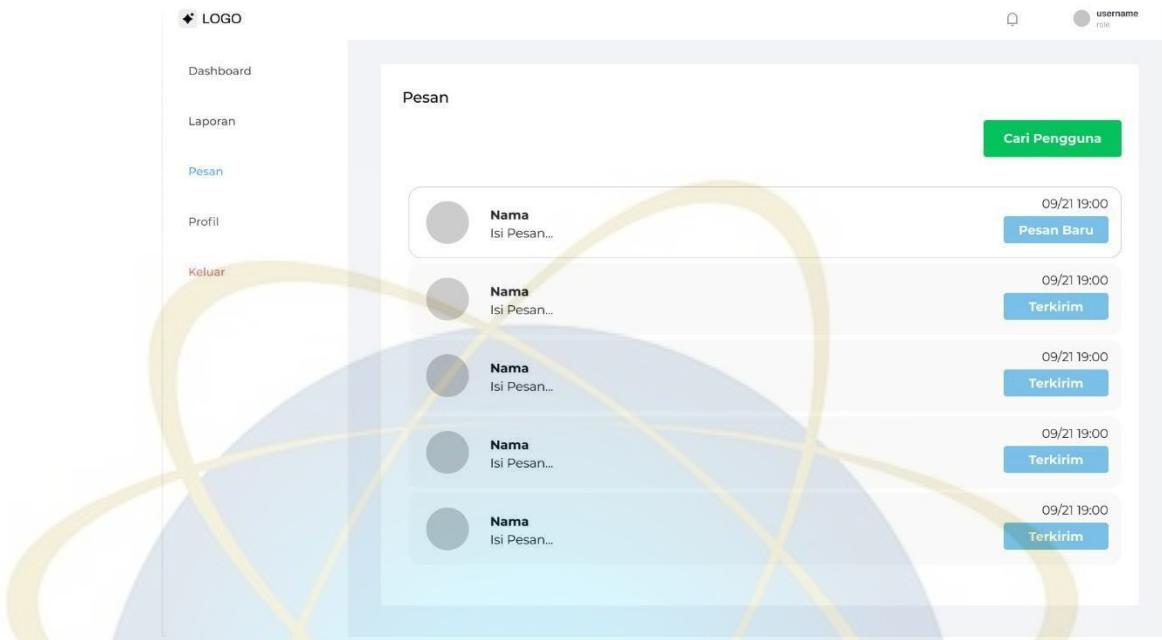


Gambar 4.101 Daftar Laporan



Gambar 4.102 Form Detail Laporan

3) Pesan Interface

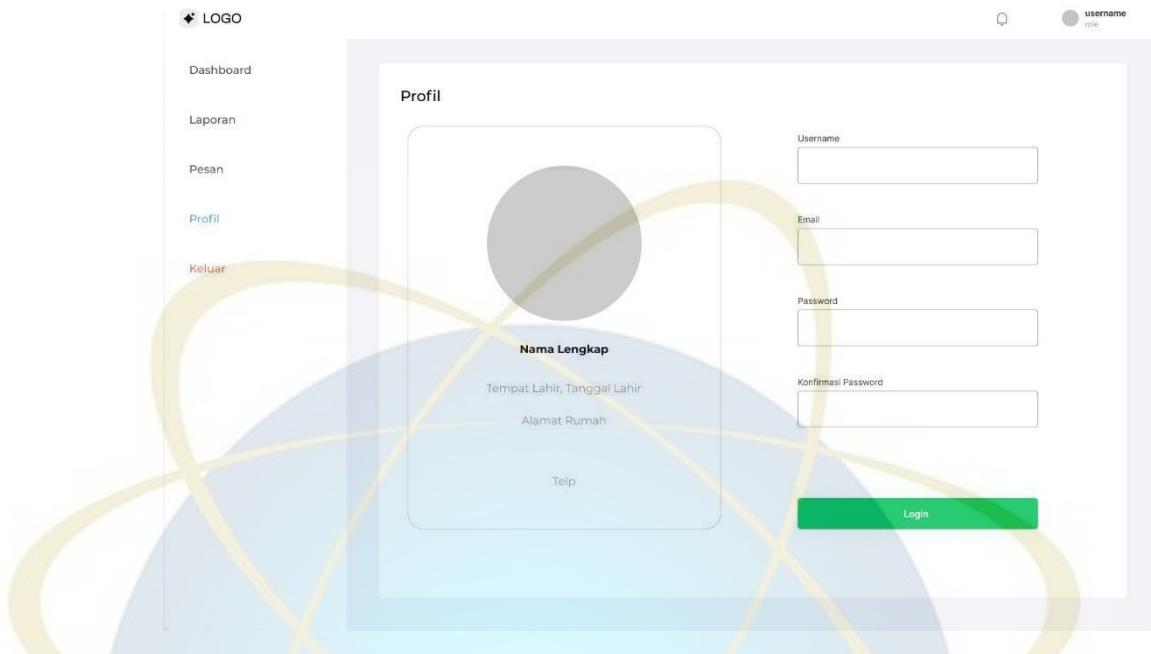


Gambar 4.103 Daftar Pesan Administrasi Proyek



Gambar 4.104 Detail Pesan Administrasi Proyek

4) Profil *Interface*



Gambar 4.105 Ubah Profil Administrasi Proyek

4.3 Implementasi

4.3.1 Pemrograman

Pada tahap ini peneliti melakukan pengkodean untuk membangun Sistem Informasi Manajemen Proyek dengan membagi menjadi dua bagian yaitu *frontend* dan *backend*. Untuk *frontend* menggunakan ReactJS dan untuk *backend* menggunakan bahasa pemrograman utama yaitu PHP dengan *framework* Laravel 8. Untuk basis data menggunakan MySQL untuk menyimpan data, dan PHPMyAdmin untuk mengelola basis data.

4.3.2 Pengujian

Pengujian atau *testing* pada Sistem Informasi Manajemen Proyek menggunakan metode *black box testing*. Proses pengujian yang dilakukan berupa

pemeriksaan sistem dengan menjalankan sistem tersebut dan melihat apakah *output* yang ditampilkan sudah sesuai dengan yang diharapkan. Pengujian dibagi berdasarkan menu yang ada pada sistem, yaitu:

a. *Login*

Tabel 4.47 Hasil Pengujian *Login*

No	Rancangan Proses	Pengguna	Hasil yang diharapkan	Hasil
1	Input <i>Username</i> dan <i>Password</i> , klik login	Semua pengguna	<ul style="list-style-type: none"> - Jika <i>username</i> atau <i>password</i> sesuai, masuk ke halaman <i>dashboard</i> - Jika <i>username</i> atau <i>password</i> tidak sesuai muncul pesan kesalahan dan tetap pada halaman <i>login</i> 	OK

b. *Dashboard*

Tabel 4.48 Hasil Pengujian *Dashboard*

No	Rancangan Proses	Pengguna	Hasil yang diharapkan	Hasil
1	Melihat <i>Dashboard</i>	Semua pengguna	<ul style="list-style-type: none"> - Menampilkan halaman <i>dashboard</i> 	OK

c. *User*

Tabel 4.49 Hasil Pengujian Menu *User*

No	Rancangan Proses	Pengguna	Hasil yang diharapkan	Hasil
1	Klik menu “ <i>User</i> ”	Admin	<ul style="list-style-type: none"> - Menampilkan halaman data <i>user</i> yang ada dalam sistem 	OK

2	Klik tombol “Tambah” pada menu “User”	Admin	<ul style="list-style-type: none"> - Menampilkan form tambah data <i>user</i> 	OK
3	Klik tombol “Simpan” pada form tambah data “User”	Admin	<ul style="list-style-type: none"> - Jika data valid, menyimpan data <i>user</i> dan menampilkan pesan berhasil - Jika data tidak valid, menampilkan pesan kesalahan 	OK
4	Klik tombol “Batal” pada form tambah data “User”	Admin	<ul style="list-style-type: none"> - Kembali ke halaman data <i>user</i> 	OK
5	Klik tombol “Ubah” pada menu “User”	Admin	<ul style="list-style-type: none"> - Menampilkan form ubah data <i>user</i> 	OK
6	Klik tombol “Simpan” pada form tambah ubah “User”	Admin	<ul style="list-style-type: none"> - Jika data valid, menyimpan data perubahan <i>user</i> dan menampilkan pesan berhasil - Jika data tidak valid, menampilkan pesan kesalahan 	OK
7	Klik tombol “Batal” pada form tambah ubah “User”	Admin	<ul style="list-style-type: none"> - Kembali ke halaman data <i>user</i> 	OK
6	Klik Tombol “Hapus” pada menu “User”	Admin	<ul style="list-style-type: none"> - Menampilkan pesan konfirmasi 	OK

7	Klik tombol “Iya” pada pesan konfirmasi hapus	Admin	- Menghapus data <i>user</i> dan menampilkan pesan berhasil	OK
8	Klik tombol “Batal” pada pesan konfirmasi hapus	Admin	- Kembali ke halaman data <i>user</i>	OK

d. Karyawan

Tabel 4.50 Hasil Pengujian Menu Karyawan

No	Rancangan Proses	Pengguna	Hasil yang diharapkan	Hasil
1	Klik menu “Karyawan”	Admin	- Menampilkan halaman data karyawan	OK
2	Klik tombol “Tambah” pada menu “Karyawan”	Admin	- Menampilkan <i>form</i> tambah data karyawan	OK
3	Klik tombol “Simpan” pada <i>form</i> tambah data “Karyawan”	Admin	<ul style="list-style-type: none"> - Jika data valid, menyimpan data karyawan dan menampilkan pesan berhasil - Jika data tidak valid, menampilkan pesan kesalahan 	OK
4	Klik tombol “Batal” pada <i>form</i> tambah data “Karyawan”	Admin	- Kembali ke halaman data karyawan	OK
5	Klik tombol “Ubah” pada menu “Karyawan”	Admin	- Menampilkan <i>form</i> ubah data karyawan	OK

6	Klik tombol “Simpan” pada <i>form</i> tambah ubah “Karyawan”	Admin	<ul style="list-style-type: none"> - Jika data valid, menyimpan data perubahan karyawan dan menampilkan pesan berhasil - Jika data tidak valid, menampilkan pesan kesalahan 	OK
7	Klik tombol “Batal” pada <i>form</i> tambah ubah “Karyawan”	Admin	<ul style="list-style-type: none"> - Kembali ke halaman data karyawan 	OK
6	Klik Tombol “Hapus” pada menu “Karyawan”	Admin	<ul style="list-style-type: none"> - Menampilkan pesan konfirmasi 	OK
7	Klik tombol “Iya” pada pesan konfirmasi hapus	Admin	<ul style="list-style-type: none"> - Menghapus data karyawan dan menampilkan pesan berhasil 	OK
8	Klik tombol “Batal” pada pesan konfirmasi hapus	Admin	<ul style="list-style-type: none"> - Kembali ke halaman data karyawan 	OK

e. Klien

Tabel 4.51 Hasil Pengujian Menu Klien

No	Rancangan Proses	Pengguna	Hasil yang diharapkan	Hasil
1	Klik menu “Klien”	Admin	<ul style="list-style-type: none"> - Menampilkan halaman data klien yang 	OK
2	Klik tombol “Tambah” pada menu “Klien”	Admin	<ul style="list-style-type: none"> - Menampilkan <i>form</i> tambah data klien 	OK

3	Klik tombol “Simpan” pada <i>form</i> tambah data “Klien”	Admin	<ul style="list-style-type: none"> - Jika data valid, menyimpan data klien dan menampilkan pesan berhasil - Jika data tidak valid, menampilkan pesan kesalahan 	OK
4	Klik tombol “Batal” pada <i>form</i> tambah data “Klien”	Admin	<ul style="list-style-type: none"> - Kembali ke halaman data klien 	OK
5	Klik tombol “Ubah” pada menu “Klien”	Admin	<ul style="list-style-type: none"> - Menampilkan <i>form</i> ubah data klien 	OK
6	Klik tombol “Simpan” pada <i>form</i> tambah ubah “Klien”	Admin	<ul style="list-style-type: none"> - Jika data valid, menyimpan data perubahan klien dan menampilkan pesan berhasil - Jika data tidak valid, menampilkan pesan kesalahan 	OK
7	Klik tombol “Batal” pada <i>form</i> tambah ubah “Klien”	Admin	<ul style="list-style-type: none"> - Kembali ke halaman data klien 	OK
6	Klik Tombol “Hapus” pada menu “Klien”	Admin	<ul style="list-style-type: none"> - Menampilkan pesan konfirmasi 	OK
7	Klik tombol “Iya” pada pesan konfirmasi hapus	Admin	<ul style="list-style-type: none"> - Menghapus data klien dan menampilkan pesan berhasil 	OK

8	Klik tombol “Batal” pada pesan konfirmasi hapus	Admin	- Kembali ke halaman data klien	OK
---	---	-------	---------------------------------	----

f. Proyek

Tabel 4.52 Hasil Pengujian Proyek

No	Rancangan Proses	Pengguna	Hasil yang diharapkan	Hasil
1	Klik menu “Proyek”	Admin	- Menampilkan halaman data proyek	OK
2	Klik tombol “Tambah” pada menu “Proyek”	Admin	- Menampilkan <i>form</i> tambah data proyek	OK
3	Klik tombol “Simpan” pada <i>form</i> tambah data “Proyek”	Admin	<ul style="list-style-type: none"> - Jika data valid, menyimpan data proyek dan menampilkan pesan berhasil - Jika data tidak valid, menampilkan pesan kesalahan 	OK
4	Klik tombol “Batal” pada <i>form</i> tambah data “Proyek”	Admin	- Kembali ke halaman data proyek	OK
5	Klik tombol “Ubah” pada menu “Proyek”	Admin	- Menampilkan <i>form</i> ubah data proyek	OK
6	Klik tombol “Simpan” pada <i>form</i> tambah ubah “Proyek”	Admin	- Jika data valid, menyimpan data perubahan proyek dan menampilkan pesan berhasil	OK

			<ul style="list-style-type: none"> - Jika data tidak valid, menampilkan pesan kesalahan 	
7	Klik tombol “Batal” pada <i>form</i> tambah ubah “Proyek”	Admin	<ul style="list-style-type: none"> - Kembali ke halaman data proyek 	OK
6	Klik Tombol “Hapus” pada menu “Proyek”	Admin	<ul style="list-style-type: none"> - Menampilkan pesan konfirmasi 	OK
7	Klik tombol “Iya” pada pesan konfirmasi hapus	Admin	<ul style="list-style-type: none"> - Menghapus data proyek dan menampilkan pesan berhasil 	OK
8	Klik tombol “Batal” pada pesan konfirmasi hapus	Admin	<ul style="list-style-type: none"> - Kembali ke halaman data proyek 	OK

g. Jadwal

Tabel 4.53 Hasil Pengujian Menu Jadwal

No	Rancangan Proses	Pengguna	Hasil yang diharapkan	Hasil
1	Klik proyek yang ingin dikelola jadwalnya pada halaman proyek	Kepala Divisi Proyek	<ul style="list-style-type: none"> - Menampilkan halaman detail proyek 	OK
2	Klik tab “Jadwal”	Kepala Divisi Proyek	<ul style="list-style-type: none"> - Menampilkan halaman data jadwal 	OK

3	Klik tombol “Tambah” pada menu “Jadwal”	Kepala Divisi Proyek	- Menampilkan form tambah data jadwal	OK
4	Klik tombol “Tambah Aktivitas” pada form tambah jadwal	Kepala Divisi Proyek	- Menampilkan <i>form</i> tambah aktivitas	OK
5	Klik tombol “Simpan” pada <i>form</i> tambah data aktivitas	Kepala Divisi Proyek	- Jika data valid, menyimpan data aktivitas dan menampilkan pesan berhasil - Jika data tidak valid, menampilkan pesan kesalahan	OK
6	Klik tombol “Simpan” pada <i>form</i> tambah data Jadwal	Kepala Divisi Proyek	- Menampilkan halaman jadwal	OK
7	Klik tombol “Ubah” pada menu “Jadwal”	Kepala Divisi Proyek	- Menampilkan <i>form</i> tambah ubah jadwal	OK
8	Klik tombol “Ubah” pada tabel aktivitas	Kepala Divisi Proyek	- Menampilkan <i>form</i> ubah aktivitas	OK
9	Klik tombol “Simpan” pada <i>form</i> tambah ubah aktivitas	Kepala Divisi Proyek	- Jika data valid, menyimpan data aktivitas dan menampilkan pesan berhasil - Jika data tidak valid, menampilkan pesan kesalahan	OK

10	Klik tombol “Hapus” pada menu “Jadwal”	Kepala Divisi Proyek	- Menampilkan pesan konfirmasi	OK
11	Klik tombol “Iya” pada pesan konfirmasi hapus	Kepala Divisi Proyek	- Menghapus data jadwal dan menampilkan pesan berhasil	OK
12	Klik tombol “Batal” pada pesan konfirmasi hapus	Kepala Divisi Proyek	- Kembali ke halaman data jadwal	OK

h. Tim

Tabel 4.54 Hasil Pengujian Menu Tim

No	Rancangan Proses	Pengguna	Hasil yang diharapkan	Hasil
1	Klik proyek yang ingin dikelola timnya pada halaman proyek	Kepala Divisi Proyek	- Menampilkan halaman detail proyek	OK
2	Klik tab “Tim”	Kepala Divisi Proyek	- Menampilkan halaman data tim	OK
3	Klik tombol “Tambah” pada menu “Tim”	Kepala Divisi Proyek	- Menampilkan <i>form</i> tambah data tim	OK
4	Klik tombol “Tambah Anggota” pada <i>form</i> tambah data aktivitas	Kepala Divisi Proyek	- Menambahkan anggota tim proyek	OK

6	Klik tombol “Simpan” pada <i>form</i> tambah data Tim	Kepala Divisi Proyek	- Menampilkan halaman data tim	OK
7	Klik tombol “Ubah” pada menu “Tim”	Kepala Divisi Proyek	- Menampilkan <i>form</i> ubah tim	OK
8	Klik tombol “Ubah” pada tabel aktivitas	Kepala Divisi Proyek	- Menampilkan <i>form</i> ubah anggota tim	OK
9	Klik tombol “Tambah Anggota” pada <i>form</i> tambah data aktivitas	Kepala Divisi Proyek	- Menambahkan anggota tim proyek	OK
10	Klik tombol “Hapus” pada menu “Tim”	Kepala Divisi Proyek	- Menampilkan pesan konfirmasi	OK
11	Klik tombol “Iya” pada pesan konfirmasi hapus	Kepala Divisi Proyek	- Menghapus data tim dan menampilkan pesan berhasil	OK
12	Klik tombol “Batal” pada pesan konfirmasi hapus	Kepala Divisi Proyek	- Kembali ke halaman data jadwal	OK

i. Analisis

Tabel 4.55 Hasil Pengujian Menu Analisis

No	Rancangan Proses	Pengguna	Hasil yang diharapkan	Hasil
1	Klik proyek yang ingin dianalisis pada halaman proyek	Kepala Divisi Proyek	- Menampilkan halaman detail proyek	OK
2	Klik tab “Analisis”	Kepala Divisi Proyek	- Menampilkan halaman data analisis	OK
3	Klik tombol “Buat Jalur Kritis” pada menu analisis	Kepala Divisi Proyek	- Menampilkan hasil perhitungan <i>critical path method</i> berupa jalur kritis pada proyek	OK
4	Klik tombol “Analisis” pada tabel status kegiatan	Kepala Divisi Proyek	- Menampilkan <i>form</i> analisis keterlambatan	OK
6	Klik tombol “Simpan” pada <i>form</i> analisis keterlambatan	Kepala Divisi Proyek	<ul style="list-style-type: none"> - Jika data valid, melakukan perhitungan dengan <i>what if analysis</i> dan menampilkan pesan berhasil - Jika data tidak valid, menampilkan pesan kesalahan 	OK
7	Klik tombol “Lihat Hasil Analisis” pada tabel status kegiatan	Kepala Divisi Proyek	- Menampilkan <i>form</i> hasil analisis keterlambatan pada kegiatan yang dipilih	OK

8	Klik tombol “Kirim” form hasil analisis keterlambatan	Kepala Divisi Proyek	- Mengirimkan hasil analisis dan menampilkan pesan berhasil	OK
---	---	----------------------	---	----

j. Verifikasi Jadwal

Tabel 4.56 Hasil Pengujian Menu Verifikasi Jadwal

No	Rancangan Proses	Pengguna	Hasil yang diharapkan	Hasil
1	Klik menu Verifikasi “Jadwal”	Direktur Teknis	- Menampilkan halaman verifikasi jadwal	OK
2	Klik tombol “Verifikasi” pada menu verifikasi jadwal	Direktur Teknis	- Menampilkan form verifikasi jadwal	OK
3	Klik tombol “Setuju” pada <i>form</i> verifikasi jadwal	Direktur Teknis	<ul style="list-style-type: none"> - Jika data valid, data berhasil diverifikasi dan menampilkan pesan berhasil - Jika data tidak valid, menampilkan pesan kesalahan 	OK
4	Klik tombol “Batal” pada <i>form</i> verifikasi jadwal	Direktur Teknis	- Kembali ke halaman verifikasi jadwal	OK
5	Klik tombol “Tolak” pada menu verifikasi jadwal	Direktur Teknis	- Menampilkan <i>form</i> tolak jadwal	OK

6	Klik tombol “Simpan” pada <i>form</i> tolak jadwal	Direktur Teknis	<ul style="list-style-type: none"> - Jika data valid, data berhasil ditolak dan menampilkan pesan berhasil - Jika data tidak valid, menampilkan pesan kesalahan 	OK
7	Klik tombol “Batal” pada <i>form</i> verifikasi jadwal	Direktur Teknis	<ul style="list-style-type: none"> - Kembali ke halaman verifikasi jadwal 	OK

k. Verifikasi Tim

Tabel 4.57 Hasil Pengujian Menu Verifikasi Tim

No	Rancangan Proses	Pengguna	Hasil yang diharapkan	Hasil
1	Klik menu Verifikasi “Tim”	Direktur Teknis	<ul style="list-style-type: none"> - Menampilkan halaman verifikasi tim 	OK
2	Klik tombol “Verifikasi” pada menu verifikasi tim	Direktur Teknis	<ul style="list-style-type: none"> - Menampilkan <i>form</i> verifikasi tim 	OK
3	Klik tombol “Setuju” pada <i>form</i> verifikasi tim	Direktur Teknis	<ul style="list-style-type: none"> - Jika data valid, data berhasil diverifikasi dan menampilkan pesan berhasil - Jika data tidak valid, menampilkan pesan kesalahan 	OK
4	Klik tombol “Batal” pada <i>form</i> verifikasi tim	Direktur Teknis	<ul style="list-style-type: none"> - Kembali ke halaman verifikasi tim 	OK

5	Klik tombol “Tolak” pada menu verifikasi tim	Direktur Teknis	- Menampilkan <i>form</i> tolak tim	OK
6	Klik tombol “Simpan” pada <i>form</i> tolak tim	Direktur Teknis	- Jika data valid, data berhasil ditolak dan menampilkan pesan berhasil - Jika data tidak valid, menampilkan pesan kesalahan	OK
7	Klik tombol “Batal” pada <i>form</i> verifikasi tim	Direktur Teknis	- Kembali ke halaman verifikasi tim	OK

1. Verifikasi Laporan

Tabel 4.58 Hasil Pengujian Menu Verifikasi Laporan

No	Rancangan Proses	Pengguna	Hasil yang diharapkan	Hasil
1	Klik menu Verifikasi “Laporan”	Direktur Teknis	- Menampilkan halaman verifikasi laporan	OK
2	Klik tombol “Verifikasi” pada menu verifikasi laporan	Direktur Teknis	- Menampilkan <i>form</i> verifikasi laporan	OK
3	Klik tombol “Setuju” pada <i>form</i> verifikasi laporan	Direktur Teknis	- Jika data valid, data berhasil diverifikasi dan menampilkan pesan berhasil	OK

			<ul style="list-style-type: none"> - Jika data tidak valid, menampilkan pesan kesalahan 	
4	Klik tombol “Batal” pada <i>form</i> verifikasi laporan	Direktur Teknis	<ul style="list-style-type: none"> - Kembali ke halaman verifikasi laporan 	OK
5	Klik tombol “Tolak” pada menu verifikasi laporan	Direktur Teknis	<ul style="list-style-type: none"> - Menampilkan <i>form</i> tolak laporan 	OK
6	Klik tombol “Simpan” pada <i>form</i> tolak laporan	Direktur Teknis	<ul style="list-style-type: none"> - Jika data valid, data berhasil ditolak dan menampilkan pesan berhasil - Jika data tidak valid, menampilkan pesan kesalahan 	OK
7	Klik tombol “Batal” pada <i>form</i> verifikasi laporan	Direktur Teknis	<ul style="list-style-type: none"> - Kembali ke halaman verifikasi laporan 	OK

m. Verifikasi Analisis

Tabel 4.59 Hasil Pengujian Menu Verifikasi Analisis

No	Rancangan Proses	Pengguna	Hasil yang diharapkan	Hasil
1	Klik menu Verifikasi “Analisis”	Direktur Teknis	<ul style="list-style-type: none"> - Menampilkan halaman verifikasi analisis 	OK
2	Klik tombol “Verifikasi” pada	Direktur Teknis	<ul style="list-style-type: none"> - Menampilkan <i>form</i> verifikasi analisis 	OK

	menu verifikasi analisis			
3	Klik tombol “Setuju” pada <i>form</i> verifikasi analisis	Direktur Teknis	<ul style="list-style-type: none"> - Jika data valid, data berhasil diverifikasi dan menampilkan pesan berhasil - Jika data tidak valid, menampilkan pesan kesalahan 	OK
4	Klik tombol “Batal” pada <i>form</i> verifikasi analisis	Direktur Teknis	<ul style="list-style-type: none"> - Kembali ke halaman verifikasi analisis 	OK
5	Klik tombol “Tolak” pada menu verifikasi analisis	Direktur Teknis	<ul style="list-style-type: none"> - Menampilkan <i>form</i> tolak analisis 	OK
6	Klik tombol “Simpan” pada <i>form</i> tolak analisis	Direktur Teknis	<ul style="list-style-type: none"> - Jika data valid, data berhasil ditolak dan menampilkan pesan berhasil - Jika data tidak valid, menampilkan pesan kesalahan 	OK
7	Klik tombol “Batal” pada <i>form</i> verifikasi analisis	Direktur Teknis	<ul style="list-style-type: none"> - Kembali ke halaman verifikasi analisis 	OK

n. Tugas

Tabel 4.60 Hasil Pengujian Menu Tugas

No	Rancangan Proses	Pengguna	Hasil yang diharapkan	Hasil
1	Klik menu “Tugas”	<i>Team Leader</i>	- Menampilkan halaman data tugas	OK
2	Klik tombol “Tambah” pada menu “Tugas”	<i>Team Leader</i>	- Menampilkan <i>form</i> tambah data tugas	OK
3	Klik tombol “Simpan” pada <i>form</i> tambah data “Tugas”	<i>Team Leader</i>	<ul style="list-style-type: none"> - Jika data valid, menyimpan data tugas dan menampilkan pesan berhasil - Jika data tidak valid, menampilkan pesan kesalahan 	OK
4	Klik tombol “Batal” pada <i>form</i> tambah data “Tugas”	<i>Team Leader</i>	- Kembali ke halaman data tugas	OK
5	Klik tombol “Ubah” pada menu “Tugas”	<i>Team Leader</i>	- Menampilkan <i>form</i> ubah data tugas	OK
6	Klik tombol “Simpan” pada <i>form</i> tambah ubah “Tugas”	<i>Team Leader</i>	<ul style="list-style-type: none"> - Jika data valid, menyimpan data perubahan tugas dan menampilkan pesan berhasil - Jika data tidak valid, menampilkan pesan kesalahan 	OK
7	Klik tombol “Batal” pada <i>form</i>	<i>Team Leader</i>	- Kembali ke halaman data tugas	OK

	tambah ubah “Tugas”			
6	Klik Tombol “Hapus” pada menu “Tugas”	<i>Team Leader</i>	- Menampilkan pesan konfirmasi	OK
7	Klik tombol “Iya” pada pesan konfirmasi hapus	<i>Team Leader</i>	- Menghapus data tugas dan menampilkan pesan berhasil	OK
8	Klik tombol “Batal” pada pesan konfirmasi hapus	<i>Team Leader</i>	- Kembali ke halaman data tugas	OK
9	Klik menu “Tugas”	<i>Tenaga Ahli</i>	- Menampilkan halaman data tugas tenaga ahli	OK
10	Klik salah satu tugas yang belum dikerjakan	<i>Tenaga Ahli</i>	- Menampilkan <i>form detail</i> tugas dan <i>form hasil tugas</i>	OK
11	Klik tombol “Mulai” pada <i>form</i> detail tugas	<i>Tenaga Ahli</i>	- Mengganti status tugas sedang dikerjakan	OK
12	Klik tombol “Selesai” pada <i>form hasil tugas</i>	<i>Tenaga Ahli</i>	- Jika data valid, menyimpan hasil tugas dan menampilkan pesan berhasil - Jika data tidak valid, menampilkan pesan kesalahan	OK
13	Klik tombol “Batal” pada <i>form</i> tambah data “Tugas”	<i>Tenaga Ahli</i>	- Kembali ke halaman data tugas	OK

o. Laporan

Tabel 4.61 Hasil Pengujian Menu Laporan

No	Rancangan Proses	Pengguna	Hasil yang diharapkan	Hasil
1	Klik menu “Laporan”	Administrasi Proyek	- Menampilkan halaman data laporan	OK
2	Klik tombol “Tambah” pada menu “Laporan”	Administrasi Proyek	- Menampilkan <i>form</i> tambah data laporan	OK
3	Klik tombol “Simpan” pada <i>form</i> tambah data “Laporan”	Administrasi Proyek	<ul style="list-style-type: none"> - Jika data valid, menyimpan data laporan dan menampilkan pesan berhasil - Jika data tidak valid, menampilkan pesan kesalahan 	OK
4	Klik tombol “Batal” pada <i>form</i> tambah data “Laporan”	Administrasi Proyek	- Kembali ke halaman data laporan	OK
5	Klik tombol “Ubah” pada menu “Laporan”	Administrasi Proyek	- Menampilkan <i>form</i> ubah data laporan	OK
6	Klik tombol “Simpan” pada <i>form</i> tambah ubah “Laporan”	Administrasi Proyek	<ul style="list-style-type: none"> - Jika data valid, menyimpan data perubahan laporan dan menampilkan pesan berhasil - Jika data tidak valid, menampilkan pesan kesalahan 	OK
7	Klik tombol “Batal” pada <i>form</i>	Administrasi Proyek	- Kembali ke halaman data laporan	OK

	tambah ubah “Laporan”			
6	Klik Tombol “Hapus” pada menu “Laporan”	Administrasi Proyek	- Menampilkan pesan konfirmasi	OK
7	Klik tombol “Iya” pada pesan konfirmasi hapus	Administrasi Proyek	- Menghapus data laporan dan menampilkan pesan berhasil	OK
8	Klik tombol “Batal” pada pesan konfirmasi hapus	Administrasi Proyek	- Kembali ke halaman data laporan	OK

p. Pesan

Tabel 4.62 Hasil Pengujian Menu Pesan

No	Rancangan Proses	Pengguna	Hasil yang diharapkan	Hasil
1	Klik menu Pesan	Semua pengguna	- Menampilkan halaman daftar pesan	OK
2	Klik tombol “Cari Pengguna” pada menu daftar pesan	Semua pengguna	- Menampilkan <i>form</i> nama-nama pengguna pada sistem	OK
3	Klik tombol salah satu user pada <i>form</i> nama-nama pengguna	Semua pengguna	- Menampilkan form detail pesan dengan pengguna yang dipilih	OK
4	Klik tombol “Kirim” pada <i>form</i> detail pesan	Semua pengguna	- Mengirim pesan ke pengguna yang dituju dan menampilkan pesan berhasil	OK

5	Klik salah satu pesan pada daftar pesan	Semua pengguna	- Menampilkan form detail pesan dengan pengguna yang dipilih	OK
---	---	----------------	--	----

q. Profil

Tabel 4.63 Hasil Pengujian Menu Profil

No	Rancangan Proses	Pengguna	Hasil yang diharapkan	Hasil
1	Klik menu Profil	Semua pengguna	- Menampilkan halaman profil pengguna	OK
2	Klik tombol “Ubah” pada menu profil	Semua pengguna	- Menampilkan <i>form</i> ubah data <i>user</i>	OK
3	Klik tombol “Simpan” pada <i>form</i> ubah data user	Semua pengguna	<ul style="list-style-type: none"> - Jika data valid, mengubah data <i>user</i> dan menampilkan pesan berhasil - Jika data tidak valid, menampilkan pesan kesalahan 	OK

r. Logout

Tabel 4.64 Hasil Pengujian Logout

No	Rancangan Proses	Pengguna	Hasil yang diharapkan	Hasil
1	Klik “Keluar” pada sidemenu	Semua pengguna	- Keluar dari sistem dan menampilkan halaman <i>login</i>	OK

BAB 5

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil pembahasan dan penjelasan pada bab-bab sebelumnya, dapat disimpulkan bahwa:

- a. Direktur teknis dan kepala divisi proyek dapat mengontrol proyek-proyek dengan menggunakan sistem informasi manajemen proyek ini melalui *dashboard* yang pada bagian *progress*.
- b. Sistem manajemen informasi proyek ini mempunyai fitur analisis keterlambatan proyek menggunakan *Critical Path Method* dan *what if analysis*.
- c. Sistem yang telah dibangun menyediakan notifikasi sebagai pengingat jika proyek, aktivitas, tugas dan laporan sudah mendekati waktu tenggat pada bagian *dashboard* sistem.
- d. Sistem informasi manajemen proyek berbasis *web* ini dilengkapi dengan fitur tugas yang dapat dibuat dan ditutup oleh *team leader* dan di-update langsung oleh tenaga ahli kapan saja dan dimana saja serta pembuatan laporan secara *otomatis*.
- e. Dalam penelitian terdapat kendala pada waktu penelitian yang terbatas sehingga mengakibatkan penelitian memiliki kekurangan khususnya pada tahap penelitian yang terbatas sampai pengujian *black box testing* dan juga analisis keterlambatan proyek yang hanya mempertimbangkan dari jumlah pekerja dan jam.

5.2 Saran

Dari kesimpulan berikut beberapa saran untuk penelitian selanjutnya:

- a. Melakukan analisis keterlambatan proyek dengan mempertimbangkan dari segi biaya proyek.
- b. Penelitian selanjutnya dapat mengembangkan sistem untuk mengelola dokumen-dokumen pengajuan proyek.
- c. Melakukan pengujian *usability* menggunakan seperti metode *Heuristic Evaluation* pada sistem yang telah dibangun agar sistem lebih baik lagi dari segi tampilan *user interface*, *user experience* dan fungsional.



DAFTAR PUSTAKA

- Abdurrasyid, A., Luqman, L., Haris, A., & Indrianto, I. (2019). Implementasi Metode PERT dan CPM pada Sistem Informasi Manajemen Proyek Pembangunan Kapal. *Khazanah Informatika: Jurnal Ilmu Komputer dan Informatika*, 5(1), 28–36. <https://doi.org/10.23917/khif.v5i1.7066>
- Agustina, B., & Kusnanto, G. (2018). Sistem Informasi Manajemen Proyek PT. Xerindo Teknologi. *KONVERGENSI*, 14(1), 37–47.
- Anggraeni, E. Y., & Irviani, R. (2017). *Pengantar Sistem Informasi*. Yogyakarta: CV Andi Offset.
- Aprisa, & Monalisa, S. (2015). Rancang Bangun Sistem Informasi Monitoring Perkembangan Proyek Berbasis Web (Studi Kasus: PT. Inti Pratama Semesta). *Jurnal Rekayasa dan Manajemen Sistem Informasi*, 1(1), 49–54. Diambil dari <http://ejournal.uin-suska.ac.id/index.php/RMSI/article/view/1305>
- Apriyanto, D. (2018). *Sistem Informasi Manajemen Proyek Di PT. Yudha Perkasa Utama* (Universitas Komputer Indonesia). Diambil dari <https://repository.unikom.ac.id/id/eprint/57955>
- Arianie, G. P., & Puspitasari, N. B. (2017). Perencanaan Manajemen Proyek dalam Meningkatkan Efisiensi dan Efektifitas Sumber Daya Perusahaan (Studi Kasus: Qiscus Pte Ltd). *J@ti Undip : Jurnal Teknik Industri*, 12(3), 189. <https://doi.org/10.14710/jati.12.3.189-196>
- Aswati, S., & Siagian, Y. (2016). Model Rapid Application Development Dalam Rancang Bangun Sistem Informasi Pemasaran Rumah (Studi Kasus : Perum

- Perumnas Cabang Medan. *Sesindo*, 5, 317–324.
- Deshpande, C. (2021). The Best Guide to Know What Is React. Diambil dari simplilearn website: <https://www.simplilearn.com/tutorials/reactjs-tutorial/what-is-reactjs>
- EMS, T. (2016). *PHP 5 dari Nol*. Jakarta: PT Elex Media Komputindo.
- Fatimah, S. N. (2019). *Antisipasi Keterlambatan Waktu Penyelesaian Proyek dengan Menggunakan Metode What If Analysis dan Crash Program*. Universitas Islam Sultan Agung.
- Firmansyah, M. R. (2017). *Analisa Perencanaan Proyek Kapal Cepat Rudal Dengan Metode CPM dan What If Analysis*. Institut Teknologi Sepuluh Nopember.
- Hamali, A. Y. (2016). *Pemahaman Strategi Bisnis & Kewirausahaan*. Jakarta: Prenada Media Group.
- Hendini, A. (2016). Pemodelan UML Sistem Informasi Monitoring Penjualan dan Stok Barang (Studi Kasus: Distro Zhezha Pontianak). *Jurnal Khatulistiwa Informatika*, IV(2), 107--117. Diambil dari <https://ejournal.bsi.ac.id/ejurnal/index.php/khatulistiwa/article/viewFile/1262/1027>
- Husodo, I. T. (2017). Perbandingan Sistem Pembiayaan Kontrak Proyek dengan Analisis Optimalisasi Linier Program. *Jurnal Ilmiah Teknosains*, 3(1), 1–10. Diambil dari <https://doi.org/10.26877/jitek.v3i1/ Mei.1385>
- Hutahaenan, J. (2015). *Konsep Sistem Informasi*. Yogyakarta: Deepublish.
- Hutauruk, M. K. (2019). UML Diagram : Use Case Diagram. Diambil 24 April

- 2020, dari <https://soc.sbinus.ac.id/2019/11/26/uml-diagram-use-case-diagram/>
- Indrajani. (2015). *Database Design*. Jakarta: PT Elex Media Komputindo.
- Institute, P. M. (2017). PMBOK 6th edition. In *Project Management Institute* (Vol. 34).
- Iwawo, E. R. M., Tjakra, J., & Pratasis, P. A. K. (2016). Penerapan Metode Cpm Pada Proyek Konstruksi (Studi Kasus Pembangunan Gedung Baru Kompleks Eben Haezar Manado). *Jurnal Sipil Statik*, 4(9), 551–558.
- Jubilee Enterprise. (2018). *HTML, PHP dan MySQL untuk Pemula*. Jakarta: PT Elex Media Komputindo.
- Kempp, S. (2018). Digital in 2018: World's Internet Users Pass The 4 Billion Mark.
- Kosasi, S. (2015). Penerapan Rapid Application Development Dalam Sistem Perniagaan Elektronik Furniture. *Citee Journal*, 2(4), 265–276. Diambil dari http://www.jurnal.stmikpontianak.ac.id/file/SANDY_KOSASI_-_CITECH_2015.pdf
- Lathif, R. (2019). Apa itu draw.io? Ini Penjelasan Lengkapnya. Diambil 25 April 2020, dari <https://surgatekno.com/tech-news/apa-itu-draw-io/>
- Lestari, C., & Amri, A. M. (2020). *Sistem Informasi Akuntansi Beserta Contoh Penerapan Aplikasi SIA Sederhana Dalam UMKM*. Yogyakarta: Deepublish.
- Maulana, A. (2018). *Analisis Metode “What If” Sebagai Antisipasi Keterlambatan Proyek (Studi Kasus Pada Proyek Pembangunan Puncak Dharma Husada Surabaya)*. Universitas 17 Agustus 1945.
- Maulani, G., Septiani, D., & Sahara, P. N. F. (2018). Rancang Bangun Sistem

- Informasi Inventory Fasilitas Maintenance Pada PT. PLN (persero) Tangerang. *Jurnal ICIT*, 4(2), 156–167. Diambil dari <https://media.neliti.com/media/publications/300355-rancang-bangun-sistem-informasi-inventor-6e3bb780.pdf>
- Minnick, J. (2016). *Web Design with HTML5 and CSS3*. Boston: Cengage Learning.
- Mulyani, S. (2016). *Sistem Informasi Manajemen Rumah Sakit: Analisis dan Perancangan* (2 ed.). Bandung: Abdi Sistematika.
- Muslihudin, M., & Oktafiano. (2016). *Analisis dan Perancangan Sistem Informasi Menggunakan Model Terstruktur dan UML*. Yogyakarta: CV Andi Offset.
- Nafiudin. (2019). *Buku Ajar Mata Kuliah Sistem Informasi Manajemen*. Pasuruan: Qiara Media.
- Nazli, R. (2018). Pemodelan Aplikasi Mobile Modul Perkuliahan Berbasis Client Server. *Jurnal Teknologi dan Open Source*, 1(1), 25–32. Diambil dari <https://media.neliti.com/media/publications/284720-pemodelan-aplikasi-mobile-modul-perkuliah-0eed6780.pdf>
- Pahlevi, S. M. (2013). *Tujuh Langkah Praktis Pembangunan Basis Data*. Jakarta: PT. Elex Meida Komputindo.
- Pramudita, R., Arifin, R. W., Alfian, A. N., Safitri, N., & Anwariya, S. D. (2021). Penggunaan Aplikasi Figma Dalam Membangun UI/UX yang Interaktif Pada Program Studi Teknik Informatika STMIK Tasikmalaya. *Jurnal Buana Pengabdian*, 3(1), 150–152. Diambil dari <https://journal.ubpkarawang.ac.id/index.php/JurnalBuanaPengabdian/article/>

view/1542/1164

- Prihandoyo, M. T. (2018). Unified Modeling Language (UML) Model Untuk Pengembangan Sistem Informasi Akademik Berbasis Web. *Jurnal Informatika: Jurnal Pengembangan IT (JPIT)*, 3(1), 126–129.
- Rompas, L. M. (2020). Penerapan Teknologi Pada Sistem Informasi Manajemen Proyek Sistem Informasi Kontraktor Dan Konsultan (Studi Kasus Kota Manado). *Tekno*, 17(74), 101–112.
- Sitanggang, N., Simamarta, J., & Luthan, P. L. (2019). *Pengantar Konsep Manajemen Proyek Untuk Teknik*. Medan: Yayasan Kita Menulis.
- Sugiyanto. (2020). *Manajemen Pengendalian Proyek*. Surabaya: Scopindo Media Pustaka.
- Sutabri, T. (2012). *Konsep Sistem Informasi*. Yogyakarta: CV Andi Offset.
- Syahrul, S. (2017). *Pengembangan Sistem Informasi Manajemen Proyek (Studi Kasus: PT Mozaik Bintang Persada)* (Vol. 4). Universitas Islam Negeri Syarif Hidayatullah.
- Unas, S. El, Hasyim, M. H., & Negara, K. P. (2014). Antisipasi Keterlambatan Proyek Menggunakan Metode What If Diterapkan Pada Microsoft Project. *Jurnal Rekayasa Sipil*, 8(3), 192–197.
- Unhelkar, B. (2018). *Software Engineering with UML*. Boca Raton: CRC Press.
- Utami, F. H., & Asnawati. (2015). *Rekayasa Perangkat Lunak* (1 ed.). Yogyakarta: Deepublish.
- Visual Paradigm. (2020a). What is Component Diagram? Diambil 18 Mei 2020, dari Visual Paradigm website: <https://www.visual-paradigm.com/guide/uml>

- unified-modeling-language/what-is-component-diagram/
- Visual Paradigm. (2020b). What is Deployment Diagram? Diambil 18 Mei 2020, dari Visual Paradigm website: <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-deployment-diagram/>
- Wibowo, R. A. (2019). *Manajemen Pemasaran*. Semarang: Semarang University.
- Wicaksono, S. R. (2017). *Rekayasa Perangkat Lunak*. Malang: Seribu Bintang.
- Widaningsih, I., Maukar, A. L., & Shinta, A. (2017). Antisipasi Keterlambatan Proyek Pembangunan Shelter dengan Menggunakan Metode What-If Analysis. *Journal of Industrial Engineering, Scientific Journal on Research and Application of Industrial System*, 2(2), 75–84.
- Widodo, A. W., & Kumianingtyas, D. (2017). *Sistem Basis Data*. Malang: UB Press.
- Yanto, R. (2016). *Manajemen Basis Data Menggunakan MySQL*. Sleman: Deepublish.
- Yasin. (2019). Pengertian MySQL, Fungsi dan Cara Kerjanya (Lengkap). Diambil dari <https://www.niagahoster.co.id/blog/mysql-adalah/>

LAMPIRAN I

HASIL WAWANCARA

Nama Narasumber	Ibu Ir. Sarwo Sri
Jabatan	Direktur Utama
Tempat	PT. Triprima Karya, Jl. Arjuna VI DF 3 No. 6, Villa Pamulang, Pondok Benda, Pamulang, Tangerang Selatan, Banten
Waktu	2 Maret 2020
1. (Pertanyaan): Bagaimana alur proses bisnis yang berjalan pada perusahaan?	
(Jawaban): Untuk memulai proyek, Triprima mengajukan terlebih dahulu kepada <i>owner proyek</i> menggunakan dokumen-dokumen administrasi dan teknis. Lalu jika <i>owner proyek</i> sudah menyetujui pengajuan, dapat dilakukan persetujuan, nego dan pembuatan kontrak. Kalau sudah ada kontrak dan SPK, bisa dimulai proyeknya. Untuk menandakan proyek selesai, kita memberikan laporan akhir ke pihak <i>owner proyek</i>	
2. (Pertanyaan): Bagaimana proses pemantauan proyek di PT Triprima Karya?	
(Jawaban): Iya untuk memantau proyek, direksi biasanya menghubungi team leader proyek melalui telepon untuk menanyakan progress proyek. Team leadernya nanti minta tolong ke tenaga ahli yang sudah dipilih untuk	

melakukan tugas kaya pengawasan, gambar, survey, mengantar dokumen dan macam-macam nanti hasilnya dikasih ke team leader lagi biasanya seminggu sekali. Laporan dari tim teknis ini dikasih ke bagian administrasi proyek yang membuat dokumen-dokumen untuk dirangkum menjadi laporan-laporan.

3. (Pertanyaan): Kendala apa yang sering dihadapi perusahaan dalam memanajamen proyek?

(Jawaban): Kendalanya kalau proyek lagi banyak, jadi lebih sulit untuk mantaunya karena banyak yang harus dikerjain. Karena kebanyakan, beberapa kali proyeknya jadi terlambat. Dari laporan yang salah-salah ketik juga membuat proyek terlambat jadi yang seharusnya sudah selesai harus direvisi lagi laporannya karena salah ketik.

4. (Pertanyaan): Laporan apa saja yang dibutuhkan untuk penutupan proyek?

(Jawaban): Tergantung dari jenis proyeknya. Di triprima proyeknya dikelompokkin jadi tiga bagian. Ada pengawasan, perencanaan dan audit Kalau proyek pengawasan laporannya harian, mingguan, bulanan, akhir. Kalau perencanaan dan audit hanya laporan akhir. Semua proyek pasti bikin laporan akhir.

LAMPIRAN II

DOKUMEN DAN SURAT-SURAT

**KEMENTERIAN AGAMA
UNIVERSITAS ISLAM NEGERI (UIN)
SYARIF HIDAYATULLAH JAKARTA
FAKULTAS SAINS DAN TEKNOLOGI**

Jl. M. H. Thamrin No. 19 Cilandak 12412 Indonesia
Tele: (021) 7499800, 7499847, Fax: (021) 7499855
Email: fst@uinjkt.ac.id
Website: fst.uinjkt.ac.id

Nomor : B- 6297 / F9 / KM.01 / 02 / 2020
Lampiran :
Perihal : Bimbingan Skripsi

Jakarta, 20 Februari 2020

Kepada Yth.
1. Zulfiandri, MMSI
2. Rinda Hesti Kusumaningtyas, MMSI

Assalamu'alaikum Wr. Wb.

Dengan ini diharapkan kesediaan Saudara untuk menjadi pembimbing I/I/ (Materi/Teknis)* penulisan skripsi mahasiswa:

Nama	:	Farah Dhaia Yasmin
Nim	:	1116093000084
Program Studi	:	Sistem Informasi
Judul Skripsi	:	"Rancang Bangun Sistem Informasi Electronic Document Management System Berbasis Website (Studi Kasus : PT. Triprima Karya)"

Judul tersebut telah disetujui oleh Program Studi bersangkutan pada tanggal dengan outline, abstraksi dan daftar pustaka terlampir. Bimbingan skripsi ini diharapkan selesai dalam waktu 6 (enam) bulan setelah ditandatanganinya surat penunjukan pembimbing skripsi.

Apabila terjadi perubahan terkait dengan skripsi tersebut selama proses pembimbingan, harap segera melaporkan kepada Program Studi bersangkutan.

Demikian atas kesedian Saudara, kami ucapkan terima kasih.

Wassalamu'alaikum Wr. Wb.

a.n Dekan
Wadik Bidang Akademik


Nafirul Hakim, S. Si., M.T., Ph.D
NIP. 19710508 200501 1 005



KEMENTERIAN AGAMA
UIN SYARIF HIDAYATULLAH JAKARTA
FAKULTAS SAINS DAN TEKNOLOGI

Jl. Ir. H. Juanda No. 95 Ciputat 15412 Indonesia
Telp. (02-21) 74938000, 7493547 Fax. (02-21) 7493015

Website : fst.uinjkt.ac.id
Email : fst@uinjkt.ac.id

Nomor : B - 0716/F9 / KM. 01 /02/2022

Jakarta, 09 Februari 2022

Lampiran : -

Hal : Permohonan Riset

Kepada Yth.
Direktur Utama
PT TRIPRIMA KARYA
Di
Tempat
Assalamualaikum, Wr Wb

Dengan hormat kami sampaikan bahwa:

Nama	:	FARAH DHIA YASMIN
Tempat/Tanggal Lahir	:	Jakarta / 10 Mei 1998
NIM	:	11160930000084
Semester	:	11
Program Studi	:	Sistem Informasi
Alamat	:	Pamulang Villa DF-3/6
Telp/HP	:	082211266792

adalah benar yang bersangkutan mahasiswa Fakultas Sains dan Teknologi UIN Syarif Hidayatullah Jakarta dan bermaksud melakukan penelitian/riset di instansi yang Bapak/Ibu pimpin, yang sedang dalam penyelesaian skripsi dengan judul skripsi:

"Rancang Bangun Sistem Informasi Manajemen Proyek Menggunakan What If Analysis Pada Critical Path Method Berbasis Web"

Untuk itu, kami mohon kesediaannya untuk memberikan kesempatan kepada mahasiswa tersebut dalam melaksanakan penelitian Data/riset di instansi/perusahaan yang Bapak/Ibu pimpin.

Demikian, atas perhatian dan kerjasamanya kami ucapkan terima kasih.

Wassalamu 'alaikum, Wr Wb



Jakarta, 09 Februari 2022



Dr. H. Siti Rochaeni, M.Si.

NIP. 196203081989032001/-

Cek Keaslian Surat di <https://e-letter.fst.uinjkt.ac.id/scan>



No: 23/TPK/SB/II/2022

Yang Terhormat,
FAKULTAS SAINS DAN TEKNOLOGI
UIN SYARIF HIDAYATULLAH
Jl. Ir. H. Juanda No. 95, Ciputat 15412

Up: Ibu Dr. Ir. Siti Rochaeni, M.Sc.

Dengan hormat,

Yang bertanda tangan di bawah ini:

Nama : Ir. Sarwo Sri
Jabatan : Direktur Utama
PT Triprima Karya Konsultan

Menerangkan bahwa,

Nama : FARAH DHIA YASMIN
NIM : 11160930000084
Semester : 11
Program Studi : Sistem Informasi
Fakultas Sains Dan Teknologi UIN Syarif Hidayatullah

Telah kami setujui untuk melaksanakan penelitian data/riset pada perusahaan kami sebagai syarat penyusunan skripsi dengan judul:

"Rancang Bangun Sistem Informasi Manajemen Proyek Menggunakan What If Analysis Pada Critical Path Method Berbasis Web"

Demikian surat ini kami sampaikan untuk dapat dipergunakan sebagaimana mestinya.

Jakarta, 10 Februari 2022
PT TRIPRIMA KARYA KONSULTAN

Ir. Sarwo Sri
Direktur Utama

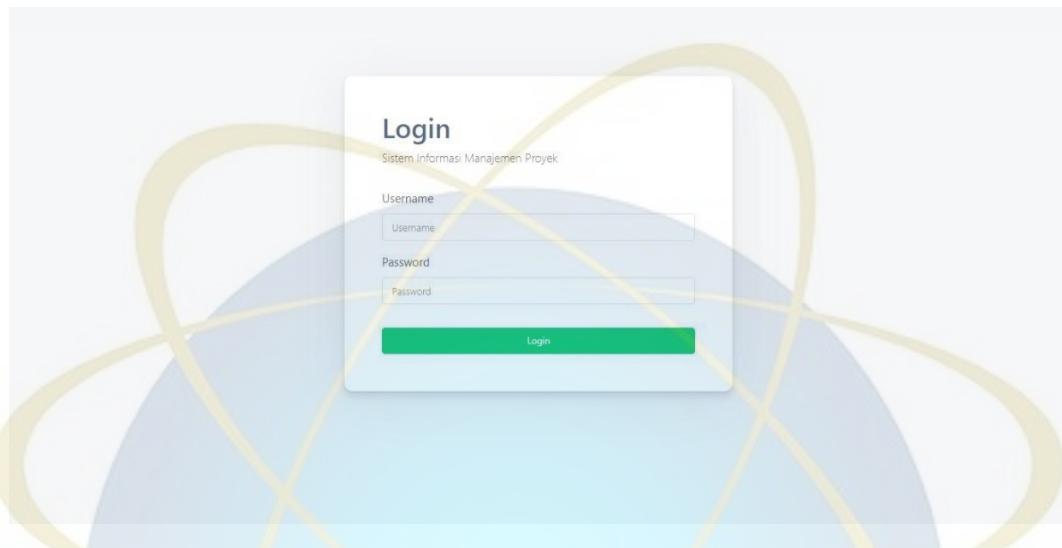
Alamat : Embrio Business Center Gd. Setiabudi 2 Lt.2 Jl. HR Rasuna Said Kav A2, Jakarta Selatan- Phone: (021)52900028
Kantor Operasional : Blok A/C No.6 Vila Mutuara, Ciputat - Phone/Fax: (021) 7412620. E-mail : triprima_karya@yahoo.co.id

PERENCANAAN, SUPERVISI, ADVISE, MANAJEMEN KONSTRUKSI / ARSITEKTUR, STRUKTUR, LANSEKAP, MEKANIKAL, ELEKTRIKAL

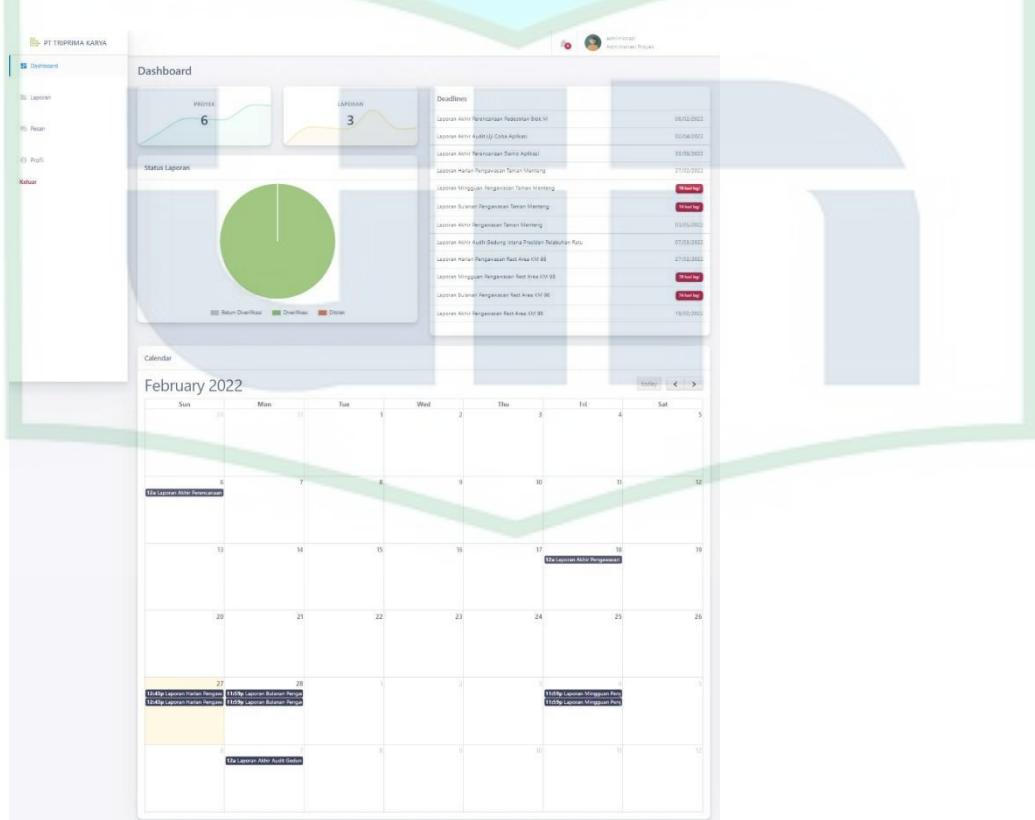
LAMPIRAN III

USER INTERFACE

1. *Login Interface*



2. *Dashboard Interface*



PT TRIPRIMA KARYA

Dashboard

Workload

Kategori	Fahira Nafiah Azhaar	Safira Noor Fauziah
Audit Uji Coba Aplikasi	1	2
Perencanaan Demo Aplikasi	0	3
Perencanaan Pedestrian Blok M	0	2

Status Proyek

Status	Persentase
Berlangsung	~70%
Selesai	~20%
Belum Berlangsung	~10%

Calendar

February 2022

Sun	Mon	Tue	Wed	Thu	Fri	Sat
30	31	1	2	3	4	5
+2 more		A				Desain 2D
6	7	8	9	10	11	12
+2 more						
13	14	15	16	17	18	19
	Pemasang	Pekerjaan	B	Laboratori	+4 more	
20	21	22	23	24	25	26
	J	D				
27	28	1	2	3	4	5
	Pekerjaan	Pekerjaan	E	+2 more	I	
6	7	8	9	10	11	12

Progress

Belum Dikerjakan | Sedang Dikerjakan | Butuh Koreksi | Revisi | Selesai

Proyek	Progress (%)	Terakhir
Perencanaan Pedestrian Blok M	100.00%	06 Feb 2022
Audit Uji Coba Aplikasi	0.00%	02 Apr 2022
Perencanaan Demo Aplikasi	100.00%	30 Sep 2022
Pengawasan Taman Menteng	0.00%	03 May 2022
Audit Gedung Istana Presiden Pelabuhan Ratu	0.00%	07 Mar 2022
Pengawasan Rest Area KM 98	0.00%	18 Feb 2022

Gantt Chart

Perencanaan Demo Aplikasi - Perencanaan

Projects	March 2022																													
	Sa	Mo	Tu	We	Th	Fr	Sa	Su	Mo	Tu	We	Th	Fr	Sa	Su	Mo	Tu	We	Th	Fr	Sa	Su	Mo	Tu	We	Th	Fr	Sa		
Pekerjaan Pendahuluan	27	28	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28
Pekerjaan Persiapan	28	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	1
Pekerjaan Konstruksi	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	1	2
Perlengkapan Penunjang	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	1	2	3
Pekerjaan Elektrikal	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	1	2	3	4
Pekerjaan Penghijauan	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	1	2	3	4	5

3. User Table Interface

User ID	Nama	Username	Role	Aksi
USR02220001	Farah Dhia Yasmin	farahdhia	Tenaga Ahli	<button>Ubah</button> <button>Hapus</button>
USR11210001	Samal Surya	teamleader	Team Leader	<button>Ubah</button> <button>Hapus</button>
USR11210004	Safira Noor Fauziah	tenagaahli	Tenaga Ahli	<button>Ubah</button> <button>Hapus</button>
USR11210002	Sarwo Sri	admin	Admin	<button>Ubah</button> <button>Hapus</button>
USR11210003	Warisno	dirtek	Direktur Teknis	<button>Ubah</button> <button>Hapus</button>
USR11210005	Samal Surya	kadiv	Kepala Divisi Proyek	<button>Ubah</button> <button>Hapus</button>
USR11210006	Sarwo Sri	administrasi	Administrasi Proyek	<button>Ubah</button> <button>Hapus</button>

4. Form User Interface

MASTER
User

Form User

Nama *
Select...

Username *
Username

Password *
Password

Role *
Select...

Batal Simpan

5. Klien Table Interface

Klien ID	Nama Perusahaan	PIC	Telp	Aksi
KLN01220001	BNI	John Doe	1234567890987	<button>Ubah</button> <button>Hapus</button>
KLN01220002	Dinas Lingkungan Hidup dan Kehutanan	Alif	0987654321	<button>Ubah</button> <button>Hapus</button>
KLN02220001	PT. Test Jaya	Farah	081234567890	<button>Ubah</button> <button>Hapus</button>
KLN02220002	PT. Farah Karya	Dhia	09876543212	<button>Ubah</button> <button>Hapus</button>
KLN02220003	PT. Yasmin Konstruksi	Yasmin	00112233445566	<button>Ubah</button> <button>Hapus</button>

6. Form Klien Interface

MASTER
Form Klien

Nama Perusahaan *

Nama PIC *

Telp *

Batal Simpan

7. Karyawan Table Interface

Karyawan ID	Nama	TTL	Alamat	Telp	Aksi
KYW02220001	Farah Dchia Yasmin	BANDUNG, 1999-02-04	Pamulang	000011112222	<button>Ubah</button> <button>Hapus</button>
KYW11210005	Fahira Nafiah Azhaar	JAKARTA, 2000-10-02	Jalan Raya	0811233444	<button>Ubah</button> <button>Hapus</button>
KYW11210001	Samal Surya	JAKARTA, 1963-11-06	Jakarta	0818114023	<button>Ubah</button> <button>Hapus</button>
KYW11210002	Sarwo Sri	JAKARTA, 1966-06-11	Perum Menteng	0812121212	<button>Ubah</button> <button>Hapus</button>
KYW11210003	Warisno	JAKARTA, 1960-07-07	Depok	082233445566	<button>Ubah</button> <button>Hapus</button>
KYW11210004	Safira Noor Fauziah	JAKARTA, 1996-05-01	Bogor	08112334455	<button>Ubah</button> <button>Hapus</button>

8. Form Karyawan Interface

The form consists of several input fields:

- Nama Lengkap ***: Input field for full name.
- Posisi ***: Input field for position, with a dropdown menu labeled "Select...".
- Tempat Lahir ***: Input field for place of birth, with a dropdown menu labeled "Select...".
- Tanggal Lahir ***: Input field for date of birth, with a date picker and a placeholder "mm/dd/yyyy".
- Alamat ***: Input field for address.
- Telp ***: Input field for phone number.
- Email ***: Input field for email address.
- NPWP**: Input field for NPWP number.

At the bottom of the form are two buttons: **Batal** (Cancel) and **Simpan** (Save).

9. Proyek Table Interface

Proyek ID	Nama Paket	Tipe	Lokasi	Waktu Mulai	Waktu Selesai	Status
PRY02220001	Perencanaan Pedestrian Blok M	Perencanaan	JAKARTA SELATAN	2021-12-08 12:55:00	2022-02-06 00:00:00	Selesai
PRY02220002	Audit Uji Coba Aplikasi	Audit	BANDUNG	2022-02-01 13:21:00	2022-04-02 00:00:00	Sedang Berlangsung
PRY02220003	Perencanaan Demo Aplikasi	Perencanaan	KOTA TANGERANG SELATAN	2022-02-01 13:23:00	2022-09-30 00:00:00	Sedang Berlangsung
PRY02220004	Pengawasan Taman Menteng	Pengawasan	JAKARTA PUSAT	2022-02-02 13:25:00	2022-05-03 00:00:00	Sedang Berlangsung
PRY02220005	Audit Gedung Istana Presiden Pelabuhan Ratu	Audit	KOTA SUKABUMI	2022-02-05 13:26:00	2022-03-07 00:00:00	Sedang Berlangsung
PRY02220006	Pengawasan Rest Area KM 98	Pengawasan	CIREBON	2022-02-13 13:29:00	2022-02-18 00:00:00	Sedang Berlangsung

10. Form Proyek Interface

The form includes the following fields:

- Program *
- Kegiatan *
- Rincian Sub Kegiatan *
- Klien *
- Type *
- Nama Paket *
- Kode Rekening *
- Lokasi *
- Durasi *
- Waktu Mulai *
- Waktu Selesai

Buttons at the bottom include "Batal" (Cancel) and "Simpan" (Save).

XXX

11. Verifikasi Jadwal *Interface*

The screenshot shows the 'VERIFIKASI Jadwal' page. The left sidebar has a 'Jadwal' section selected. The main area displays a table with columns: Jadwal ID, Nama Proyek, Status, and Aksi. The table contains five rows of data:

Jadwal ID	Nama Proyek	Status	Aksi
JWL02220001	Perencanaan Pedestrian Blok M	Diverifikasi	
JWL02220002	Audit Uji Coba Aplikasi	Diverifikasi	
JWL02220003	Perencanaan Demo Aplikasi	Diverifikasi	
JWL02220004	Pengawasan Taman Menteng	Ditolak	
JWL02220005	Audit Gedung Istana Presiden Pelabuhan Ratu	Belum Diverifikasi	<button>Verifikasi</button> <button>Tolak</button>

At the bottom right of the table, there are buttons for 'First', '<', 'Page 1 of 1', '>', 'Last', and 'Go to page'.

12. Verifikasi Tim *Interface*

The screenshot shows the 'VERIFIKASI Tim' page. The left sidebar has a 'Tim' section selected. The main area displays a table with columns: Tim ID, Nama Proyek, Status, and Aksi. The table contains five rows of data:

Tim ID	Nama Proyek	Status	Aksi
TIM02220001	Perencanaan Pedestrian Blok M	Diverifikasi	
TIM02220002	Audit Uji Coba Aplikasi	Diverifikasi	
TIM02220003	Pengawasan Taman Menteng	Belum Diverifikasi	<button>Verifikasi</button> <button>Tolak</button>
TIM02220004	Perencanaan Demo Aplikasi	Diverifikasi	
TIM02220005	Audit Gedung Istana Presiden Pelabuhan Ratu	Ditolak	

At the bottom right of the table, there are buttons for 'First', '<', 'Page 1 of 1', '>', 'Last', and 'Go to page'.

13. Verifikasi Laporan Interface

The screenshot shows the 'VERIFIKASI Laporan' (Report Verification) screen. The left sidebar has a 'Laporan' item selected under 'VERIFIKASI'. The main area displays a table with three rows of report data:

Laporan ID	Nama Proyek	Judul Laporan	Tipe	Status	Aksi
LAP02220001	Perencanaan Pedestrian Blok M	Laporan Harian	Harian	Diverifikasi	
LAP02220002	Perencanaan Pedestrian Blok M	Laporan Bulanan	Bulanan	Diverifikasi	
LAP02220003	Perencanaan Pedestrian Blok M	Laporan Akhir Perencanaan Pedestrian Blok M	Akhir	Diverifikasi	

Below the table are navigation controls: 'Show 10', 'First', '<', 'Page 1 of 1', '>', 'Last', and 'Go to page 1'.

14. Verifikasi Analisis Interface

The screenshot shows the 'VERIFIKASI Analisis' (Analysis Verification) screen. The left sidebar has an 'Analisis' item selected under 'VERIFIKASI'. The main area displays a table with one row of analysis data:

Whatif ID	Nama Proyek	Aktivitas	Delay (hari)	Status	Aksi
WIF02220001	Perencanaan Demo Aplikasi	Pekerjaan Pendahuluan	7	Batum Diverifikasi	<button>Verifikasi</button> <button>Tolak</button>

Below the table are navigation controls: 'Show 10', 'First', '<', 'Page 1 of 1', '>', 'Last', and 'Go to page 1'.

15. Jadwal Interface

The screenshot shows the 'Detail Proyek' page with the 'Jadwal' tab selected. The main content area displays a table of activities with columns: Aktivitas, Durasi, Waktu Mulai, Waktu Selesai, and Successor. The activities listed are:

Aktivitas	Durasi	Waktu Mulai	Waktu Selesai	Successor
Pekerjaan Pendahuluan	14	2022-02-01	2022-02-15	
Pekerjaan Persiapan	14	2022-02-16	2022-03-02	Pekerjaan Pendahuluan
Pekerjaan Konstruksi	70	2022-03-02	2022-05-11	Pekerjaan Persiapan
Perlengkapan Perunjang	63	2022-03-02	2022-05-04	Pekerjaan Persiapan
Pekerjaan Mekanikal dan Plumbing	63	2022-05-12	2022-07-14	Pekerjaan Konstruksi
Pekerjaan Elektrikal	56	2022-07-15	2022-09-09	Pekerjaan Mekanikal dan Plumbing
Pekerjaan Penghijauan	28	2022-02-01	2022-03-01	

Below the table are navigation controls: 'Show 10', 'First', '<', 'Page 1 of 1', '>', 'Last', and 'Go to page 1'. A large yellow circle highlights the entire table area.

16. Tim Interface

The screenshot shows the 'Detail Proyek' page with the 'Tim' tab selected. The main content area displays a table of team members with columns: Anggota Tim, Team Leader, Operator, and Drafter. The team members listed are:

Anggota Tim	Team Leader	Operator	Drafter
Samal Surya	Samal Surya	Safira Noor Fauziah	Fahira Nafiah Azhaar

Below the table is a 'Kembali' button.

17. Analisis Interface

The image displays two screenshots of a project management application interface, likely from PT TRIPRIMA KARYA, illustrating the results of a Critical Path Method (CPM) analysis and a What If Analysis (Keterlambatan).

Screenshot 1: Hasil Analisis CPM (Critical Path)

This screenshot shows the "Detail Proyek" page under the "Analisis" tab. It features a large circular diagram representing the Critical Path Method (CPM) analysis. The diagram consists of five main nodes connected by arrows: "Pekerjaan Pendahuluan" (Preparatory Work), "Pekerjaan Persiapan" (Preparation Work), "Pekerjaan Konstruksi" (Construction Work), "Pekerjaan Mekanikal dan Plumbing" (Mechanical and Plumbing Work), and "Pekerjaan Elektrikal" (Electrical Work). The "Pekerjaan Pendahuluan" node is at the top, followed by "Pekerjaan Persiapan", "Pekerjaan Konstruksi", "Pekerjaan Mekanikal dan Plumbing", and finally "Pekerjaan Elektrikal" at the bottom. Arrows indicate the flow from one node to the next. Below the diagram, there is a table titled "Status Aktivitas" (Activity Status) listing various tasks along with their start dates and remaining duration.

Aktivitas	Tanggal Mulai	Sisa Waktu	Aksi
Pekerjaan Pendahuluan	15 Feb 2022	-12 hari lagi	Lihat Hasil Analisis Keterlambatan
Pekerjaan Persiapan	02 Mar 2022	75 hari lagi	Analisis
Pekerjaan Konstruksi	11 May 2022	284 hari lagi	Analisis
Perlengkapan Penunjang	04 May 2022	277 hari lagi	Analisis
Pekerjaan Mekanikal dan Plumbing	14 Jul 2022	487 hari lagi	Analisis
Pekerjaan Elektrikal	09 Sep 2022	682 hari lagi	Analisis
Pekerjaan Penghijauan	01 Mar 2022	74 hari lagi	Analisis

Screenshot 2: Hasil Analisis Keterlambatan (What If Analysis)

This screenshot shows the "Detail Proyek" page under the "Analisis" tab, specifically the "Menunggu Diverifikasi" (Waiting for Verification) section. It displays the results of a "What If Analysis" for the "Pekerjaan Pendahuluan" (Preparatory Work) activity. The table below provides details on how delaying this activity would affect other tasks.

Aktivitas	Percepatan (hari)	Kondisi Percepatan	Jumlah Penambahan Karyawan	Jumlah Penambahan Jam	Aksi
Pekerjaan Persiapan	7	✓	5	8	Analisis
Pekerjaan Konstruksi	63	✓	1	1	Analisis
Pekerjaan Mekanikal dan Plumbing	56	✓	1	1	Analisis
Pekerjaan Elektrikal	49	✓	1	1	Analisis
Perlengkapan Penunjang	182	✗	0	0	Analisis

18. Tugas Table Interface

Tugas ID	Proyek	Judul	Deadline	PIC	Status	Aksi
TGS02220001	Perencanaan Pedestrian Blok M	Survey pedestrian blok m	10 Dec 2021 16:21:00	Safira Noor Fauziah	Selesai	<button>Ubah</button> <button>Hapus</button>
TGS02220002	Audit Uji Coba Aplikasi	Survey ke Gedung BNI Kota Tua	09 Feb 2022 16:26:00	Safira Noor Fauziah	Butuh Koreksi	<button>Ubah</button> <button>Hapus</button>
TGS02220003	Perencanaan Demo Aplikasi	Survey Aplikasi	07 Feb 2022 16:31:00	Safira Noor Fauziah	Selesai	<button>Ubah</button> <button>Hapus</button>
TGS02220004	Audit Uji Coba Aplikasi	Analisis Hasil Survey	11 Feb 2022 16:34:00	Safira Noor Fauziah	Butuh Koreksi	<button>Ubah</button> <button>Hapus</button>
TGS02220005	Audit Uji Coba Aplikasi	Analisis Sample Laboratorium	15 Feb 2022 16:35:00	Fahira Nafiah Azhaar	Belum Dikerjakan	<button>Ubah</button> <button>Hapus</button>
TGS02220006	Perencanaan Demo Aplikasi	Persiapan Demo Aplikasi	17 Feb 2022 16:37:00	Safira Noor Fauziah	Selesai	<button>Ubah</button> <button>Hapus</button>
TGS02220007	Audit Uji Coba Aplikasi	Analisis uji coba aplikasi	24 Feb 2022 16:45:00	Safira Noor Fauziah	Sedang Dikerjakan	<button>Ubah</button> <button>Hapus</button>

Show 10 ▼ First < Page 1 of 1 > Last Go to page 1

19. Form Tugas Interface

MASTER Form Tugas

Proyek *

Aktivitas *

Jenis *

Deadline *

PIC *

Judul *

Deskripsi

Kembali Simpan

20. Form Hasil Tugas Interface

The screenshot shows a web-based application interface for managing tasks. On the left, there is a vertical sidebar with navigation links: Dashboard, Tugas (selected), Pesan, Profil, and Keluar. The main content area is titled "MASTER Form Tugas". It contains several input fields:

- Proyek *: Audit Uji Coba Aplikasi
- Aktivitas *: E
- Jenis *: Analisis
- Deadline *: 02/24/2022 04:45 PM
- PIC *: Safira Noor Fauziah
- Judul *: Analisis uji coba aplikasi
- Deskripsi: analisis aplikasi

Below these fields is a green button labeled "Sedang Dikerjakan". Further down, there is a section titled "Kirim Hasil Tugas" with fields for "Judul Hasil *" and "File*", which includes a "Choose Files" button. At the bottom of this section is a blue button labeled "Selesai". Below the "Selesai" button is a "Kembali" button.

21. Laporan Table Interface

The screenshot shows a table-based reporting interface. The left sidebar has links for Dashboard, Laporan (selected), Pesan, Profil, and Keluar. The main area is titled "Laporan" and displays a table of reports:

Laporan ID	Judul	Tipe	Status	Aksi
LAP02220001	Laporan Harian	Harian	Diverifikasi	<button>Ubah</button> <button>Hapus</button>
LAP02220002	Laporan Bulanan	Bulanan	Diverifikasi	<button>Ubah</button> <button>Hapus</button>
LAP02220003	Laporan Akhir Perencanaan Pedestrian Blok M	Akhir	Diverifikasi	<button>Ubah</button> <button>Hapus</button>

At the bottom of the table area, there are pagination controls: "Show 10", "First", "< Page 1 of 1 > Last", and "Go to page 1".

22. Form Laporan *Interface*

The screenshot shows the 'Form Laporan' (Report Form) interface for the project 'Perencanaan Pedestrian Blok M'. The form includes fields for Project (Perencanaan Pedestrian Blok M), Title (Laporan Akhir Perencanaan Pedestrian Blok M), and Type (Akhir). A preview window displays the report's content:

LAPORAN PERENCANAAN TEKNIS

PROGRAM	:	Program Proyek
KEGIATAN	:	Kegiatan Proyek
SUB KEGIATAN	:	Sub Kegiatan Proyek
RINCIAN SUB KEGIATAN	:	Rincian Sub Kegiatan Proyek
NAMA PAKET	:	Perencanaan Pedestrian Blok M
KODE REKENING	:	1234 Pedestrian Blok M
LOKASI KEGIATAN	:	JAKARTA SELATAN
TAHUN ANGGARAN	:	2022

PT TRIPRIMA KARYA
Anggaran Tahun 2022

23. Daftar Pesan *Interface*

PT TRIPRIMA KARYA

Pesan

Cari Pengguna

teamleader
Test pesan pertama untuk teamleader
2021-12-19 03:06:38
Dibaca

teamleader
Test pesan kedua untuk teamleader dari admin nih
2021-12-19 03:44:30
Dibaca

tenagaahli
test dari admin untuk tenaga ahli
2021-12-19 03:44:57
Dibaca

teamleader
Test pesan balasan dari team leader untuk admin
2021-12-19 04:15:58
Dibaca

teamleader
pesan balasan kedua dr team leader untuk admin
2021-12-19 04:21:08
Dibaca

teamleader
admin membalas team leader
2021-12-19 04:58:29
Terkirim

tenagaahli
Baik pak mohon maaf
2021-12-19 05:03:06
Pesanan Baru

24. Pesan Detail *Interface*

PT TRIPRIMA KARYA

Detail Pesan

2021-12-19 03:06:38

Anda
Test pesan pertama untuk teamleader

2021-12-19 03:44:30

Anda
Test pesan kedua untuk teamleader dari admin nih

2021-12-19 04:15:58

teamleader
Test pesan balasan dari team leader untuk admin

2021-12-19 04:21:08

teamleader
pesan balasan kedua dr team leader untuk admin

2021-12-19 04:58:29

Anda
admin membalas team leader

Pesan untuk teamleader
Ketik pesan...

Batal

Kirim

25. Profil *Interface*



LAMPIRAN IV

SOURCE CODE

Proyek.js

```
import React, {useState, useEffect} from "react";
import {
  Container,
  Row,
  Col,
  Card,
  CardHeader,
  CardBody,
  Button,
  Badge
} from "shards-react";

import PageTitle from
"../../components/common/PageTitle";
import DataTable from
"../../components/common/Table";

import moment from "moment";
import Swal from 'sweetalert2'
import withReactContent from
'sweetalert2-react-content'

const Proyek = ({ history }) =>
{
  const [data, setData] =
  useState([]);

  let user = [];
  let role = "";
  try {
    user =
    JSON.parse(localStorage.getItem
    ("userInfo"));
    role =
    user.data._role.role;
  } catch (e) {}

  useEffect(() => {
    fetchData();
  }, []);

  async function fetchData() {

    let result = await
    fetch("http://localhost:8000/ap
    i/proyek");
    result = await
    result.json();
    setData(result);
  }

  function doTambah(){
    localStorage.setItem('dataP
    royek', JSON.stringify({
      durasi: "",
      kegiatan: "",
      klien_id: "",
      kode_rekening: "",
      lokasi: "",
      nama_paket: "",
      program: "",
      proyek_id: "",
      rincian_sub_kegiatan: "",
      status: "",
      sub_kegiatan: "",
      tahun: "",
      tipe: "",
      waktu_mulai: "",
      waktu_selesai: "",
      _klien: ""
    }));
    localStorage.setItem('klien
    Detail', JSON.stringify([]));
    const readOnly = "tambah";
    localStorage.setItem('actio
    n', JSON.stringify(readOnly));
    history.push("/proyekForm")
  }

  async function doLihat(data){
    // console.log('cell',data)
    localStorage.setItem('dataP
    royek', JSON.stringify(data));

    const readOnly = "lihat";
    localStorage.setItem('actio
    n', JSON.stringify(readOnly));
  }
}
```

```

        history.push("/proyekForm")
    }

    async function doEdit(data){
        localStorage.setItem('dataP
royek', JSON.stringify(data));

        let klien = await
fetch("http://localhost:8000/ap
i/klien");
        klien = await klien.json();
        const klienDetail =
klien.filter(item =>
item.klien_id ===
data.klien_id);
        const newDetailKlien =
klienDetail.map(({ klien_id,
nama_perusahaan }) => ({
            label: nama_perusahaan,
            value: klien_id
        }));
        localStorage.setItem('klien
Detail',
JSON.stringify(newDetailKlien))
;

        const readOnly = "edit";
        localStorage.setItem('actio
n', JSON.stringify(readOnly));
        history.push("/proyekForm")
    }

    function doDelete(id){
        const Alert =
withReactContent(Swal)
        Alert.fire({
            title: 'Yakin?',
            text: "Yakin ingin
menghapus data?",
            icon: 'warning',
            showCancelButton: true,
            // confirmButtonColor:
            '#3085d6',
            // cancelButtonColor:
            '#d33',
            confirmButtonText: 'Iya',
            cancelButtonText:
'Batal',
            preConfirm: () => {
                return
            }
        })
        fetch(`http://localhost:8000/ap
i/delete/proyek/${id}`, {
            method: "DELETE",
        })
        .then(response => {
            if (!response.ok) {
                throw new
Error(response.statusText)
            }
            return
        })
        .catch(error => {
            Swal.showValidation
Message(
`Request failed:
${error}`
)
        })
        .then((result) => {
            if (result.isConfirmed) {
                Swal.fire(
                    'Berhasil!',
                    'Berhasil menghapus
data!',
                    'success'
                )
                fetchData();
            }
        })
    }

    const columns =
React.useMemo(
() => [
{
    Header: "Proyek ID",
    accessor: "proyek_id"
},
{
    Header: "Nama Paket",
    accessor: "nama_paket"
},
{
    Header: "Tipe",
    accessor: "tipe"
},
{
    Header: "Lokasi",
    accessor: "lokasi"
},
{
    Header: "Waktu Mulai",
    accessor: "waktu_mulai"
}
])

```

```

        accessor: "waktu_mulai"
    },
    {
        Header: "Waktu
Selesai",
        accessor:
"waktu_selesai"
    },
    {
        Header: "Status",
        accessor:(row) =>
        row.status === "0"  &&
row.waktu_mulai <
moment().format('YYYY-MM-DD
HH:mm:ss') ? "Sedang
Berlangsung" :
        row.status ===
"0"  && row.waktu_mulai >
moment().format('YYYY-MM-DD
HH:mm:ss') ? "Belum
Berlangsung" :
        row.status === "1"
? "Selesai" : "",
        Cell: ({ value }) => (
            <Badge theme={
                value === "Sedang
Berlangsung" ? "primary" :
                value === "Belum
Berlangsung" ? "secondary" :
                value === "Selesai"
? "success" : "light"
            }>
                {value}
            </Badge>
        )
    },
    {
        id: role != 'Admin' ?
"aksi" : "",
        Header: "Aksi",
        // accessor:
"progress",
        Cell: ({ cell }) => (
            <div>
                <Button
theme="info" className="mb-1
mr-1" onClick={(e) =>
{e.stopPropagation();
doEdit(cell.row.original)}}>
                    Ubah
                </Button>
            <Button
theme="danger" className="mb-1
mr-1" onClick={(e) =>
{e.stopPropagation();
doDelete(cell.row.values.proyek
_id)}}>
                    Hapus
                </Button>
            </div>
        )
    );
}

// const data =
React.useMemo(() =>
makeData(1000), []);

return (
    <Container fluid
className="main-content-
container px-4">
    /* Page Header */
    <Row noGutters
className="page-header py-4">
        <PageTitle
sm="4"
title="Proyek"
subtitle="Master"
className="text-sm-left"
/>
    </Row>
    /* Default Light Table
*/
    <Row>
        <Col>
            <Card small
className="mb-4 overflow-
hidden">
                <CardBody>
                    <DataTable
columns={column
s}>
                        <tbody>
                            <tr>
                                <td>{data}>
                                <td>{tambah}>
                                <td>{role}>
                                <td>{status}>
                                <td>{aksi}>
                            </tr>
                        </tbody>
                    </DataTable>
                </CardBody>
            </Card>
        </Col>
    </Row>
)

```

```

        doLihat={(data)
=> doLihat(data)}
        />
        <React.Fragment>
        </React.Fragment>
        </CardBody>
        </Card>
        </Col>
        </Row>
        </Container>
    );
    // }
};

export default Proyek;

Proyek Controller
<?php

namespace App\Http\Controllers;
use App\Models\Proyek;
use App\Models\Posisi;

use App\Models\Jadwal;
use App\Models\Tim;
use App\Models\Laporan;

use App\Models\Aktivitas;
use App\Models\Successor;
use App\Models\Cpm;
use App\Models\WhatIf;
use App\Models\WhatIfDetail;
use App\Models\User;
use App\Models\Notifikasi;

use Illuminate\Http\Request;
use Haruncpi\LaravelIdGenerator\IdGenerator;
use Carbon\Carbon;

class ProyekController extends Controller
{
    function index(){
        $data =
        Proyek::orderBy('created_at',
        'desc')->with('_klien')->get();

        return response()->json($data);
    }

    doLihat=(data)
    />
    <React.Fragment>
    </React.Fragment>
    </CardBody>
    </Card>
    </Col>
    </Row>
    </Container>
};

function create(Request
$request) {
    $this-
    >validate($request,
    Proyek::$rules);

    $prefix =
    "PRY".date('my');
    $data = [
        'proyek_id' =>
        IdGenerator::generate(['table'
=>
        'proyek','field'=>'proyek_id',
        'length' => 11, 'prefix' =>
        $prefix,
        'reset_on_prefix_change'=>'true
        ]),
        'klien_id' =>
        $request->klien_id,
        'program' =>
        $request->program,
        'kegiatan' =>
        $request->kegiatan,
        'sub_kegiatan' =>
        $request->sub_kegiatan,
        'rincian_sub_kegiatan' =>
        $request->rincian_sub_kegiatan,
        'tipe' => $request-
        >tipe,
        'nama_paket' =>
        $request->nama_paket,
        'kode_rekening' =>
        $request->kode_rekening,
        'lokasi' =>
        $request->lokasi,
        'tahun' =>
        $request->tahun,
        'durasi' =>
        $request->durasi,
        'waktu_mulai' =>
        $request->waktu_mulai,
        'waktu_selesai' =>
        $request->waktu_selesai,
        'status'=> 0
    ];
}

```

```

        $data =
Proyek::create($data);

        $findUser =
User::whereIn('role_id',
['ROL1121002', 'ROL1121003',
'ROL1121006'])->get();

        foreach($findUser as
$user){
            $prefix =
"NTF".date('my');
            $notif = [
                'notifikasi_id' =>
IdGenerator::generate(['table' =>
'notifikasi','field'=>'notifikasi_id', 'length' => 11,
'prefix' => $prefix,
'reset_on_prefix_change'=>'true
']),
                'user_id' =>
$user->user_id,
                'created_user' =>
$request->created_user,
                'name' =>
'Proyek',
                'description' =>
'Admin menambah proyek '.
$data->nama_paket,
                'is_read' => 0,
];
            $notif =
Notifikasi::create($notif);
}

        return response()->json([
            'success' => true,
            'message' =>
'Berhasil',
            'data'=> $data,
]);
}

        function update(Request
$request, $id) {
    $this-
>validate($request,
Proyek::rules_update($id));
    $prefix =
"PRY".date('my');
    $data = [
        'klien_id' =>
$request->klien_id,
        'program' =>
$request->program,
        'kegiatan' =>
$request->kegiatan,
        'sub_kegiatan' =>
$request->sub_kegiatan,
        'rincian_sub_kegiatan' =>
$request->rincian_sub_kegiatan,
        'tipe' => $request->tipe,
        'nama_paket' =>
$request->nama_paket,
        'kode_rekening' =>
$request->kode_rekening,
        'lokasi' =>
$request->lokasi,
        'tahun' =>
$request->tahun,
        'durasi' =>
$request->durasi,
        'waktu_mulai' =>
$request->waktu_mulai,
        'waktu_selesai' =>
$request->waktu_selesai,
        'status'=> 0
];
    $data =
Proyek::where('proyek_id',$id)->update($data);

    return response()->json([
            'success' => true,
            'message' =>
'Berhasil',
            'data'=> $data,
]);
}

        function delete($id){
    $data =
Proyek::where('proyek_id',$id)->first();
}

```

```

        if($data){
            $data =
Proyek::where('proyek_id',$id)->delete();
            return response()->json([
                'success' =>
true,
                'message' =>
'Berhasil menghapus data!', // 'data'=>
$detail
            ]);
        } else {
            return response()->json([
                'success' =>
false,
                'message' =>
'Gagal menghapus data!', ]);
        }
    }

    function tutup(Request $request, $id) {
        $proyek =
Proyek::where('proyek_id',$id)->first();

        $condition1 = $proyek->waktu_mulai <= Carbon::now();
        $jadwal =
Jadwal::where('proyek_id',$id)->first();
        if($jadwal){
            $statusJadwal =
$jadwal->status == 1;
        } else {
            $statusJadwal =
false;
        }
        $tim =
Tim::where('proyek_id',$id)->first();
        if($tim){
            $statusTim = $tim->status == 1;
        } else {
            $statusTim = false;
        }
        $laporan =
Laporan::where('proyek_id',$id)->where('tipe', 'Akhir')->first();
        if($laporan){
            $statusLaporan =
$laporan->status == 1;
        } else {
            $statusLaporan =
false;
        }
        if($condition1 &&
$jadwal && $statusJadwal &&
$tim && $statusTim && $laporan && $statusLaporan){
            $data = [
                'status'=> 1
            ];
            $data =
Proyek::where('proyek_id',$id)->update($data);

            return response()->json([
                'success' => true,
                'message' =>
'Berhasil',
                'data'=> $data,
            ]);
        } else {
            if(!$condition1){
                $errors[] =
"Proyek belum dimulai";
            }
            if($jadwal ==
null){
                $errors[] =
"Jadwal belum dibuat";
            }
            if(!$statusJadwal){
                $errors[] =
"Jadwal belum diverifikasi";
            }
            if($tim == null){

```

```

        $errors[] =
"Tim belum dibuat";
    }
    if(!$statusTim){
        $errors[] =
"Tim belum diverifikasi";
    }
    if($laporan ==
null){
        $errors[] =
"Laporan Akhir belum dibuat";
    }
    if(!$statusLaporan)
{
        $errors[] =
"Laporan Akhir belum
diverifikasi";
    }
}
return response()->json($data);
}

function index_cpm_detail($id){
    $data =
Cpm::where('cpm_id',$id)
->with('_aktivitas._successor')
->get();
return response()->json($data);
}

function analisis_cpm($id){
    $jadwal =
Jadwal::where('proyek_id',
$id)->first();
if($jadwal){

    $aktivitas =
Aktivitas::where('jadwal_id',
$jadwal->jadwal_id)-
>orderBy('aktivitas_id', 'asc')-
>with('_successor')-
>get();
// return $aktivitas;
foreach($aktivitas as
$act){
    if($act-
>_successor->isEmpty()){
        $pendahulu[$act-
>aktivitas_id] = $act-
>_successor;
        $es[$act-
>aktivitas_id] = 0;
        $ef[$act-
>aktivitas_id] = $es[$act-
>aktivitas_id] + $act->durasi;
    } else {
        $succArr =
$act->_successor;
        // return
        $succArr;
    }
}
}
}

```

```

        $succTotal =
$act->_successor->count();
        $succName =
$act->_successor-
>pluck('successor');
        $pendahulu[$act-
>aktivitas_id] = $succName;
        // $ef = "";
        // $test[$act-
>aktivitas_id] = $succArr;
        foreach($succAr-
r as $value){
            if($succTot-
al > 1){
                $arr[$v-
alue->aktivitas_id][] =
$ef[$value->successor];
                $max[$a-
ct->aktivitas_id] =
max($arr[$value-
>aktivitas_id]);
                $es[$ac-
t->aktivitas_id] = $max[$act-
>aktivitas_id];
            } else {
                $es[$ac-
t->aktivitas_id] = $ef[$value-
>successor];
            }
            $ef[$act-
>aktivitas_id] = $es[$act-
>aktivitas_id] + $act->durasi;
        }
    }
    // return $max;

    $fromLastAktivitas =
Aktivitas::where('jadwal_id',
$jadwal->jadwal_id)
->with('_successor')-
>orderBy('aktivitas_id',
'DESC')->get();
// return $fromLastAktivitas;
foreach($fromLastAktivitas as $act){
        $succ =
Successor::where('successor',
$act->aktivitas_id)->get();
        $suc[$act-
>aktivitas_id] = $succ;
        if($suc[$act-
>aktivitas_id]->isEmpty()){
            $max =
max($ef);
            $lf[$act-
>aktivitas_id] = $max;
        } else {
            $succArr =
$suc[$act->aktivitas_id];
            $succTotal =
$succArr->count();
            $succName =
$succArr-
>pluck('aktivitas_id');
            $test[$act-
>aktivitas_id] = $succName;
            foreach($succNa-
me as $value){
                //
                $aktivitas =
Aktivitas::where('aktivitas_id',
$value)->first();
                if($succTot-
al > 1){
                    $arrs[$a-
ct->aktivitas_id][] =
$ls[$value];
                    //
                    $arr[$value] = $ls[$value];
                    $min[$a-
ct->aktivitas_id] =
min($arrs[$act->aktivitas_id]);
                    $lf[$ac-
t->aktivitas_id] = $min[$act-
>aktivitas_id];
                    //
                    $lf[$act->aktivitas_id] = 6;
                } else {
                    $lf[$ac-
t->aktivitas_id] = $ls[$value];
                }
            }
        }
}

```

```

        $ls[$act-
>aktivitas_id] = $lf[$act-
>aktivitas_id] - $act->durasi;
    }
    // return [$suc, $arrs,
$min];
    $findAktivitas =
Aktivitas::where('jadwal_id',
$jadwal->jadwal_id)->get();

        foreach($findAktivitas
as $akt){
            $prefix_cpm =
"CPM".date('my');
            $float = $ls[$akt-
>aktivitas_id] - $es[$akt-
>aktivitas_id];
            if($float != 0){
                $critical = 0;
            } else {
                $critical = 1;
            }
            $cpm = [
                'cpm_id'=>
IdGenerator::generate([
                    'table' =>
'cpm',
                    'cpm_id',
                    'field'=>'c
11,
                    'length' =>
$prefix_cpm,
                    'prefix' =>
'refix_change'=>'true'
                ],
                'aktivitas_id'=>
$akt->aktivitas_id,
                'es' =>
$es[$akt->aktivitas_id],
                'ef' =>
$ef[$akt->aktivitas_id],
                'ls' =>
$ls[$akt->aktivitas_id],
                'lf' =>
$lf[$akt->aktivitas_id],
                'float' =>
$float,
                'critical' =>
$critical,
            ];
}
$cpm =
Cpm::create($cpm);
}

$data = [
    'es' => $es,
    'ef' => $ef,
    'ls' => $ls,
    'lf' => $lf,
    'pendahulu' =>
$pendahulu,
    'suc' => $suc,
    'success' => true,
];
return response()->json([
    'aktivitas'=>
$aktivitas,
    'data'=> $data,
    'cpm' => $cpm,
    'success' => true,
    'message' =>
'Berhasil',
]);
}

function
getSuccessor($aktivitas_nama) {
    $succ =
Successor::where('successor',
$aktivitas_nama)->get();
    $subCategories = [];
    foreach($succ as $i =>
$item){
        if($item['successor
'] == $aktivitas_nama){
            $children =
$this->getSuccessor($item['aktivitas_
id']));
            if ($children)
{
                $item['chil
dren'] = $children;
            }
            $subCategories[
] = $item;
        }
    }
}

```

```

        return $subCategories;
    }

    function
array_single_dimensional($items)
{
    $singleDimensional =
[];
    foreach ($items as
$item) {
        $children
= isset($item['children']) ?
$item['children'] : null;
//temporarily store children if
set
        unset($item['chil
dren']); //delete children
before adding to new array
        $singleDimensiona
l[] = $item; // add parent to
new array
        if (
!empty($children) ){ // if has
children
            //convert
children to single dimensional
            $childrenSing
leDimensional = $this-
>array_single_dimensional($chil
dren);

            //merge the
two, this line did the trick!
            $singleDimens
ional =
array_merge($singleDimensional,
$childrenSingleDimensional);
        }
    }
    return
$singleDimensional;
}

function index_whatif(){
    $data =
WhatIf::where('status', '!=',
0)
        // ->where('critical',
1)
        // ->orderBy('cpm_id',
'asc')

        ->with('_detail',
'_cpm._aktivitas._jadwal._proye
k')
        ->get();
    return response()->
json([
    'success' => true,
    'data'=> $data
]);
// } else {
//     return
response()->json([]);
// }
}

function
analisis_whatif(Request
$request, $id){
    // return $id;
    $cpm =
Cpm::where('cpm_id', $id)->
with('_aktivitas._successor')-
>first();

    $deadline_rencana =
Carbon::parse($cpm->_aktivitas-
>waktu_selesai);
    $deadline_realisasi =
Carbon::parse($request-
>deadline_realisasi);

    $delay =
$deadline_rencana-
>diffInDays($deadline_realisasi
);
    // $delay = 0.7;
    $delayp = $delay -
$cpm->float;
    $cond_delay = $delayp >
0;

    $this-
>validate($request,
WhatIf::$rules);

    $prefix_what_if =
"WIF".date('my');
    $what_if = [
        'whatif_id'=>
IdGenerator::generate([
            'table' =>
'what_if',

```

```

        'field'=>'w
hatif_id',
                    'length' =>
11,
                    'prefix' =>
$prefix_what_if,
                    'reset_on_p
refix_change'=>'true'
                ],
                'cpm_id' =>
$id,
                'delay' =>
$delay,
                'delay_p' =>
$delayp,
                'kondisi_delay'
=> $cond_delay,
                'status' => 0,
                'deadline_reali
sasi' => $request-
>deadline_realisasi,
                'pekerja_max'=>
$request->pekerja_max,
                'jam_max'=>$req
uest->jam_max,
];
                $what_if =
WhatIf::create($what_if);
                if($cond_delay){
                    // $aktivitas =
Aktivitas::where('jadwal_id',
$cpm->aktivitas->jadwal_id)->
select('aktivitas_id')->get();
                    $akt = $this-
>getSuccessor($cpm->aktivitas-
>aktivitas_id);
                    $aktivitas_terpenga
ruh = $this-
>array_single_dimensional($akt)
;
                    $percepatan = [];
//d's aktivitas
                    $f = []; //float
                    $d = []; //durasi
                    $cond1 = []; //d's
< durasi
                    $cond2 = [];
//1/2durasi > delayp

```

```

                $add_employee = [];
//penambahan pekerja
                $add_hour = [];
//penambahan jam
                $emp_opt = [];
//kondisi penambahan pekerja
                $hour_opt = [];
//kondisi penambahan jam
                if(!empty($aktivita
s_terpengaruh)){
                    $unique_array =
[];
                    foreach($aktivi
tas_terpengaruh as $element) {
//supaya kegiatan yang sama
tidak dobel
                        $hash =
$element['aktivitas_id'];
                        $unique_arr
ay[$hash] = $element;
}
                    $result =
array_values($unique_array);
                    foreach($result
as $key => $item){
                        $suc =
Cpm::where('aktivitas_id',
$item->aktivitas_id)
->
with('_aktivitas')
->first();
                        $f[$suc-
>aktivitas->aktivitas] = $suc-
>float;
                        $d[$suc-
>aktivitas->aktivitas] = $suc-
>aktivitas->durasi;
                        $ds = $suc-
>aktivitas->durasi; //durasi
kegiatan
                        $float = $suc-
>float; //float kegiatan
                        $H = $suc-
>aktivitas->waktu_normal;
                        $n = $suc-
>aktivitas->total_pic;
                        $mh =
$ds*$H*$n;

```

```

        $durasi_percepa
tan = $ds + $float - $delayp;

        $percepatan[$su
c->_aktivitas->aktivitas] =
$durasi_percepatan;

        $cond1[$suc-
>_aktivitas->aktivitas] =
$durasi_percepatan < $ds;
        $cond2[$suc-
>_aktivitas->aktivitas] =
$ds*0.5 >= $delayp;

        $cond_percep
atan[$suc->_aktivitas-
>aktivitas]{
        $add_employ
ee[$suc->_aktivitas->aktivitas]
= ($mh/($durasi_percepatan*$H))-n;
        $add_emp_op
t[$suc->_aktivitas->aktivitas]
= $add_employee[$suc-
>_aktivitas->aktivitas] + $n ;
        $emp_opt[$s
uc->_aktivitas->aktivitas] =
round($add_emp_opt[$suc-
>_aktivitas->aktivitas]) <
intval($request-
>pekerja_maksimum);

        $add_hour[$
suc->_aktivitas->aktivitas] =
($mh/($durasi_percepatan*$n))-H;
        $add_hour_o
pt[$suc->_aktivitas->aktivitas]
= $add_hour[$suc->_aktivitas-
>aktivitas] + $H;
        $hour_opt[$
suc->_aktivitas->aktivitas] =
round($add_hour_opt[$suc-

```

```

>_aktivitas->aktivitas]) <
intval($request->jam_maksimum);

    } else {
        $add_employ
ee[$suc->_aktivitas->aktivitas]
= 0;
        $add_hour[$
suc->_aktivitas->aktivitas] =
0;
        $emp_opt[$s
uc->_aktivitas->aktivitas] =
false;
        $hour_opt[$
suc->_aktivitas->aktivitas] =
false;
    }

    $prefix_whatif_
detail = "WFD".date('my');
    $whatif_detail
= [
    'whatif_det
ail_id'=>
IdGenerator::generate([
    'table'
=> 'what_if_detail',
    'field'
=>'whatif_detail_id',
    'length'
' => 11,
    'prefix'
' => $prefix_whatif_detail,
    'reset_
on_prefix_change'=>'true'
]),
    'whatif_id'
=> $what_if->whatif_id,
    'aktivitas_
id' => $suc->_aktivitas-
>aktivitas,
    'percepatan'
' => $percepatan[$suc-
>_aktivitas->aktivitas],
    'kondisi_pe
rcepatan' =>
$cond_percepatan[$suc-
>_aktivitas->aktivitas],
    'additonal_
employee' =>

```

```

$add_employee[$suc->aktivitas->aktivitas],
                    'additonal_
hour' => $add_hour[$suc->aktivitas->aktivitas],
                    'kondisi_em
ployee' => $emp_opt[$suc->aktivitas->aktivitas],
                    'kondisi_ho
ur' => $hour_opt[$suc->aktivitas->aktivitas],
                    ];
                    $whatif_detail
=
WhatIfDetail::create($whatif_de
tail);
}
}
return response()->json([
        'float' =>
$cpm->float,
        '$cpm-
>aktivitas->aktivitas'=>$cpm-
>aktivitas->aktivitas,
        'delayp' =>
$delayp,
        'aktivitas_terp
engaruh' =>
$aktivitas_terpengaruh,
        'dS' =>
$percepatan,
        'f' => $f,
        'd' => $d,
        'cond1' =>
$cond1,
        'cond2' =>
$cond2,
        'cond' =>
$cond_percepatan,
        'add_employee' => $add_employee,
        'add_hour' =>
$add_hour,
                    // 'emp_opts'
=> $add_emp_opt,
                    // 'hour_opts'
=> $add_hour_opt,
                    'emp_opt' =>
$emp_opt,
                    'hour_opt' =>
$hour_opt,
                    'suc' => $suc,
                    'success' =>
true,
                ]);
}
return response()->json([
        // 'float' => $cpm-
>float,
        // 'delayp' =>
$delayp,
        'message' => 'Tidak
mempengaruhi keterlambatan
proyek',
        'success' => true,
    ]);
}

function
ajukan_analisis(Request
$request, $id){
    $data = [
        'status' => 1,
    ];
    $data =
WhatIf::where('whatif_id',$id)->update($data);
    $whatif =
WhatIf::where('whatif_id',$id)->first();
    $findUser =
User::where('role_id',
'ROL11210002')->get();
    foreach($findUser as
$user){
        $prefix =
"NTF".date('my');
        $notif = [
            'notifikasi_id' =>
IdGenerator::generate(['table' =>
'notifikasi','field'=>'notifikasi_id', 'length' => 11,
'prefix' => $prefix,

```

```

'reset_on_prefix_change'=>'true
']),
        'user_id' =>
$user->user_id,
            'created_user' =>
$request->created_user,
            'name' =>
'Analisis',
            'description' =>
'Terdapat analisis keterlambatan ('.$whatif->whatif_id.') proyek yang belum diverifikasi',
            'is_read' => 0,
];
$notif =
Notifikasi::create($notif);
}

return response()->json([
    'success' => true,
    'message' =>
'Berhasil',
    'data'=> $data,
]);
}

function verifikasi_analisis(Request $request, $id){
    $data = [
        'status' => 2,
    ];

    $data =
WhatIf::where('whatif_id',$id)->update($data);

    $whatif =
WhatIf::where('whatif_id',$id)->first();
    $findUser =
User::where('role_id',
'ROL1121003')->get();

    foreach($findUser as
$user){
        $prefix =
"NTF".date('my');
        $notif = [
            'notifikasi_id' =>
IdGenerator::generate(['table' =>
'notifikasi','field'=>'notifikasi_id', 'length' => 11,
'prefix' => $prefix,
'reset_on_prefix_change'=>'true
']),
            'user_id' =>
$user->user_id,
            'created_user' =>
$request->created_user,
            'name' =>
'Analisis',
            'description' =>
'Hasil analisis '.$whatif->whatif_id.' diverifikasi',
            'is_read' => 0,
];
$notif =
Notifikasi::create($notif);
}

return response()->json([
    'success' => true,
    'message' =>
'Berhasil',
    'data'=> $data,
]);
}

function tolak_analisis(Request $request, $id){
    $data = [
        'catatan' =>
$request->catatan,
        'status' => 3,
    ];

    $data =
WhatIf::where('whatif_id',$id)->update($data);

    $whatif =
WhatIf::where('whatif_id',$id)->first();
}

```

```
$findUser =  
User::where('role_id',  
'ROL11210003')->get();  
  
foreach($findUser as  
$user){  
    $prefix =  
"NTF".date('my');  
    $notif = [  
        'notifikasi_id'  
=>  
IdGenerator::generate(['table'  
=>  
'notifikasi', 'field'=>'notifikasi_id', 'length' => 11,  
'prefix' => $prefix,  
'reset_on_prefix_change'=>'true  
]),  
        'user_id' =>  
$user->user_id,  
        'created_user'  
=> $request->created_user,  

```

Source Code lengkap bisa diakses di

Front-end: https://gitlab.com/farahdhiay/simp_frontend

Back-end: https://gitlab.com/farahdhiay/simp_backend