

**SISTEM INFORMASI MANAJEMEN SALON
(STUDI KASUS QORINAH DAY SPA AND SALON)**

TUGAS AKHIR

**Diajukan Sebagai Salah Satu Syarat
Untuk Memperoleh Gelar Sarjana Jurusan Teknik Informatika**



Disusun oleh :

Nama : Rabiaini Dwi Citra

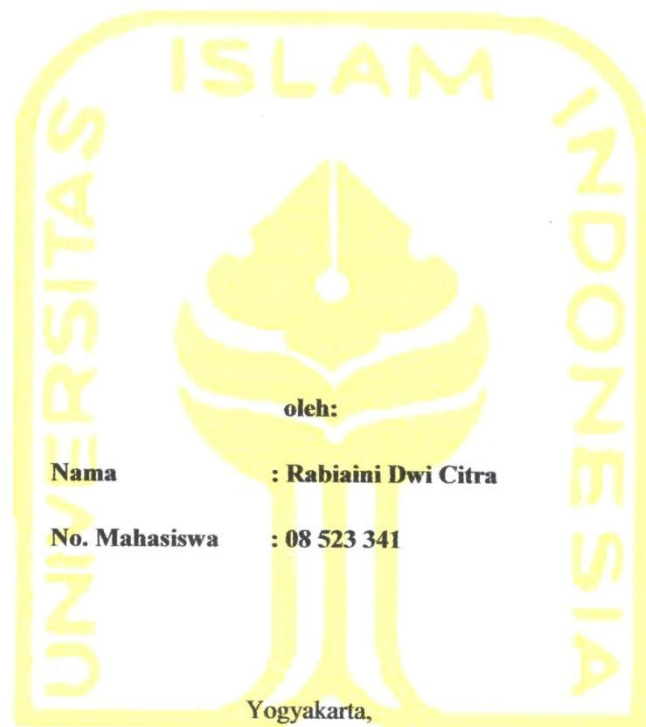
NIM : 08523341

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA**

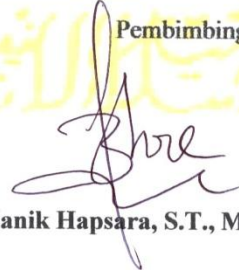
2014

LEMBAR PENGESAHAN DOSEN PEMBIMBING
SISTEM INFORMASI MANAJEMEN SALON

TUGAS AKHIR



Pembimbing,


Manik Hapsara, S.T., M.Sc., Ph.D.

LEMBAR PERNYATAAN KEASLIAN HASIL TUGAS AKHIR

Saya yang bertandatangan di bawah ini,

Nama : Rabiaini Dwi Citra

No. Mahasiswa : 08 523 341

Menyatakan bahwa seluruh komponen dan isi dalam Laporan Tugas Akhir ini adalah hasil karya saya sendiri. Apabila di kemudian hari terbukti bahwa ada beberapa bagian dari karya ini adalah bukan hasil karya saya sendiri, maka saya siap menanggung risiko dan konsekuensi apapun.

Demikian pernyataan ini saya buat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta,

Rabiaini Dwi Citra

**LEMBAR PENGESAHAN DOSEN PENGUJI
SISTEM INFORMASI MANAJEMEN SALON**

TUGAS AKHIR

oleh:

Nama : Rabiaini Dwi Citra

No. Mahasiswa : 08523341

Telah Dipertahankan di Depan Sidang Penguji sebagai Salah Satu Syarat
untuk Memperoleh Gelar Sarjana Jurusan Teknik Informatika Fakultas
Teknologi Industri Universitas Islam Indonesia
Yogyakarta,

Tim Penguji,

Ketua

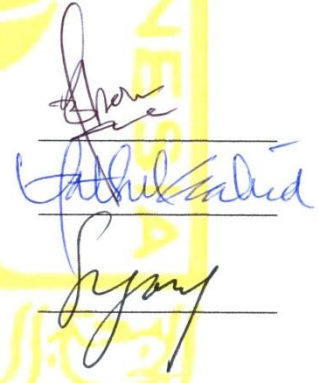
Manik Hapsara, S.T, M.Sc, Ph.D

Anggota I

Fathul Wahid, S.T, M.Sc, Ph.D

Anggota II

Syarif Hidayat, S.Kom, MIT



Mengetahui,

Ketua Jurusan Teknik Informatika

Fakultas Teknologi Industri

Universitas Islam Indonesia



Yudi Prayudi, S.Si., M.Kom

HALAMAN PERSEMBAHAN

Tugas akhir ini saya persembahkan untuk :

- Qorinah Day Spa and Salon yang sudah memberikan kesempatan untuk melakukan penelitian
- Almamater Universitas Islam Indonesia, khususnya jurusan Teknik Informatika
- Ayah Cipto dan Mama Imah yang selalu memberikan dukungan, kasih sayang, dan semangat yang tiada batasnya. Terima kasih selalu buat do'anya.
- Mbak Ika, Abang Dimas, dan Adek Rifah terima kasih atas do'a dan dukungannya.
- Sahabat – sahabat kesayanganku : Sajida, Uluth, Dandut, Wastika, Aa Rian, Abang Rio, Pahe, Syukron, Aulia Curtis, dan yang lainnya. Kalian semua luar biasa!! Terima kasih atas do'a dan dukungannya. Terima kasih untuk canda, tawa, dan memori yang sudah kita buat.

HALAMAN MOTTO

“Hidup adalah soal keberanian. Menghadapi jang tanda tanja. Tanpa bisa kita mengerti. Tanpa kita bisa menawar. Terima dan hadapilah” (Soe Hok Gie, 1966)

“First they ignore you, then they laugh at you.

Then they fight you, then you win” (Mahatma Gandhi)

“You don’t need a shoulder to cry on when you have the ground to put your head down” (Anonim)

KATA PENGANTAR

Assalamualaikum Warahmatullahiwabarakatuh,

Alhamdulillahirobbilalamin, segala puji dan syukur penulis panjatkan kehadirat Allah SWT, yang telah melimpahkan rahmat dan hidayah-Nya. Sholawat dan salam kepada junjungan kita Nabi Muhammad SAW beserta keluarga dan para sahabat, sehingga penulis dapat menyelesaikan Tugas Akhir yang berjudul Sistem Informasi Manajemen Salon (Studi Kasus : Qorinah Day Spa and Salon).

Laporan tugas akhir ini disusun sebagai salah satu syarat agar memperoleh gelar Sarjana Teknik Informatika di Universitas Islam Indonesia dan sebagai sarana untuk dapat mempraktekkan ilmu dan teori yang telah diperoleh di selama menjalani masa studi di Jurusan Teknik Informatika.

Penulis menyampaikan ucapan terima kasih dan penghargaan yang setinggi-tingginya kepada semua pihak atas bantuan, bimbingan dan dukungan demi kelancaran penulisan Tugas Akhir kepada :

1. Bapak Manik Hapsara, S.T, M.Sc, Ph.D. sebagai dosen pembimbing yang telah memberikan perhatian, kesabaran, dan bimbingan selama pelaksanaan tugas akhir.
2. Bapak Yudi Prayudi, S.Si., M.Kom. selaku Ketua Jurusan Teknik Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia.
3. Bapak dan Ibu dosen yang telah memberikan ilmu selama menjalani studi di Jurusan Teknik Informatika.
4. Ayah Cipto, Mama Imah, Mbak Ika, Bang Dimas, dan Adik Rifah yang memberikan semangat, dukungan, dan kasih sayang yang tiada habisnya.
5. Uluth, Sajida, Wastika, Dandut, Mbak Zuka, Mbak Evi, Adis, dan Arlin yang selalu pengertian, memberikan motivasi dan semangat.
6. Teman seperjuanganku Aak, Uda Rio, Aulia, Pahe, Nunu, Syukron, Mima, Hanafi, Risma, Bella, Maya, Bundo Mayang, Renny, Ariy yang sudah banyak *sharing* tentang ilmu apapun, dan selalu memberikan motivasi. Terima kasih untuk canda tawa yang kita lewati.

7. Teman-teman SNIPER Informatika 2008, semoga kita selalu solid.

Semoga Allah SWT melimpahkan rahmat dan hidayahNya kepada semua pihak yang telah membantu dalam penyelesaian tugas akhir ini. Penulis menyampaikan permohonan maaf dan menyadari masih banyak kekurangan dalam penulisan laporan serta menerima kritik yang membangun untuk penyempurnaan di masa datang.

Semoga tugas akhir ini dapat bermanfaat bagi penulis dan pembaca serta pihak lain yang membutuhkan.

Wassalamualaikum Warahmatullahi Wabarakatuh.

Yogyakarta,

Penulis

ABSTRAKSI

Penelitian ini untuk membuat sistem informasi manajemen salon yang bertujuan menata arsip kegiatan salon, memasukkan data member, karyawan, dan membuat basis data salon. Hasil penelitian ini diharapkan dapat memberikan masukan bagi pihak-pihak terkait agar dapat memberikan pelayanan yang lebih baik, cepat dan mudah bagi para penggunanya. Subyek dari penelitian ini adalah Qorinah Day Spa and Salon. Hasil penelitian ini menunjukkan mempermudah dari manual kinerja ke sistem yang mempercepat proses kinerja di salon.

Sistem ini menggunakan model pengembangan perangkat lunak yang disebut *linear sequential model*. Metode ini dipilih karena paling umum digunakan dalam *software engineering*. Diawali dengan *system / information engineering* guna mengumpulkan informasi menggunakan metode wawancara dan observasi. Hasilnya digunakan dalam *analysis* dan *design* sistem untuk dirubah menjadi *unified modelling language* (UML). Proses *coding* menggunakan bahasa pemrograman *hyper-text preprocessor* (PHP) dan *database* MySQL, sementara *testing* sistem menggunakan metode *black-box*.

Kata kunci : Sistem Informasi Manajemen, salon, Qorinah Day Spa and Salon.

TAKARIR

<i>Activity diagram</i>	Diagram aktifitas
<i>Atomic</i>	Bersifat / bernilai tunggal
<i>Attribute</i>	Atribut
<i>Coding</i>	Proses pengkodean
<i>Conceptual design</i>	Desain konseptual
<i>Database</i>	Basis data
<i>Database life cycle</i>	Siklus hidup basis data
<i>Database management system</i>	Perangkat lunak yang berfungsi untuk manajemen basis data
<i>Entity</i>	Entitas
<i>Field</i>	Bagian basis data yang terisi data
<i>Hardware</i>	Perangkat keras
<i>Input</i>	masukan
<i>Interface</i>	antarmuka
<i>Linear sequential model</i>	Sebuah model pengembangan perangkat lunak yang dilakukan secara bertahap
<i>Logical design</i>	Desain logis
<i>Login</i>	Suatu proses yang digunakan untuk mengakses komputer dengan memasukkan identitas diri dan kata sandi
<i>Multivalued</i>	Bernilai ganda
<i>Output</i>	Keluaran
<i>Password</i>	Kata sandi
<i>Primary key</i>	Atribut kunci utama dari sebuah tabel dalam basis data
<i>Physical design</i>	Desain fisik
<i>Process</i>	Proses
<i>Software</i>	Perangkat lunak

Software Engineering

Rekayasa perangkat lunak

Testing

Pengujian

Use case diagram

Diagram penggunaan

User

Pengguna

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PENGESAHAN PEMBIMBING	ii
LEMBAR PENGESAHAN PENGUJI	iii
LEMBAR PERNYATAAN KEASLIAN	iv
HALAMAN PERSEMBAHAN	v
MOTTO	vi
KATA PENGANTAR.....	vii
ABSTRAK	ix
TAKARIR.....	x
DAFTAR ISI	xii
DAFTAR TABEL	xvi
DAFTAR GAMBAR	xviii

BAB I PENDAHULUAN

1.1 Latar Belakang	1
1.2 Rumusan Masalah	1
1.3 Batasan Masalah	2
1.4 Tujuan Penelitian.....	2
1.5 Manfaat Penelitian	3
1.6 Metodologi Penelitian	3
1.7 Sistematika Penelitian	3

BAB II LANDASAN TEORI

2.1 Definisi Sistem Informasi Manajemen	5
2.2 Rekayasa Perangkat Lunak	5
2.2.1. Definisi Rekayasa Perangkat Lunak	5
2.2.2. Software Process Model	6
2.2.2.1. The Linear Sequential Model	6
2.3 Database Development Lifecycle	8
2.3.1. Database Planning	8
2.3.2. Database Design	9

2.4	Requirement Collection and Analisis	9
2.4.1.	Wawancara	10
2.4.2.	Observasi	10
2.5	UML (Unified Modelling Language)	11
2.5.1.	Diagram UML	11
2.5.1.1.	Use Case Diagram	11
2.5.1.2.	Activity Diagram	12
2.6	Black Box Testing	13
2.7	Qorinah Day Spa and Salon	14

BAB III METODOLOGI

3.1	Studi Literatur dan Observasi	15
3.1.1.	Database Life Cycle	15
3.1.2.	Rencana Proyek	16
3.1.3.	Algoritma Proses Bisnis	16
3.2	Perencanaan Basisdata dan Analisis Kebutuhan Sistem	17
3.2.1.	Perencanaan Basisdata	17
3.2.1.1 .	Mission Statement	18
3.2.1.2.	Mission Objective	19
3.2.2.	Definisi Sistem	20
3.3	Analisis Kebutuhan Sistem	20
3.3.1.	Analisis Kebutuhan Basis Data	20
3.4	Desain Basisdata	22
3.4.1.	Conceptual Design	22
3.4.2.	Logical Design	25
3.4.2.1	Skema Logis	25
3.4.2.2	Relasi Tabel	30
3.4.3.	Normalisasi	30
3.4.3.1.	Bentuk Normal Pertama (1NF)	31
3.4.3.2.	Bentuk Normal Kedua (2NF)	31
3.4.3.3.	Bentuk Normal Ketiga (3NF)	32
3.4.4.	Physical Design	32
3.4.4.1.	Struktur Tabel dan SQL Treatment	32

3.5	Desain Aplikasi	36
3.5.1.	Use Case Diagram	36
3.5.2.	Activity Diagram	37
3.5.2.1.	Activity Diagram Login Sistem	37
3.5.2.2.	Activity Diagram Edit Profil	38
3.5.2.3.	Activity Diagram Ubah Password	39
3.5.2.4.	Activity Diagram Manajemen Pelanggan	40
3.5.2.5.	Activity Diagram Tambah Pelanggan	41
3.5.2.6.	Activity Diagram Manajemen Treatment dan Lihat Treatment.....	42
3.5.2.7.	Activity Diagram Tambah Treatment	43
3.5.2.8.	Activity Diagram Manajemen Karyawan dan Lihat Karyawan.....	44
3.5.2.9	Activity Diagram Tambah Karyawan	45
3.5.2.10.	Activity Diagram Presensi	46
3.5.2.11.	Activity Diagram Manajemen Order dan Tambah Order	46
3.5.2.12.	Activity Diagram Lihat Report	48
3.6	Desain Antarmuka Sistem	49
3.6.1.	Halaman Login	49
3.6.2.	Home Terapis	49
3.6.3.	Home Manajer	50
3.6.4.	Halaman Edit Profil	50
3.6.5.	Halaman Ubah Password	51
3.6.6.	Halaman Manajemen Pelanggan	52
3.6.7.	Halaman Tambah Pelanggan	53
3.6.8.	Halaman Manajemen Treatment	54
3.6.9.	Halaman Tambah Treatment	55
3.6.10.	Halaman Manajemen Karyawan	56
3.6.11.	Halaman Tambah Karyawan	57
3.6.12.	Halaman Manajemen Order	57
3.6.13.	Halaman Tambah Order	58
3.6.14.	Halaman Report	59

BAB IV HASIL & PEMBAHASAN

4.1	Implementasi dan Pengujian Sistem	61
4.1.1.	Halaman Login	61
4.1.2.	Halaman Edit Profil	64
4.1.3.	Halaman Ubah Password	66
4.1.4.	Halaman Manajemen Order	68
4.1.5.	Halaman Manajemen Pelanggan	72
4.1.6.	Halaman Manajemen Treatment	78
4.1.7.	Halaman Manajemen Karyawan	82
4.1.8.	Halaman Report.....	87
4.1.9.	Halaman Report Presensi	88
4.1.10.	Halaman Report Transaksi	89
4.1.11.	Halaman Report Treatment	90

BAB V KESIMPULAN DAN SARAN

5.1	Kesimpulan	91
5.2	Saran	91

DAFTAR PUSTAKA	92
-----------------------------	-----------

LAMPIRAN	93
-----------------------	-----------

DAFTAR TABEL

Tabel 3.1	Database Life Cycle	15
Tabel 3.2	Tabel <i>Mission Statement</i>	18
Tabel 3.3	Tabel <i>Mission Objective</i>	19
Tabel 3.4	Tabel Interaksi	21
Tabel 3.5	Detail Informasi.....	21
Tabel 3.6	Tabel Entitas	23
Tabel 3.7	Kardinalitas	23
Tabel 3.8	Atribut Entitas	23
Tabel 3.9	Tabel Entitas Pelanggan	25
Tabel 3.10	Tabel Entitas Karyawan	26
Tabel 3.11	Tabel Entitas Treatment	27
Tabel 3.12	Tabel Entitas Presensi	27
Tabel 3.13	Tabel Relasi Karyawan-Melakukan-Absensi	28
Tabel 3.14	Tabel Relasi Karyawan – Melayani – Pelanggan – Mengorder – <i>Treatment</i>	29
Tabel 3.15	Tabel Relasi <i>Treatment</i> – Memiliki – Jenis_nama.....	29
Tabel 3.16	Tabel Pelanggan	33
Tabel 3.17	Tabel Karyawan	33
Tabel 3.18	Tabel Treatment	34
Tabel 3.19	Tabel Presensi	35
Tabel 3.20	Tabel Order	36
Tabel 4.1	Tabel Pengujian Login	64
Tabel 4.2	Tabel Pengujian Edit Profil	66
Tabel 4.3	Tabel Pengujian Ubah Password	68
Tabel 4.4	Tabel Pengujian Tambah Order	70
Tabel 4.5	Tabel Pengujian Tambah Order Treatment	72
Tabel 4.6	Tabel Pengujian Tambah Pelanggan	75
Tabel 4.7	Tabel Pengujian Edit Pelanggan	77
Tabel 4.8	Tabel Pengujian Tambah Treatment	80

Tabel 4.9	Tabel Pengujian Edit Pelanggan	82
Tabel 4.10	Tabel Pengujian Tambah Karyawan	85
Tabel 4.11	Tabel Pengujian Edit Karyawan.....	87
Tabel 4.12	Tabel Pengujian Report	88
Tabel 4.13	Tabel Pengujian Report Presensi.....	89
Tabel 4.14	Tabel Pengujian Report Transaksi	90
Tabel 4.15	Tabel Pengujian Report Treatment	91

DAFTAR GAMBAR

Gambar 2.1	The Linear Sequential Model	7
Gambar 2.2	Database Development Lifecycle	8
Gambar 2.3	Use Case Diagram	12
Gambar 2.4	Activity Diagram	13
Gambar 3.1	Desain Relasi Tabel.....	30
Gambar 3.2	Activity Diagram Login	37
Gambar 3.3	Activity Diagram Edit Profil	38
Gambar 3.4	Activity Diagram Ubah Password	39
Gambar 3.5	Activity Diagram Manajemen Pelanggan	40
Gambar 3.6	Activity Diagram Tambah Pelanggan	41
Gambar 3.7	Activity Diagram Manajemen Treatment dan Lihat Treatment	42
Gambar 3.8	Activity Diagram Tambah Treatment	43
Gambar 3.9	Activity Diagram Manajemen Karyawan Dan Lihat Karyawan	44
Gambar 3.10	Activity Diagram Tambah Karyawan	45
Gambar 3.11	Activity Diagram Presensi.....	46
Gambar 3.12	Activity Diagram Manajemen Order dan Tambah Order	47
Gambar 3.13	Activity Diagram Lihat Report	48
Gambar 3.14	Login Site	49
Gambar 3.15	Home Terapis	50
Gambar 3.16	Home Manajer	50
Gambar 3.17	Halaman Edit Profil (Login Sebagai Manajer).....	51
Gambar 3.18	Halaman Edit Profil (Login Sebagai Terapis)	51
Gambar 3.19	Halaman Ubah Password (Login Sebagai Manajer)	52
Gambar 3.20	Halaman Ubah Password (Login Sebagai Terapis)	52
Gambar 3.21	Halaman Manajemen Pelanggan (Login Sebagai Manajer)...	53
Gambar 3.22	Halaman Lihat Pelanggan (Login Sebagai Terapis)	53
Gambar 3.23	Halaman Tambah Pelanggan (Login Sebagai Manajer)	54

Gambar 3.24	Halaman Tambah Pelanggan (Login Sebagai Terapis)	54
Gambar 3.25	Halaman Manajemen Treatment (Login Sebagai Manajer) ...	55
Gambar 3.26	Halaman Lihat Treatment (Login Sebagai Terapis).....	55
Gambar 3.27	Halaman Tambah Treatment	56
Gambar 3.28	Halaman Manajemen Karyawan (Login Sebagai Manajer) ..	56
Gambar 3.29	Halaman Lihat Karyawan (Login Sebagai Terapis).....	57
Gambar 3.30	Halaman Tambah Karyawan	57
Gambar 3.31	Halaman Manajemen Order (Login Sebagai Manajer)	58
Gambar 3.32	Halaman Manajemen Order (Login Sebagai Terapis).....	58
Gambar 3.33	Halaman Tambah Order (Login Sebagai Manajer).....	59
Gambar 3.34	Halaman Tambah Order (Login Sebagai Terapis)	59
Gambar 3.35	Halaman Report (Login Sebagai Manajer)	60
Gambar 3.36	Halaman Report (Login Sebagai Terapis).....	60
Gambar 4.1	Halaman Login	62
Gambar 4.2	Home Terapis	62
Gambar 4.3	Home Manajer	63
Gambar 4.4	Pesan Gagal Login	63
Gambar 4.5	Halaman Edit Profil (Login Sebagai Manajer)	64
Gambar 4.6	Halaman Edit Profil (Login Sebagai Terapis)	65
Gambar 4.7	Pesan Error Edit Profil	65
Gambar 4.8	Pesan Gagal Edit Profil	65
Gambar 4.9	Halaman Ubah Password (Login Sebagai Manajer)	67
Gambar 4.10	Halaman Ubah Password (Login Sebagai Terapis)	67
Gambar 4.11	Pesan Error Ubah Password	67
Gambar 4.12	Pesan Error Ubah Password	68
Gambar 4.13	Pesan Error Ubah Password	68
Gambar 4.14	Halaman Manajemen Order (Login Sebagai Manajer)	69
Gambar 4.15	Halaman Manajemen Order (Login Sebagai Terapis).....	69
Gambar 4.16	Halaman Tambah Order (Login Sebagai Manajer).....	70
Gambar 4.17	Halaman Tambah Order (Login Sebagai Terapis)	70
Gambar 4.18	Halaman Tambah Order Treatment(Login Sebagai Manajer)	71
Gambar 4.19	Halaman Tambah Order Treatment (Login Sebagai Terapis)	71

Gambar 4.20	Halaman Manajemen Pelanggan (Login Sebagai Manajer) .	72
Gambar 4.21	Halaman Lihat Pelanggan (Login Sebagai Terapis).....	73
Gambar 4.22	Halaman Tambah Pelanggan (Login Sebagai Manajer)	73
Gambar 4.23	Halaman Tambah Pelanggan (Login Sebagai Terapis)	74
Gambar 4.24	Pesan Berhasil Menyimpan Data	74
Gambar 4.25	Pesan Error Gagal Menyimpan Data	74
Gambar 4.26	Pesan Error Data Tidak Lengkap	75
Gambar 4.27	Halaman Edit Pelanggan (Login Sebagai Manajer)	76
Gambar 4.28	Halaman Edit Pelanggan (Login Sebagai Terapis)	76
Gambar 4.29	Pesan Berhasil Mengedit Data	77
Gambar 4.30	Pesan Error Gagal Menyimpan Data	77
Gambar 4.31	Pesan Error Data Tidak Lengkap	77
Gambar 4.32	Halaman Manajemen Treatment (Login Sebagai Manajer) ...	78
Gambar 4.33	Halaman Lihat Treatment (Login Sebagai Terapis).....	78
Gambar 4.34	Halaman Tambah Treatment (Login Sebagai Manajer).....	79
Gambar 4.35	Pesan Berhasil Menyimpan Data	79
Gambar 4.36	Pesan Error Gagal Menyimpan Data	80
Gambar 4.37	Pesan Error Data Tidak Lengkap	80
Gambar 4.38	Halaman Edit Treatment (Login Sebagai Manajer)	81
Gambar 4.39	Pesan Berhasil Mengedit Data	81
Gambar 4.40	Pesan Error Gagal Menyimpan Data	81
Gambar 4.41	Pesan Error Data Tidak Lengkap	82
Gambar 4.42	Halaman Manajemen Karyawan (Login Sebagai Manajer) ..	83
Gambar 4.43	Halaman Lihat Karyawan (Login Sebagai Terapis)	83
Gambar 4.44	Halaman Tambah Karyawan	84
Gambar 4.45	Pesan Berhasil Menyimpan Data	84
Gambar 4.46	Pesan Error Gagal Menyimpan Data	84
Gambar 4.47	Pesan Error Data Tidak Lengkap	85
Gambar 4.48	Halaman Edit Karyawan	86
Gambar 4.49	Pesan Berhasil Mengedit Data	86
Gambar 4.50	Pesan Error Gagal Menyimpan Data	86
Gambar 4.51	Pesan Error Data Tidak Lengkap	87

Gambar 4.52	Halaman Report (Login Sebagai Manajer)	88
Gambar 4.53	Halaman Report (Login Sebagai Terapis).....	88
Gambar 4.54	Halaman Report Presensi	89
Gambar 4.55	Halaman Report Transaksi	90
Gambar 4.56	Halaman Report Treatment	91

BAB I

PENDAHULUAN

1.1 Latar Belakang

Qorinah Day Spa and Salon adalah salon yang menawarkan berbagai macam produk dan perawatan yang menarik khusus untuk wanita. Saat sekarang ini Qorinah Day Spa and Salon masih menggunakan sistem manual atau tertulis untuk menangani order pelanggan, dan manajemen datanya. Penanganan secara manual bisa saja dilakukan tetapi banyak mengalami hambatan yaitu tidak efisien waktu dan tenaga, dan kemungkinan kehilangan data. Selain itu, Qorinah Day Spa and Salon memiliki pelanggan yang cukup banyak. Maka dari itu, teknologi informasi dapat dimanfaatkan untuk mempermudah semua pekerjaan tadi. Sistem informasi berfungsi untuk mendukung keputusan yang dilakukan oleh pihak perusahaan.

Oleh sebab itu, untuk mengatasi permasalahan tersebut, perlu adanya suatu sistem informasi yang terstruktur agar menghasilkan data / informasi yang bermanfaat secara tepat dan cepat. Sistem Informasi Manajemen Qorinah Day Spa and Salon dapat membantu dalam proses order. Sistem ini dibuat juga dengan tujuan mempermudah dalam proses pelaporannya. Dengan dibuatnya suatu sistem informasi manajemen salon ini dapat membantu dalam mengurangi kesalahan karena penggunaan sistem yang masih manual.

1.2 Rumusan Masalah

Pada penelitian ini, berikut adalah rumusan masalahnya :

1. Proses order, laporan, dan manajemen data masih dilakukan secara manual
2. Teknologi informasi belum dimanfaatkan pada Qorinah Day Spa and Salon

1.3 Batasan Masalah

Dalam pembuatan sistem informasi manajemen salon ini yang menjadi batasan masalah adalah sebagai berikut:

1. Studi kasus dilakukan di Qorinah Day Spa and Salon
2. Tidak termasuk perhitungan poin pelanggan
3. Tidak termasuk perhitungan gaji/ upah karyawan

1.4 Tujuan Penelitian

Tujuan dari penelitian ini adalah sebagai berikut :

1. Membuat sistem informasi yang dapat membantu dalam proses order dan manajemen data lainnya.
2. Mempermudah dalam pembuatan laporan yang mencakup order, transaksi, karyawan, presensi, dan treatment.

1.5 Manfaat Penelitian

Adapun manfaat penelitian ini adalah :

1. Bagi Penulis
Memenuhi syarat kelulusan S1 Teknik Informatika dan mengetahui proses bisnis order yang ada di Qorinah Day Spa and Salon.
2. Bagi Instansi
Mempermudah pengelolaan data yang dibutuhkan instansi, serta pembuatan laporannya.

1.6 Metodologi Penelitian

Metode penelitian adalah urutan langkah yang dilakukan dalam penelitian. Tugas akhir ini dilakukan dalam tahapan sebagai berikut:

1. Pengumpulan dan analisis data

Tahap ini berkaitan dengan proses pengumpulan data-data yang dibutuhkan untuk mendukung tahap analisis kebutuhan sistem. Proses pengumpulan data antara lain dengan cara interview, observasi, dan studi literatur.

2. Analisis kebutuhan sistem

Tahap ini berkaitan dengan proses analisa terhadap segala sesuatu yang dibutuhkan oleh sistem nantinya, seperti analisis kebutuhan *input*, analisis kebutuhan proses, analisis kebutuhan *output*, dan analisis kebutuhan *interface*.

3. Perancangan sistem

Tahap ini berkaitan dengan pembuatan rancangan awal dari sistem yang akan dibuat berdasarkan hasil dari analisis kebutuhan sistem. Perancangan yang dilakukan meliputi perancangan *databaseconceptual*, *logical* dan *physical*, serta perancangan aplikasi yang meliputi fungsi dan interface. Metode perancangan sistem menggunakan metode UML (*Unified Modeling Language*).

4. Pengkodean (*coding*)

Tahap selanjutnya adalah melakukan pengkodean (*coding*). Yang dimaksud disini adalah dilakukan proses pembuatan sistem berdasarkan hasil dari perancangan sistem dengan menggunakan *source code* dalam bahasa pemrograman PHP.

5. Pengujian sistem

Tahap ini berkaitan dengan pengujian terhadap sistem yang telah dibuat, untuk mengetahui apakah sistem sudah berjalan dengan baik serta sesuai dengan tujuan.

1.7 Sistematika Penulisan

Dalam penulisan laporan tugas akhir ini terdiri dari 5 bab dimana dalam sistematika penulisan memberikan gambaran secara umum dan menyeluruh dari laporan.

Bab I Pendahuluan, yang menjelaskan latar belakang tugas akhir yang dirancang, gambaran umum permasalahan beserta batasan masalah yang menjadi tolak ukur penulisan dalam melakukan penelitian, tujuan penelitian, manfaat penelitian, serta metodologi penelitian dan sistematika penulisan yang digunakan.

Bab II Landasan Teori, yang merupakan pembahasan tentang teori – teori yang digunakan dan relevan dengan topik tugas akhir, dimulai dari konsep sistem informasi manajemen yang bersifat umum sampai dengan teori tentang perangkat lunak yang digunakan untuk mendukung perancangan sistem informasi manajemen salon.

Bab III Metodologi, berisi analisis kebutuhan perangkat lunak, didalamnya memuat analisis kebutuhan sistem yang akan diterapkan dalam tugas akhir antara lain metode analisis, analisis kebutuhan dan hasil analisis.

Bab IV Hasil dan Pembahasan, membahas tentang implementasi perangkat lunak, pengujian, dan analisis perangkat lunak.

Bab V Kesimpulan dan Saran, berisi tentang beberapa kesimpulan dan saran bagi pengembangan sistem berdasarkan pengujian yang telah dilakukan.

BAB II

LANDASAN TEORI

2.1 Definisi Sistem Informasi Manajemen

Sistem informasi manajemen adalah kegiatan penyeleksian, pengelompokan, penyimpanan, pembaruan, dan pemanfaatan informasi dalam berbagai bentuk media penyimpanan (beragam bentuk dari lembar isian/ formulir, dokumen, serta data elektronik) (Rohim, 2002).

Sistem informasi manajemen adalah kumpulan dari sumber-sumber daya modal di dalam suatu organisasi yang bertanggung jawab mengumpulkan dan mengolah data untuk menghasilkan informasi yang berguna untuk semua tingkatan manajemen di dalam kegiatan perencanaan dan pengendalian.

Manfaat dari sistem informasi manajemen adalah:

1. Memberikan dukungan dalam pengumpulan informasi atau perancangan rangkaian alternatif tindakan, memutuskan untuk memilih tindakan yang terbaik dari alternatif yang tersedia.
2. Mendukung setiap tingkatan pada proses pengambilan keputusan dan dapat digunakan untuk memperoleh dan menyimpan informasi yang berkaitan dengan masalah, standar dan situasi sekarang.
3. Membantu untuk meralisasikan keputusan dalam tindakan dan mengawasi tindakan serta memberikan umpan balik yang berkaitan dengan hasilnya.

2.2 Rekayasa Perangkat Lunak

2.2.1 Definisi Rekaya Perangkat Lunak

Menurut Fritz Bauer (Pressman, 2001), *software engineering* merupakan pembuatan dan penggunaan prinsip *engineering* dalam mendapatkan software yang ekonomis, handal, dan efisien. Definisi Software Engineering menurut **IEEE (Institute of Electrical and Electronics Engineers)** adalah aplikasi sistematis, disiplin, pendekatan kuantitatif untuk pengembangan, operasi dan pemeliharaan

dari software; atau dapat disimpulkan sebagai teknik aplikasi untuk perangkat lunak. Intinya Software engineering berkaitan dengan pembangunan produk program.

2.2.2 Software Process Model

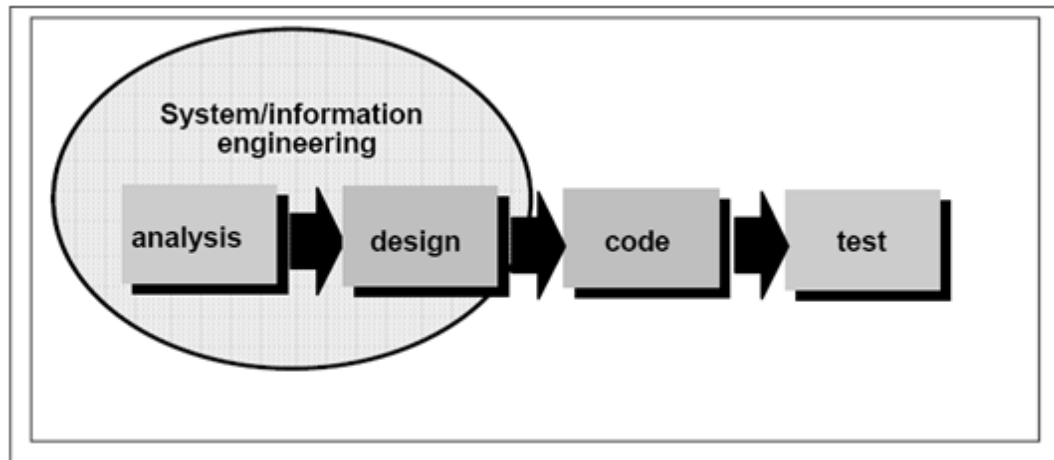
Sebuah *process model* untuk rekayasa perangkat lunak dipilih berdasarkan sifat proyek dan aplikasi, metode dan alat-alat yang akan digunakan, dan kontrol yang diperlukan (Pressman, 2001). Beberapa contoh software process model antara lain :

1. The Prototyping Model
2. The Rapid Application Development Model
3. The Incremental Model
4. The Spiral Model
5. The WINWIN Spiral Model
6. The Concurrent Development Model

Sistem yang akan dirancang menggunakan Linear Sequential Model, karena model ini menggunakan SDM yang sedikit dalam pengerjaannya, dan untuk waktu yang tidak terlalu panjang, serta tidak perlu meminta saran kepada instansi yang bersangkutan selama pengerjaan berlangsung.

2.2.2.1 The Linear Sequential Model (Waterfall Model)

Biasanya disebut juga *classic life cycle* atau model air terjun (waterfall). Menunjukkan pendekatan yang sistematis dan terurut untuk pengembangan software yang dimulai pada tingkat sistem dan kemajuan melalui analisis, desain, *coding*, dan *testing* (Pressman, 2001). Desain tersebut dapat dilihat pada Gambar 2.1.



Gambar 2.1 *The Linear Sequential Model*

Berikut adalah siklus tahap / kegiatan dalam model tersebut menurut Pressman (2001) :

1. *Analysis*

Tahap ini bisa disebut dengan analisa kebutuhan perangkat lunak. Dalam tahap ini, proses pengumpulan persyaratan diintensifkan dan difokuskan secara khusus pada perangkat lunak. Segala persyaratan tersebut didokumentasikan dan dikaji. Tujuannya adalah untuk memahami sifat program yang akan dibuat meliputi fungsi yang diperlukan, perilaku, kinerja, dan antarmuka.

2. *Design*

Tahap ini terfokus pada empat proses, yaitu struktur data, arsitektur perangkat lunak, detail prosedur / algoritma, dan representasi *interface*. Hasil yang diperoleh dari tahap ini, akan merepresentasikan kebutuhan dari perangkat lunak yang akan dibuat. Tujuannya adalah untuk menilai kualitas perangkat lunak sebelum proses coding dimulai.

3. *Coding*

Tahap ini menerjemahkan desain perangkat lunak yang dibuat pada tahap sebelumnya, ke dalam bentuk yang dapat dimengerti oleh komputer.

4. Testing

Setelah tahap *coding* selesai dilakukan, selanjutnya adalah tahap *testing*. Tahap ini untuk mengetahui kesalahan yang mungkin ada pada program yang telah selesai dibuat. Hal tersebut untuk memastikan apakah fungsi *input* berjalan dengan baik, sehingga menghasilkan *output* yang sesuai.

2.3 Database Development Lifecycle

Menurut Connolly & Begg (2005), *database development lifecycle* adalah komponen fundamental di dalam sistem informasi yang berguna untuk mendesain basis data dari tingkat awal sampai akhir. Berikut adalah aktivitas utama dari database development lifecycle dapat dilihat pada Gambar 2.2:

Stage	Main activities
<i>Database planning</i>	Planning how the stages of the lifecycle can be realized most efficiently and effectively.
<i>System definition</i>	Specifying the scope and boundaries of the database system, including the major user views, its users, and application areas.
<i>Requirements collection and analysis</i>	Collection and analysis of the requirements for the new database system.
<i>Database design</i>	Conceptual, logical, and physical design of the database.
<i>DBMS selection (optional)</i>	Selecting a suitable DBMS for the database system.
<i>Application design</i>	Designing the user interface and the application programs that use and process the database.
<i>Prototyping (optional)</i>	Building a working model of the database system, which allows the designers or users to visualize and evaluate how the final system will look and function.
<i>Implementation</i>	Creating the physical database definitions and the application programs.
<i>Data conversion and loading</i>	Loading data from the old system to the new system and, where possible, converting any existing applications to run on the new database.
<i>Testing</i>	Database system is tested for errors and validated against the requirements specified by the users.
<i>Operational maintenance</i>	Database system is fully implemented. The system is continuously monitored and maintained. When necessary, new requirements are incorporated into the database system through the preceding stages of the lifecycle.

Gambar 2.2 *Database Development Lifecycle*

2.3.1 Database Planning

Menurut Connolly and Begg, *database planning* adalah aktivitas manajemen yang memungkinkan bagian pengembangan siklus hidup basis data untuk direalisasikan secara efektif dan efisien. Hal penting pertama dalam *database planning* adalah mendefinisikan *mission statement* secara jelas untuk sistem basis data. *Mission statement* membantu untuk mengklarifikasi tujuan dari

sistem basis data dan memberikan jalan yang lebih jelas terhadap efisiensi dan efektifitas sistem basis data yang diperlukan.

Setelah *mission statement* ditetapkan, langkah selanjutnya adalah mengidentifikasi *mission objective*. *Mission objective* harus mendefinisikan tugas tertentu yang mendukung basis data. Kedua langkah tersebut bisa dilakukan dengan metode observasi dan wawancara untuk mengumpulkan informasi terkait.

2.3.2 Database Design

Menurut Connolly and Begg (2004), *database design* merupakan proses membentuk sebuah desain yang bisa membantu *mission objective* dan *mission statement* perusahaan untuk kebutuhan sistem basis data.

Database design terbagi atas 3, yaitu :

a. *Conceptual Design*

Conceptual design atau desain konseptual adalah proses membangun sebuah model data yang berasal dari informasi dan data organisasi yang bersifat independen dari seluruh aspek fisik.

b. *Logical Database Design*

Logical database design atau desain logika basis data adalah proses membangun sebuah model data yang digunakan organisasi dengan data model yang spesifik, tetapi independen dari bagian sistem manajemen basis data dan pertimbangan fisik lainnya.

c. *Physical Database Design*

Physical database design menjelaskan relasi dasar, data organisasi, dan indeks yang digunakan untuk mencapai akses data yang efisien. *Physical database design* merupakan tahap akhir dalam proses desain basis data yang ditetapkan oleh desainer dan akan diimplementasikan.

2.4 Requirement Collection and Analysis

Requirement Collection and Analysis adalah proses mengumpulkan dan menganalisis informasi yang menjadi bagian dari suatu organisasi yang didukung

oleh sistem database, dan menggunakan informasi untuk mengidentifikasi persyaratan untuk suatu sistem yang baru (Connolly and Begg, 2005).

Teknik yang digunakan untuk mengumpulkan informasi adalah dengan observasi dan wawancara.

2.4.1 Wawancara

Wawancara merupakan interaksi yang melibatkan paling sedikitnya dua orang, yaitu pewawancara dan narasumber. Wawancara merupakan salah satu teknik pengumpulan informasi perusahaan terkait. Wawancara yang dilakukan pada penelitian ini ditujukan pada beberapa subyek yang terkait pada Qorinah Day Spa and Salon.

Menurut Connolly & Begg (2005), penggunaan wawancara sebagai cara memperoleh informasi memberi beberapa keuntungan, antara lain:

1. Memungkinkan narasumber untuk merespon bebas dan terbuka terhadap pertanyaan.
2. Memungkinkan narasumber untuk merasa menjadi bagian dari proyek.
3. Memungkinkan pewawancara untuk menindaklanjuti komentar menarik yang dibuat oleh orang yang diwawancarai.
4. Memungkinkan pewawancara untuk beradaptasi atau merubah pertanyaan selama wawancara.
5. Memungkinkan pewawancara untuk mengamati bahasa tubuh yang diwawancarai

Hasil dari wawancara yang dilakukan berupa masalah dan kebutuhan *mission objective*, *mission statement*, dan kebutuhan database.

2.4.2 Observasi

Observasi dilakukan dengan cara mengamati proses bisnis yang terjadi di perusahaan. Observasi bertujuan untuk mengumpulkan informasi dan fakta perusahaan. Metode ini dapat digunakan untuk melengkapi informasi yang belum lengkap yang telah dilakukan dengan metode lain.

Menurut Connolly & Begg (2005), metode observasi memiliki beberapa keuntungan, diantaranya :

1. Memungkinkan pengecekan validitas fakta dan data.
2. Peneliti dapat melihat apa yang sedang dilakukan.
3. Peneliti juga dapat memperoleh data yang menggambarkan lingkungan fisik dari sistem.
4. Relatif murah
5. Peneliti dapat melakukan pengukuran kinerja

Proses observasi yang dilakukan memberikan hasil berupa algoritma order treatment yang dilakukan oleh pelanggan.

2.5 UML (*Unified Modelling Language*)

UML adalah keluarga notasi grafis yang didukung oleh meta-model tunggal, yang membantu pendeskripsian dan desain sistem perangkat lunak, khususnya sistem yang dibangun menggunakan pemrograman berorientasi objek (Fowler, 2003). UML dibagi menjadi 13 bagian yang bisa digunakan untuk *software modelling* (Pressman, 2001).

Dalam UML dibutuhkan sebuah konsep model yang dibangun menggunakan blokdasar UML, seperti *class*, *interface*, kolaborasi, komponen, *node*, dependensi, generalisasi, dan asosiasi (Fowler, 2003). Sistem informasi manajemen ini tidak memasukkan tipe diagram perilaku yang lain karena sudah dapat mewakili proses bisnis, keterbatasan ruang pengujian, dan lingkup pembahasan yang tidak mencakup algoritma pemrograman.

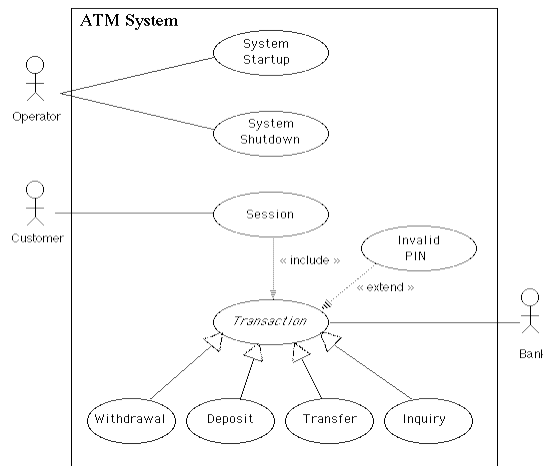
2.5.1 Diagram UML

2.5.1.1 Use Case Diagram

Use case adalah sebuah teknik untuk menangkap kebutuhan fungsional dari sistem (Fowler, 2003). *Use case diagram* adalah pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Penelitian ini menggunakan *use case diagram* untuk menggambarkan interaksi

yang khas antara pengguna sistem dengan sistem itu sendiri, memberikan narasi tentang bagaimana sistem yang digunakan. *Use case* bekerja menggunakan skenario interaksi antara *user* dengan sistem.

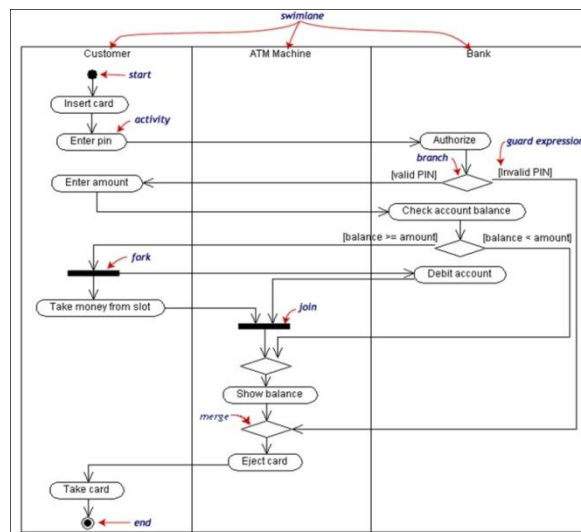
Usecase dapat dilihat pada Gambar 2.3 berikut ini :



Gambar 2.3 *Use Case Diagram*

2.5.1.2 Activity Diagram

Activity diagram adalah teknik untuk menggambarkan logika prosedural, proses bisnis, dan alur kerja (Fowler, 2003). *Activity* diagram biasanya merupakan permintaan dari operasi, langkah sebuah proses bisnis, atau proses bisnis secara keseluruhan. *Activity* diagram menunjukkan alur kontrol dan cabang yang bisa digunakan (Connolly and Begg, 2005). *Activity* diagram berfungsi untuk memungkinkan siapapun untuk memilih proses atau urutan mana yang mereka ingin ikuti. Hal ini penting untuk pemodelan bisnis, karena proses sering terjadi secara paralel.



Gambar 2.4 Activity Diagram

2.6 Black Box Testing

Black box testing merupakan salah satu teknik pengujian perangkat lunak yang berfokus pada faktor fungsionalitas dan spesifikasi perangkat lunak (Pressman, 2001). Pengujian ini umumnya digunakan untuk mengetahui kualitas dari hasil akhir pembuatan sebuah perangkat lunak. Hasil tersebut berupa kebenaran dari *output* yang dihasilkan oleh sistem, berdasarkan data atau kondisi dari *input* yang diberikan. Selain itu, menurut Pressman (2001) pengujian dilakukan untuk menjawab pertanyaan berikut :

1. Bagaimana kebenaran fungsional diuji?
2. Bagaimana capaian dan perilaku sistem diuji?
3. Kelas masukan manakah yang akan berhasil dalam pengujian kasus?
4. Apakah sistem sensitif terhadap nilai-nilai masukan tertentu?
5. Bagaimana batasan dari suatu data diisolasi?
6. Berapapasio data dan jumlah data yang dapat ditoleransi oleh sistem?
7. Apa akibat yang akan timbul dari kombinasi data spesifik pada operasi sistem?

Teknik pengujian perangkat lunak dengan menggunakan *black-box testing* terdiri dari beberapa metode pengujian. Salah satu metode tersebut kemudian diimplementasikan pada penelitian ini. Metode tersebut disebut dengan metode *equivalence partitioning*.

2.7 Qorinah Day Spa and Salon

Salon adalah sebuah usaha yang menyediakan layanan umum yang berkaitan dengan kesehatan kulit, wajah, perawatan kaki, meditasi, terapi, dan lain sebagainya. Kebanyakan salon masih memanajemen salonnya dengan manual, tidak menggunakan sistem terkomputerisasi. Saat sekarang ini, banyak salon yang dikhususkan untuk wanita.

Qorinah Day Spa and Salon salah satunya. Qorinah Day Spa and Salon berlokasi di Jl. Kaliurang Km 15 Ruko Degolan No. 18-19 Yogyakarta. Salon ini memiliki motto “Sehat – Bugar – Menarik”. Salon ini memiliki terapis yang telah dilatih terlebih dahulu. Qorinah Day Spa and Salon memiliki standar kualitas pelayanan tinggi yang mengutamakan kepuasan pelanggan.

Salon ini menawarkan beberapa macam produk untuk perawat rambut sampai dengan kaki, diantaranya Spa Treatment Package, Ala Carte Body Treatment, Facial Treatment, Hand and Foot Treatment, Hair Treatment Package, Hair Treatment, Waxing, dan Prewedding Package. Harga yang ditawarkan salon ini berkisar dari Rp. 5000, 00 – Rp. 700.000, 00.

Pada saat sekarang ini, salon ini masih menggunakan sistem manual dalam segala pengelolaannya. Hal tersebut sangat tidak efektif dan efisien, seharusnya sudah menggunakan sistem secara terkomputerisasi untuk mengatasi banyaknya data yang ada, bisa terjadi kemungkinan adanya duplikasi data, dan data yang tidak masuk dalam laporan.

BAB III

METODOLOGI

3.1 Studi Literatur dan Observasi Instansi

3.1.1. *Database Life Cycle*

Tujuan utama dari sub-bab ini adalah untuk mendefinisikan secara rinci metode yang digunakan sebagai acuan dalam membangun aplikasi. Tabel 3.1 menghadirkan tahapan yang ada dalam siklus hidup basisdata yang diambil dari Connolly and Begg (2003). Berikut adalah uraian dari masing-masing tahap:

Tabel 3.1 *Database Life Cycle*

Langkah	Aktifitas
<i>Database Planning</i>	Merencanakan bagaimana tahapan <i>lifecycle</i> dapat direalisasikan secara efisien dan efektif.
<i>System Definition</i>	Menentukan ruang lingkup dan batas-batas aplikasi <i>database</i> , penggunaanya, dan area aplikasi.
<i>Requirements Collection and Analysis</i>	Mengumpulkan dan menganalisis kebutuhan pengguna dan area aplikasi.
<i>Database Design</i>	Merancang konsep, desain logis dan fisik dari <i>database</i> .
<i>Application Design</i>	Merancang grafis antarmuka pengguna dan program aplikasi yang menggunakan dan memproses <i>database</i> .
<i>Prototyping</i>	Membangun model kerja dari aplikasi <i>database</i> , yang memungkinkan pengguna untuk memvisualisasikan dan mengevaluasi bagaimana sistem final akan berfungsi.
<i>Implementation</i>	Menciptakan definisi <i>database</i> eksternal, konseptual, dan internal serta program aplikasi.
<i>Data conversion and loading</i>	Mengambil data dari sistem lama ke sistem baru
<i>Testing</i>	Menguji kekurangan aplikasi <i>database</i> dan divalidasi terhadap persyaratan yang ditentukan oleh pengguna.

3.1.2 Rencana Proyek

Rencana proyek ini dikerjakan dalam waktu 9bulan. berikut penjelasan setiap tahapan-tahapan yang dikerjakan berdasarkan waktu yang telah ditentukan.

- Bulan 1&2 : Observasi perusahaan
Pengumpulan kebutuhan basisdata
- Bulan3& 4 : Desain basisdata
Evaluasi desain basisdata
- Bulan5, 6 &7 : Implementasi basisdata
Analisis kebutuhan dan spesifikasi aplikasi
Desain aplikasi
- Bulan8&9 : Implentasi desain aplikasi
Melakukan uji coba perangkat lunak

Bulan pertama dan kedua fokus pada perusahaan dan mengumpulkan informasi-informasi yang diperlukan untuk membangun aplikasi. Dalam bulan awal dilakukan observasi proses bisnis perusahaan yang kemudian dilanjutkan dengan wawancara untuk mendapatkan informasi yang lebih mendalam atau menguji validitas hasil observasi.

Bulanke 3 digunakan untuk merancang basisdata dan hasil dari bulan ke 4 digunakan untuk evaluasi rancangan basisdata. Jika ada perubahan, akan dilakukan pada rancangan basisdata tersebut.

Implementasi basisdata dimulai pada bulan ke 5 dan 6 bersamaan dengan analisis kebutuhan dan spesifikasi aplikasi, lalu bulan ke 7 dan 8 dilanjutkan dengan desain aplikasi dan evaluasi. Bulan 9 melaksanakan implementasi aplikasi dan uji coba perangkat lunak.

3.1.3 Algoritma Proses Bisnis

Qorinah Day Spa and Salon merupakan salon kecantikan khusus wanita yang menyediakan layanan *treatment* bagi pelanggannya. Perusahaan ini masih menggunakan sistem manual. Mereka menyadari bahwa mereka memiliki pekerjaan yang harus dilakukan dalam rangka untuk menjaga agar perusahaan berkembang. Oleh karena itu dibutuhkannya sistem yang mendukung berbagai

macam pekerjaan mereka tersebut, antara lain adalah yang berhubungan dengan manajemen karyawan, pelanggan, laporan, dan order.

Proses order, dilakukan pelanggan dengan menghubungi terlebih dahulu atau langsung datang ke Qorinah Day Spa and Salon. Alur order dibawah ini didapat dengan wawancara. Verbatim hasil dari wawancara dilihat pada Lampiran

1. Berikut adalah proses saat mengorder :

1. Pelanggan menghubungi atau datang langsung ke Qorinah Day Spa and Salon.
2. Pelanggan menanyakan ketersediaan jam yang ada di Qorinah Day Spa and Salon.
3. Karyawan memberikan informasi ketersediaan jam kepada pelanggan.
4. Diasumsikan pelanggan setuju dengan ketersediaan jam lalu karyawan mencatat identitas pelanggan berupa nama pelanggan dan nama *treatment* yang akan diorder.
5. Terapis memasukkan order berdasarkan nama pelanggan dan memasukkan *treatment* yang akan diorder oleh pelanggan.
6. Pelanggan melakukan *treatment* yang sudah diorder.
7. Pelanggan yang sudah selesai melakukan *treatment*, kemudian melakukan proses pembayaran.

3.2 Perencanaan Basisdata dan Analisis Kebutuhan Sistem

3.2.1. Perencanaan Basisdata

Selama proses ini berlangsung digunakan beberapa metode untuk pengumpulan informasi. Metode pertama adalah observasi. Observasi dilakukan untuk mengamati, melihat dan menonton proses bisnis yang berjalan. Misalnya adalah mengamati seorang terapis dalam menangani order dari pelanggan dan data apa saja yang diperlukan untuk menjalankan proses tersebut.

Metode kedua adalah wawancara. Metode ini merupakan yang paling umum digunakan untuk mencari informasi yang dibutuhkan. Wawancara dilakukan

dengan mengajukan pertanyaan kepada orang-orang yang dianggap berpotensi sebagai pengguna sistem.

Hasil dari tahap ini akan menjadi pernyataan misi dan tujuan misi proyek. Pernyataan misi mendefinisikan tujuan utama pengembangan aplikasi *database*. Tahap selanjutnya adalah mengidentifikasi tujuan misi, yang akan mengidentifikasi tugas-tugas tertentu yang membutuhkan dukungan *database*.

3.2.1.1. Mission Statement

Mission statement didapat melalui observasi dan mewawancara manager sekaligus *owner* dari Qorinah Day Spa and Salon. Verbatim hasil wawancara dilampirkan pada Lampiran 1. Hasil wawancara adalah sebagai berikut :

Tabel 3.2 Tabel *Mission Statement*

Tujuan Umum	Untuk memudahkan proses bisnis order <i>treatment</i> pelanggan
Keinginan Qorinah Day Spa and Salon akan basisdata	1. Perusahaan menginginkan basisdata yang dapat menyimpan data order pelanggan. 2. Basisdata dapat menyimpan data karyawan, pelanggan, dan <i>treatment</i>. 3. Perusahaan memperoleh sistem yang dapat menyediakan laporanpresensi, transaksi, <i>treatment</i>, jam dan hari kerja.

Tujuan misi dari proyek ini adalah membangun sistem informasi yang dapat membantu manajemen data order, pelanggan, *treatment*, dan karyawan untuk memudahkan proses bisnis order dan pembuatan laporannya.

3.2.1.2. Mission Objective

Tahap selanjutnya adalah menentukan tujuan proyek. Tujuan proyek didapat dengan wawancara ke beberapa orang yang dianggap berperan untuk memberikan informasi yang dibutuhkan. Verbatim hasil dari wawancara dapat dilihat pada Lampiran1. Hasil wawancara dapat dilihat pada tabel dibawah ini :

Tabel 3.3 Tabel *Mission Objective*

Interviewee	Subject	Summary
Owner / Manager	Peran di instansi	Memiliki hak penuh atas perusahaan, mengawasi kinerja karyawan, dan berperan ganda sebagai terapis.
	Data yang dibutuhkan	1. Data pelanggan 2. Data <i>treatment</i> 3. Data order 4. Data karyawan 5. Data presensi
	Tugas harian	1. Memantau kinerja karyawan 2. Memastikan semua proses berjalan lancar 3. Menerima laporan yang dibuat oleh karyawan 4. Melakukan presensi
	Laporan yang diterima	1. Laporan transaksi 2. Laporan presensi, jam dan hari kerja 3. Laporan <i>treatment</i>
Terapis	Peran di Instansi	Melayani pelanggan dan membuat laporan
	Data yang dibutuhkan	1. Data pelanggan 2. Data order 3. Data presensi
	Tugas harian	1. Mengumpulkan data pelanggan 2. Melayani order pelanggan 3. Merekap dan mengelola data pelanggan 4. Membuat laporan transaksi 5. Melakukan presensi
	Laporan yang dibuat	1. Laporan transaksi 2. Laporan <i>treatment</i>

Dengan tabel diatas, maka akan menghasilkan tujuan misi proyek yaitu:

Untuk mengelola (*input, update, delete*) data pelanggan

Untuk mengelola (*input, update, delete*) data *treatment*

Untuk mengelola (*input, update, delete*) data order

Untuk mengelola (*input, update, delete*) data karyawan

Untuk mengelola (*input*) presensi

Untuk melaporkan rekapitulasi transaksi

Untuk melaporkan rekapitulasi *treatment*

Untuk melaporkan rekapitulasi presensi, jam dan hari kerja karyawan

3.2.2. Definisi Sistem

Seperti yang telah dijelaskan sebelumnya, pada bagian ini akan dibahas mengenai ruang lingkup aplikasi *database* dan pandangan pengguna utama. Mengenai aplikasi *database* yang digunakan, tugasnya menyangkut bagaimana menangani data order pelanggan.

Selanjutnya, *user* yang paling potensial dalam penggunaan sistem adalah manajer dan terapis. Terapis lebih banyak berperan dalam masalah order pelanggan, namun manajer juga bisa merangkap sebagai terapis apabila diperlukan. Pernyataan di atas merujuk pada dokumen Verbatim yang dilampirkan pada Lampiran 1.

3.3 Analisis Kebutuhan Sistem

3.3.1. Analisis Kebutuhan Basisdata

Observasi dan wawancara adalah teknik yang digunakan pada tahap ini. Dokumen yang dikumpulkan digunakan sebagai dasar informasi mengenai data yang digunakan oleh karyawan selama bekerja. Dokumen tersebut adalah *treatment* dan pelanggan.

Metode lain yang digunakan dalam tahap ini adalah wawancara. Tabulasi hasil wawancara dapat dilihat pada tabel 3.4, sedangkan verbatim dari proses wawancara tersedia dalam Lampiran 1.

Tabel 3.4 Tabel Interaksi

Data	Jenis Akses	Manager / Owner	Terapis
Data Pelanggan	Kelola	X	X
	Query	X	X
	Laporan	X	X
Data <i>Treatment</i>	Kelola	X	X
	Query	X	X
	Laporan	X	
Data Karyawan	Kelola	X	
	Query	X	X
	Laporan	X	
Data Presensi	Kelola	X	
	Query	X	X
	Laporan	X	X
Data Order	Kelola	X	X
	Query	X	X
	Laporan	X	X

Tabel di atas menunjukkan bagaimana interaksi pengguna yang berperan dengan sistem. Berdasarkan tabel tersebut, manager atau terapis adalah orang-orang yang sering berinteraksi dengan sistem. Jadi desain sistem *database* akan fokus pada bagaimana menyediakan aplikasi yang mendukung mereka.

Daftar informasi dibawah ini merujuk pada *logical design* dalam bentuk *Entity Relationship Diagram (ERD)* yang dapat dilihat pada Lampiran 2. Lebih lanjut, Tabel 3.5 memberikan daftar informasi dalam sebuah proses order dan presensi.

Tabel 3.5 Detail Informasi

Data	Data yang Dibutuhkan
Data Pelanggan	Nama pelanggan
	Alamat pelanggan
	Tanggal lahir

Data	Data yang Dibutuhkan
	Nomor telepon
	Email
	Status
Data Treatment	Nama <i>treatment</i>
	Jenis <i>treatment</i>
	Durasi <i>treatment</i>
	Harga <i>treatment</i>
	Keterangan
Data Karyawan	Nama karyawan
	Alamat karyawan
	Tanggal lahir karyawan
	Nomor telepon karyawan
	Jabatan karyawan
	Username
	Password
Data Presensi	Foto karyawan
	Jam checkin
	Jam checkout
Data Order	Tanggal presensi
	Tanggal order
	Status order
	Jumlah order
	Total harga

3.4 Desain Basisdata

Desain basisdata terbagi menjadi tiga tahapan yaitu tahap *conceptual design*, *logical design*, dan *physical design*. Tahap *conceptual design* menentukan entitas penting, relasi dan atribut yang nantinya akan digunakan dalam tahap *logical design*. Tahap *logical design* kemudian membangun relasi logis antar entitas. Tahap ketiga *physical design*, menghasilkan desain akhir yang nantinya akan digunakan dalam implementasi basisdata.

3.4.1 Conceptual Design

Langkah awal dalam desain basisdata adalah menentukan entitas (Connolly and Begg, 2003). Seperti yang telah dijelaskan sebelumnya, terdapat lima data yang akan dikelola yaitu: data karyawan, data pelanggan, data *treatment*, data order, dan data presensi. Data order merupakan relasi antara data pelanggan dan

treatment. Selain data order, semua data tersebut kemudian dirubah menjadi entitas.

Tabel 3.6 Tabel Entitas

Nama Entitas	Deskripsi
Karyawan	Karyawan yang melayani pelanggan dan memberikantreatment kepada pelanggan
Pelanggan	Seseorang yang menggunakan jasa dari Qorinah Day Spa and Salon
Treatment	Jenis jasa yang diberikan oleh Qorinah Salon and Spa kepada pelanggan
Presensi	Tanda hadir dan pulang dari karyawan

Setelah mengidentifikasi entitas dari database, maka langkah selanjutnya adalah menentukan relasi di antara entitas tersebut. Relasi yang ada di antara entitas tersebut adalah :

- Karyawan melayani pelanggan mengorder *treatment*
- Karyawan melakukan presensi

Hubungan antar entitas tersebut memiliki aturan kardinalitas yang berbeda satu sama lain, dijelaskan pada tabel 3.7

Tabel 3.7 Kardinalitas

Hubungan Entitas	Kardinalitas	Deskripsi
Karyawan melayani pelanggan mengorder <i>treatment</i>	Many – to – Many	Satu karyawan bisa melayani beberapa pelanggan. Satu pelanggan bisa dilayani oleh beberapa karyawan.
		Satu pelanggan bisa mengorder beberapa <i>treatment</i> . Setiap satu <i>treatment</i> bisa diorder oleh beberapa pelanggan
Hubungan Entitas	Kardinalitas	Deskripsi
Karyawan melakukan presensi	One – to – Many	Satu karyawan boleh melakukan banyakpresensi. Satu presensi dilakukan oleh satu karyawan.

Tahap terakhir dalam *conceptual design* adalah mengidentifikasi dan mengasosiasikan atribut dengan entitas. Kelima entitas yang sudah dijelaskan di atas, masing-masing memiliki atribut. Atribut dari entitas akan diperjelas selanjutnya yaitu pada *logical design* dengan menggunakan *Entity Relationship Diagram (ERD)*. *ERD* dapat dilihat pada Lampiran 2. Ringkasan tabulasi atribut dari masing-masing entitas akan dijelaskan pada tabel 3.8

Tabel 3.8 Atribut Entitas

Nama Entitas	Atribut
Karyawan	Id Nama Tanggal lahir Alamat Nomor telepon Foto Jabatan Username Password
Pelanggan	Id Nama Alamat Tanggal lahir Nomor telepon Email Status
<i>Treatment</i>	Id Nama Harga Durasi Keterangan <i>Jenis_treatment</i>
Presensi	Id Karyawan Tanggal Jam check-in Jam check-out

Berdasarkan tabel diatas, sebagian besar atribut adalah *single-attribute*. Namun untuk *treatment* memiliki *multivalued* atribut, yaitu *jenis_treatment*. Hal tersebut karena sistem harus memberikan kemudahan kepada pengguna untuk memasukkan nilai dari atribut tersebut, misalnya, dengan menyediakan beberapa opsi sehingga pengguna dapat mudah memilih salah satu nilai.

3.4.2 Logical Design

Pada tahap ini proses desain *database* akan fokus pada bagaimana membangun skema logis yang benar dan efisien. Tujuannya adalah agar dapat mewakili semua informasi yang dijabarkan pada skema konseptual. Hasil dari skema logis dapat dilihat pada Lampiran 3.

3.4.2.1 Skema Logis

Langkah pertama dalam tahap ini adalah dengan mengidentifikasi entitas-entitas beserta atributnya. Berikutnya adalah mengidentifikasi relasi entitas *many to many* dan *one to many*.

Langkah 1 : Identifikasi entitas

Berdasarkan apa yang dijelaskan dari tahap desain konseptual *database*, maka disimpulkan terdapat lima entitas utama, yaitu : pelanggan, karyawan, *treatment*, order, dan presensi. Semuanya akan digambarkan dalam satu langkah, termasuk atribut didalamnya.

Entitas ‘pelanggan’

Entitas pelanggan adalah entitas yang memuat data mengenai pelanggan, yang terdiri dari beberapa atribut, yaitu pelanggan_id, pelanggan_nama, pelanggan_alamat, pelanggan_tgl_lahir, pelanggan_email, pelanggan_no_telepon, pelanggan_status. Data entitas pelanggan ditunjukkan pada Tabel 3.9.

Tabel 3.9 Tabel Entitas Pelanggan

Entitas	Atribut	Keterangan
Pelanggan	Pelanggan_id	Primary Key
	Pelanggan_nama	-
Entitas	Atribut	Keterangan
Pelanggan	Pelanggan_alamat	-
	Pelanggan_tgl_lahir	-
	Pelanggan_email	-
	Pelanggan_no_telepon	-
	Pelanggan_status	-

Entitas ‘karyawan’

Entitas karyawan adalah entitas yang memuat data mengenai karyawan, yang terdiri dari beberapa atribut, yaitu *karyawan_id*, *karyawan_nama*, *karyawan_alamat*, *karyawan_tgl_lahir*, *pelanggan_no_telepon*, *karyawan_jabatan*, *karyawan_username*, *karyawan_password*, *karyawan_foto*. Data entitas karyawan ditunjukkan pada Tabel 3.10.

Tabel 3.10 Tabel Entitas Karyawan

Entitas	Atribut	Keterangan
Karyawan	<i>Karyawan_id</i>	Primary Key
	<i>Karyawan_nama</i>	-
	<i>Karyawan_alamat</i>	-
	<i>Karyawan_tgl_lahir</i>	-
	<i>Karyawan_no_telepon</i>	-
	<i>Karyawan_jabatan</i>	-
	<i>Karyawan_username</i>	-
	<i>Karyawan_password</i>	-
	<i>Karyawan_foto</i>	-

Entitas ‘treatment’

Entitas *treatment* adalah entitas yang memuat data mengenai *treatment*, yang terdiri dari beberapa atribut, yaitu *treatment_id*, *treatment_nama*, *treatment_durasi*, *treatment_harga*, *treatment_keterangan*. Data entitas *treatment* ditunjukkan pada Tabel 3.11.

Tabel 3.11 Tabel Entitas *Treatment*

Entitas	Atribut	Keterangan
<i>Treatment</i>	<i>Treatment_nama</i>	
	<i>Treatment_durasi</i>	
	<i>Treatment_harga</i>	
	<i>Treatment_keterangan</i>	
	<i>Jenis_nama</i>	

Entitas ‘presensi’

Entitas presensi adalah entitas yang memuat data mengenai presensi, yang terdiri dari beberapa atribut, yaitu karyawan_id, presensi_tanggal, presensi_jam_checkin, presensi_jam_checkout. Data entitas presensi ditunjukkan pada Tabel 3.12.

Tabel 3.12 Tabel Entitas Presensi

Entitas	Atribut	Keterangan
Presensi	Karyawan_id	Foreign Key
	Presensi_tanggal	-
	Presensi_jam_checkin	-
	Presensi_jam_checkout	-

Langkah 2 : Identifikasi relasi entitas *one to many*

Pada langkah ini adalah menggambarkan relasi antara entitas-entitas yang sudah dijelaskan sebelumnya. Dari beberapa entitas yang ada, terdapat beberapa relasi yang terjadi. Salah satu dari relasi tersebut adalah relasi *one to many*. Berikut adalah relasi *one to many* beserta entitasnya :

- Karyawan – melakukan – absen

Relasi tersebut menjelaskan bahwa setiap satu karyawan dapat melakukan beberapa presensi, sedangkan satu presensi hanya dilakukan oleh satu karyawan. Relasi tersebut mengambil *primary key* dari entitas karyawan. *Primary key* tersebut kemudian diidentifikasi sebagai *foreign key* di tabel presensi. Kemudian presensi_tanggal, presensi_jam_checkin sebagai candidate key, dan terdapat atribut presensi_jam_checkout. Tabel relasi ditunjukkan pada Tabel 3.13.

Tabel 3.13 Tabel Relasi Karyawan-Melakukan-Absensi

Relasi	Atribut	Keterangan
Karyawan – melakukan – presensi	karyawan_id	Foreign Key
	presensi_tanggal	Candidate Key
	presensi_jam_checkin	Candidate Key
	presensi_jam_checkout	-

Langkah 3 : Identifikasi relasi entitas *many to many*

Pada langkah ini adalah menggambarkan relasi antara entitas-entitas yang sudah dijelaskan sebelumnya. Dari beberapa entitas yang ada, terdapat beberapa relasi yang terjadi. Salah satu dari relasi tersebut adalah relasi *many to many*. Berikut adalah relasi *many to many* beserta entitasnya :

- Karyawan – melayani – pelanggan – mengorder – *treatment*
Relasi tersebut menjelaskan bahwa satu karyawan bisa melayani beberapa pelanggan, dan satu pelanggan bisa dilayani oleh beberapa karyawan. Kemudian satu pelanggan bisa mengorder beberapa *treatment*, dan setiap satu *treatment* bisa diorder oleh beberapa pelanggan.

Relasi tersebut menghasilkan tabel yang berisi *primary key* dari entitas karyawan, pelanggan, dan *treatment*. *Primary key* tersebut kemudian diidentifikasi sebagai *foreign key* di tabel yang baru yang bernama tabel *tbl_order*. Di dalam tabel *tbl_order* kemudian ditambahkan atribut baru yaitu *tbl_order_tanggal*, *tbl_order_status*, *tbl_total_harga*, *tbl_jumlah_order*. Tabel relasi ditunjukkan pada Tabel 3.14.

Tabel 3.14 Tabel Relasi Karyawan – melayani – pelanggan – mengorder – *treatment*

Relasi	Atribut	Keterangan
Karyawan – melayani – pelanggan – mengorder – <i>treatment</i>	Karyawan_id	Foreign Key
	Pelanggan_id	Foreign Key
	<i>Treatment_id</i>	Foreign Key
	Tbl_order_tanggal	-
	Tbl_order_status	-
	Tbl_total_harga	-

	Tbl_jumlah_order	-
--	------------------	---

Langkah 4 : Identifikasi *treatment* – memiliki – jenis_nama (*multivalued attribute* jenis_nama)

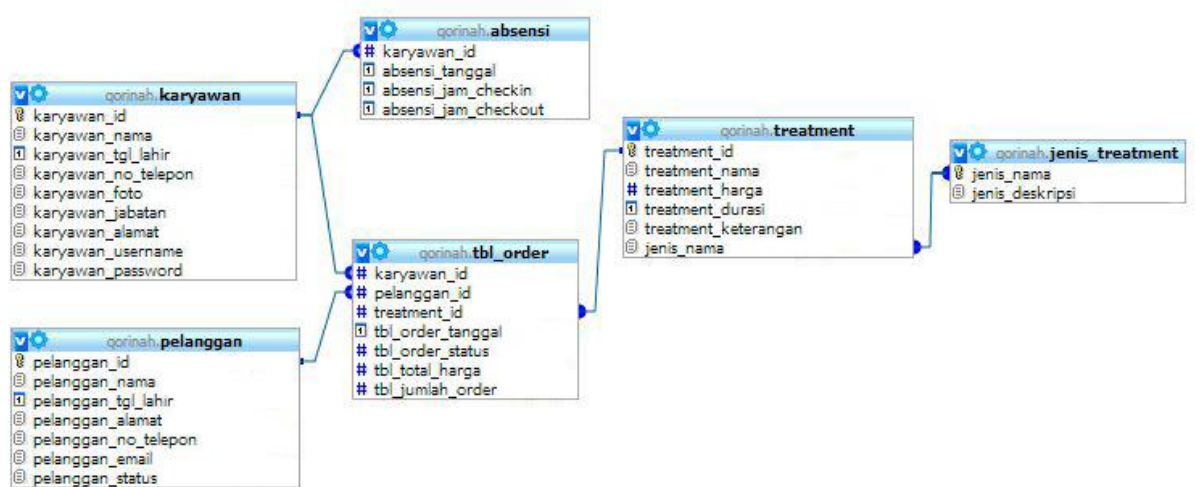
Pada langkah ini adalah menjelaskan *multivalued atributte* yang terdapat pada tabel *treatment*. Tabel *treatment* memiliki satu *multivalued attribute*, yaitu jenis_nama. Jenis_nama ada pada tabel jenis_*treatment*, yang memiliki atribut jenis_nama dan jenis_deskripsi. Dari relasi tersebut maka tabel *treatment* memiliki atribut sebagai berikut : *treatment_id*, *treatment_nama*, *treatment_harga*, *treatment_durasi*, dan jenis_nama. Tabel relasi ditunjukkan pada Tabel 3.15.

Tabel 3.15 Tabel Relasi *treatment* – memiliki – jenis_nama

Relasi	Atribut	Keterangan
<i>Treatment</i> – memiliki – jenis_nama	<i>Treatment_id</i>	Primary key
	<i>Treatment_nama</i>	-
	<i>Treatment_harga</i>	-
Relasi	Atribut	Keterangan
<i>Treatment</i> – memiliki – jenis_nama	<i>Treatment_durasi</i>	-
	<i>Treatment_keterangan</i>	-
	Jenis_nama	<i>Multivalued attribute</i>

3.4.2.2 Relasi Tabel

Relasi tabel *database* pada sistem informasi manajemen ini dapat dilihat pada gambar berikut.



Gambar 3.1 Desain Relasi Tabel

3.4.3 Normalisasi

Normalisasi adalah proses yang digunakan untuk menentukan pengelompokan atribut-atribut dalam sebuah relasi sehingga diperoleh relasi baru yang berstruktur baik (Kadir, 2009). Terdapat beberapa bentuk normalisasi *database*, berikut adalah bentuk yang digunakan dalam mendesain *database* sistem informasi manajemen Qorinah Day Spa and Salon.

3.4.3.1 Bentuk Normal Pertama (1NF)

Tahap pertama normalisasi dilakukan uji coba terhadap atribut. Syarat dari bentuk normal pertama setiap atribut bersifat *atomic*, yaitu tidak adanya atribut yang bersifat *multivalued* dalam tiap baris. Bentuk normal pertama digunakan pada tabel yang belum ternormalisasi, Berdasarkan lampiran tersebut, kemudian dihasilkan 5 tabel yaitu pelanggan, *treatment*, presensi, dan karyawan, ditambah tabel *tbl_order* sebagai tabel relasi. Bentuk normal pertama digunakan pada tabel yang belum ternormalisasi, tabel tersebut ada pada Lampiran 4.

Masing-masing tabel telah dibuat untuk menyimpan atribut data yang bernilai tunggal (*atomic*). Tabel pelanggan, misalnya, dianggap dalam bentuk

normal pertama karena didalamnya tidak dimungkinkan menyimpan atribut data pelanggan yang sama dalam satu *field*, jika ditemui hal tersebut maka akan dipisah menjadi *field* yang berbeda. Contoh pengujian terhadap tabel tersebut adalah pada *field* alamat, dapat menerima inputan alamat pelanggan tanpa harus ada pembagian daerah, kecamatan, kelurahan, dsb. Begitu juga dengan *field* yang lainnya dianggap sebagai *atomic* karena *record* di dalamnya bernilai tunggal. Setelah memeriksa semua atribut dalam desain *database*, disimpulkan bahwa skema ini telah dalam Bentuk Normal Pertama. Oleh karena itu, tidak ada perubahan lebih lanjut yang perlu dilakukan karena telah lulus uji pertama normalisasi.

3.4.3.2 Bentuk Normal Kedua (2NF)

Bentuk normal kedua digunakan untuk memastikan semua tabel harus memiliki *primary key* seperti yang ditunjukkan pada Lampiran 4. Setelah tabel melalui bentuk normal pertama, maka akan ditemui beberapa tabel yang merupakan hasil dekomposisi. Atribut data dalam tabel tersebut wajib memiliki ikatan dengan *primary key* di masing-masing tabel.

Tabel pelanggan, misalnya menggunakan *pelanggan_id* sebagai *primary key* dalam tabel. Tabel karyawan menggunakan *karyawan_id* sebagai *primary key* dalam tabel. Tabel *treatment* menggunakan *treatment_id* sebagai *primary key* dalam tabel. Namun, jika melihat pada desain *database* maka antara tabel pelanggan dan *treatment* menghasilkan tabel baru, yaitu tabel order yang merupakan hasil dekomposisi pada bentuk normal kedua. *Primary key* dari tabel pelanggan dan *treatment* disimpan didalamnya.

Demikian hingga bisa dikatakan bahwa skema ini telah dalam Bentuk Normal Kedua dan tidak diperlukan perubahan lebih lanjut.

3.4.3.3 Bentuk Normal Ketiga (3NF)

Bentuk normal ketiga menjelaskan bahwa dalam desain *database*, terbukti bahwa setiap data pada masing-masing tabel memiliki ketergantungan pada *primary key* pada setiap tabel yang ditempati oleh data tersebut, seperti yang ditunjukkan pada Lampiran 4. Misalkan pada tabel pelanggan, setiap data pada

tabel pelanggan harus memiliki ketergantungan hanya pada *primary key* di tabel tersebut yaitu pelanggan_id. Begitu pula dengan tabel-tabel lainnya. Demikian bisa dikatakan bahwa skema ini telah dalam Bentuk Normal Ketiga, dan tidak memerlukan perubahan lebih lanjut.

3.4.4 Physical Design

Proses implementasi dari hasil desain *database* secara logis akan dijelaskan pada tahap ini. Rancangan *database* menggunakan aplikasi DBMS (*Data Base Management System*) MySQL. Tahap ini bisa dianggap sebagai transformasi perancangan logis dari *database* ke dalam bentuk fisik yang dapat disimpan pada media penyimpanan.

3.4.4.1 Struktur Tabel dan SQL Statement

Berdasarkan apa yang dijelaskan pada tahap sebelumnya, bahwa terdapat lima entitas utama yang akan diubah menjadi tabel dalam *database*. Kelima entitas tersebut yaitu: pelanggan, karyawan, *treatment*, jenis *treatment*, presensi, dan detail order. Berikut adalah struktur tabel dari *database* tersebut.

1. Tabel Pelanggan

Tabel pelanggan berisi pelanggan_id, pelanggan_nama, pelanggan_tgl_lahir, pelanggan_alamat, pelanggan_no_telepon, pelanggan_email, pelanggan_status. Tabel ini digunakan untuk menyimpan data pribadi pelanggan. Tabel ini merujuk kepada kartu pelanggan di Lampiran 5. Struktur tabel pelanggan dapat dilihat pada tabel 3.16.

Tabel 3.16 Tabel Pelanggan

Field	Type	Keterangan
pelanggan_id	Int (11)	NOT NULL, Primary Key
pelanggan_nama	Varchar (50)	NOT NULL
pelanggan_tgl_lahir	Date	NOT NULL
pelanggan_alamat	Varchar (255)	NOT NULL
pelanggan_no_telepon	Int (11)	NOT NULL
pelanggan_email	Varchar (50)	-

pelanggan_status	Varchar (25)	-
------------------	--------------	---

Berikut adalah syntax SQL untuk membuat tabel pelanggan:

```
CREATE TABLE pelanggan(
  pelanggan_id INTEGER(11) NOT NULL AUTO_INCREMENT,
  pelanggan_nama VARCHAR(50) NOT NULL,
  pelanggan_alamat VARCHAR(100) NOT NULL,
  pelanggan_tgl_lahir DATE NOT NULL,
  pelanggan_no_telepon INTEGER(11) NOT NULL,
  pelanggan_email VARCHAR(50),
  pelanggan_status VARCHAR(25)
  PRIMARY KEY(pelanggan_id));
```

2. Tabel Karyawan

Tabel pelanggan berisi karyawan_id, karyawan_nama, karyawan_tgl_lahir, karyawan_alamat, karyawan_no_telepon, karyawan_email, karyawan_jabatan, karyawan_username, karyawan_password, karyawan_foto. Tabel ini digunakan untuk menyimpan data pribadi karyawan. Struktur tabel pelanggan dapat dilihat pada tabel 3.17.

Tabel 3.17 Tabel Karyawan

Field	Type	Keterangan
karyawan_id	Int (11)	NOT NULL, Primary Key
karyawan_nama	Varchar (50)	NOT NULL
Field	Type	Keterangan
karyawan_tgl_lahir	date	NOT NULL
karyawan _no_telepon	Varchar (15)	NOT NULL
karyawan_foto	Varchar (200)	-
karyawan_jabatan	Varchar (20)	NOT NULL
karyawan_alamat	Varchar (150)	NOT NULL
karyawan_username	Varchar(20)	NOT NULL
karyawan_password	Varchar(20)	NOT NULL

Berikut adalah syntax SQL untuk membuat table karyawan:

```
CREATE TABLE karyawan(
  karyawan_id INTEGER(11) NOT NULL AUTO_INCREMENT,
  karyawan_nama VARCHAR(50) NOT NULL,
  karyawan_alamat VARCHAR(150) NOT NULL,
```

```

karyawan_tgl_lahir DATE NOT NULL,
karyawan_no_telepon INTEGER(11) NOT NULL,
karyawan_foto VARCHAR(20),
karyawan_jabatan VARCHAR(20) NOT NULL,
karyawan_username VARCHAR(20) NOT NULL,
karyawan_password VARCHAR(40) NOT NULL
PRIMARY KEY(karyawan_id));

```

3. Tabel *Treatment*

Tabel pelanggan berisi *treatment_id*, *treatment_nama*, *treatment_harga*, *treatment_durasi*, *treatment_keterangan*. Tabel ini digunakan untuk menyimpan data *treatment*. Tabel ini merujuk kepada Struktur tabel *treatment* dapat dilihat pada tabel 3.18.

Tabel 3.18 Tabel *Treatment*

Field	Type	Keterangan
<i>treatment_id</i>	Int (11)	NOT NULL, Primary Key
<i>treatment_nama</i>	Varchar (50)	NOT NULL
<i>treatment_harga</i>	Float	NOT NULL
<i>jenis_nama</i>	Varchar (50)	NOT NULL
<i>treatment_durasi</i>	Varchar (50)	NOT NULL
<i>treatment_keterangan</i>	Varchar (50)	-

Berikut adalah syntax SQL untuk membuat *table treatment*:

```

CREATE TABLE treatment(
treatment_id INTEGER(10) NOT NULL AUTO_INCREMENT,
treatment_nama VARCHAR(50) NOT NULL,
treatment_harga FLOAT NOT NULL,
treatment_durasi TIME NOT NULL,
treatment_keterangan VARCHAR (25),
jenis_nama VARCHAR (50) NOT NULL)
PRIMARY KEY(treatment_id);
FOREIGN KEY (jenis_nama)
REFERENCE jenis_treatment(jenis_nama);

```

4. Tabel *Presensi*

Tabel presensi berisi *karyawan_id*, *presensi_tanggal*, *presensi_jam_checkin*, *presensi_jam_checkout*. Tabel ini digunakan untuk menyimpan data presensi karyawan. Struktur tabel *absnsi* dapat dilihat pada tabel 3.19.

Tabel 3.19 Tabel Presensi

Field	Type	Keterangan
karyawan_id	Int (11)	Foreign Key
presensi_tanggal	Date	NOT NULL, Candidate Key
presensi_jam_checkin	Time	NOT NULL, Candidate Key
presensi_jam_checkout	Time	-

Berikut adalah syntax SQL untuk membuat tabel presensi:

```
CREATE TABLE presensi(  
  karyawan_id INTEGER (11) NOT NULL,  
  presensi_tanggal DATE NOT NULL,  
  presensi_jam_checkin TIME NOT NULL,  
  presensi_jam_checkout TIME,  
  FOREIGN KEY (karyawan_id)  
  REFERENCES karyawan(karyawan_id));  
INDEX presensi_tanggal DATE,  
INDEX presensi_jam_checkin TIME );
```

5. Tabel Order

Tabel order merupakan tabel relasi antara entitas karyawan, pelanggan, dan *treatment*. Tabel order berisi karyawan_id, pelanggan_id, treatment_id, order_tanggal, order_status, detail_total_harga, detail_jumlah_order. Tabel ini digunakan untuk menyimpan data order pelanggan. Struktur tabel order dapat dilihat pada tabel 3.20

Tabel 3.20 Tabel Order

Field	Type	Keterangan
karyawan_id	Int (11)	Foreign Key
pelanggan_id	Int (11)	Foreign Key
treatment_id	Int (11)	Foreign Key
tbl_order_tanggal	Date	NOT NULL, Candidate Key
tbl_order_status	Tinyint(1)	-
tbl_total_harga	Float	-
tbl_jumlah_order	Int (11)	-

Berikut adalah syntax SQL untuk membuat tabel tbl_order:

```
CREATE TABLE tbl_order(  
  karyawan_id INTEGER (11) NOT NULL,  
  treatment_id INTEGER (11) NOTNULL,  
  pelanggan_id INTEGER (11) NOT NULL,
```

```

tbl_order_tanggal DATE NOT NULL,
tbl_order_status BOOL,
tbl_total_harga FLOAT,
tbl_jumlah_order INT (11),
FOREIGN KEY(karyawan_id)
REFERENCES karyawan(karyawan_id),
FOREIGN KEY(treatment_id)
REFERENCES treatment(treatment_id),
FOREIGN KEY(pelanggan_id)
REFERENCES pelanggan(pelanggan_id));
INDEX tbl_order_tanggal DATE );

```

3.5 Desain Aplikasi

3.5.1 Use Case Diagram

Use case diagram adalah salah satu bentuk pemodelan dalam metode pengembangan perangkat lunak. *Use case diagram* menjelaskan skenario interaksi yang dilakukan antara aktor terhadap sistem. Sistem informasi manajemen salon ini terdapat 2 aktor, yaitu terapis dan manager.

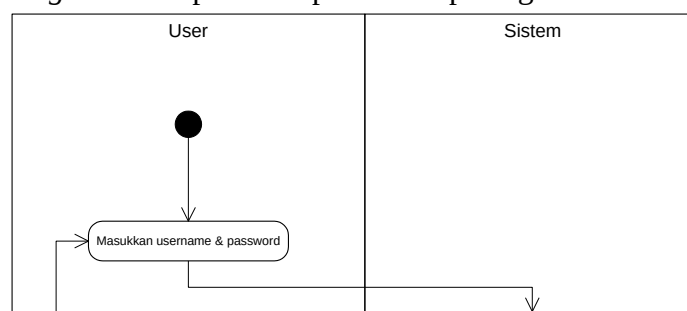
Terapis berwenang untuk melakukan manajemen data pelanggan, manajemen order, serta melihat report presensi. Bentuk manajemen data meliputi *input*, *edit*, dan *delete* data. Manager berhak untuk mengakses semua menu. Tabel 3.4 Tabel Interaksi telah menjelaskan bahasan mengenai aktor dan wewenang dari masing-masing aktor. Proses tersebut dapat dilihat di Lampiran 6.

3.5.2 Activity Diagram

Activity diagram digunakan untuk menggambarkan berbagai alur aktifitas sistem dan siapa *user* yang menjalankan. Diagram ini berkaitan dengan proses bisnis dan alur kerja, yang sudah digambarkan dari *use case diagram* Lampiran 6. Berikut ini adalah gambaran activity diagram dari sistem informasi manajemen salon.

3.5.2.1 Activity Diagram Login Sistem

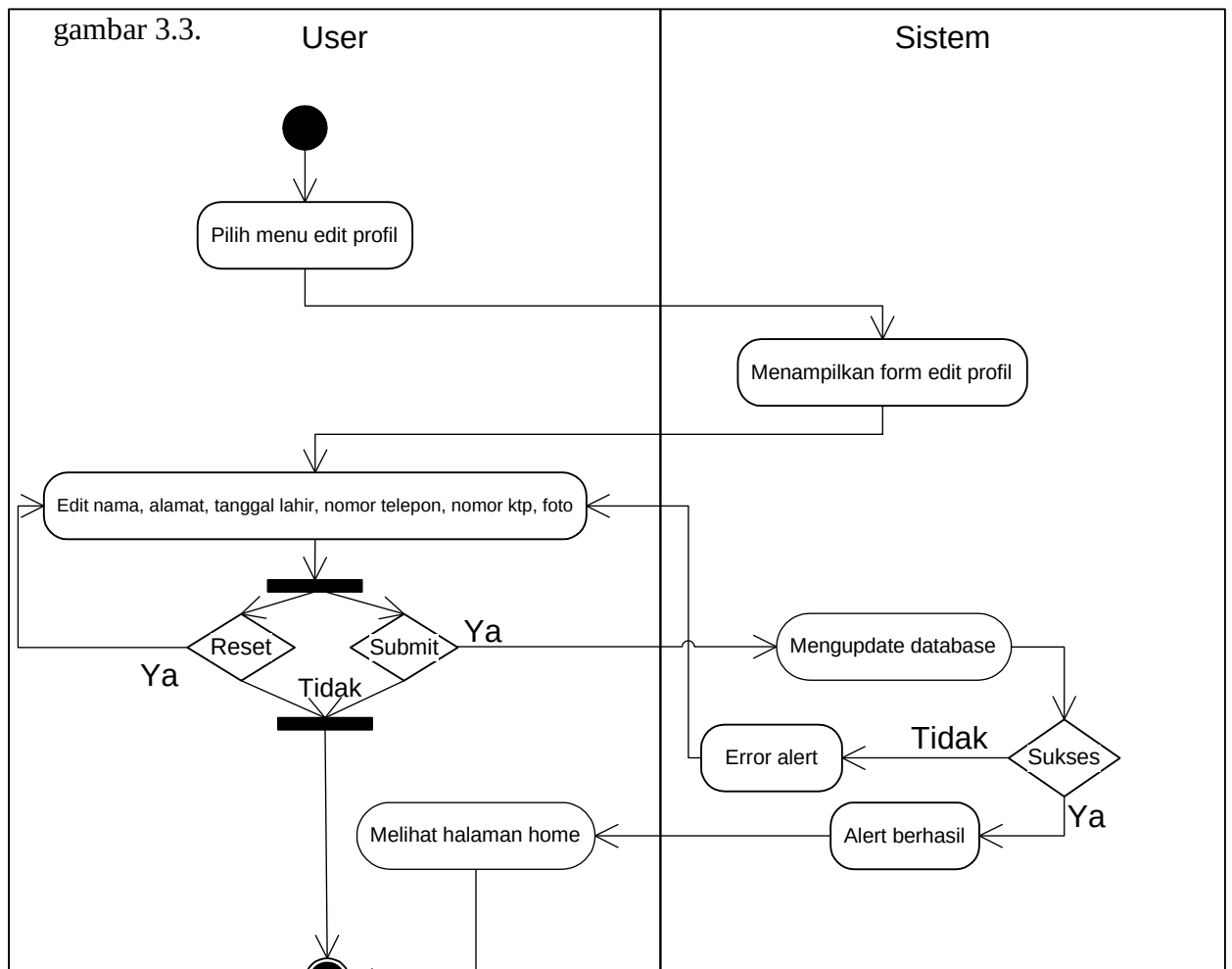
Untuk dapat mengakses sistem, maka *user* harus masuk ke dalam sistem terlebih dahulu. *User* diharuskan melakukan *login* dengan menggunakan *username* dan *password* masing-masing. Apabila *user* berhasil masuk maka akan muncul tampilan halaman *home* dari sistem. Sedangkan jika gagal, diharuskan mengulang proses *login*. Detail proses dapat dilihat pada gambar 3.2.



Gambar 3.2 Activity Diagram Login

3.5.2.2 Activity Diagram Edit Profil

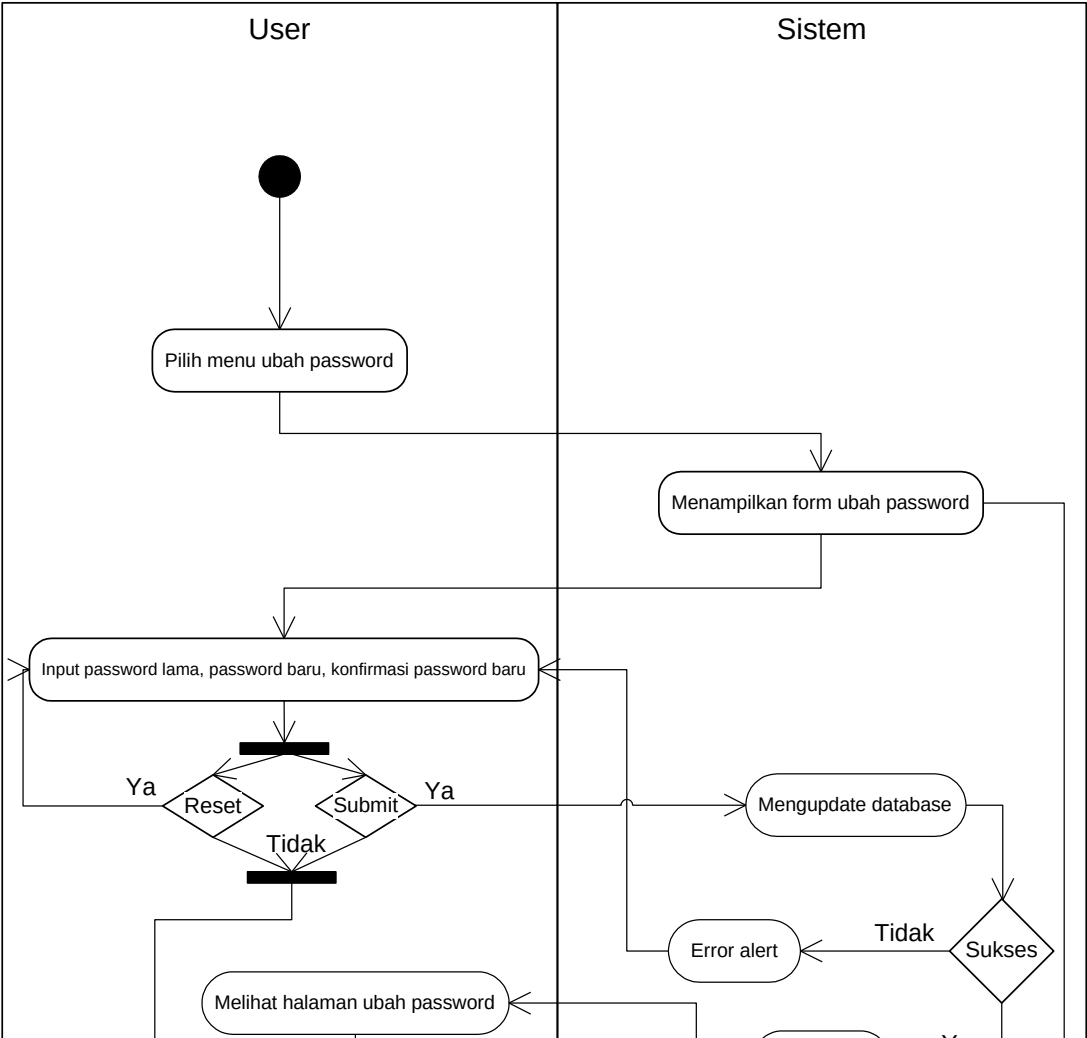
Terapis dan manager berhak untuk merubah nama, alamat, tanggal lahir, nomor ktp, nomor telepon, dan foto. Detail proses diatas digambarkan dalam



Gambar 3.4 Activity Diagram Edit Profil

3.5.2.3 Activity Diagram Ubah Password

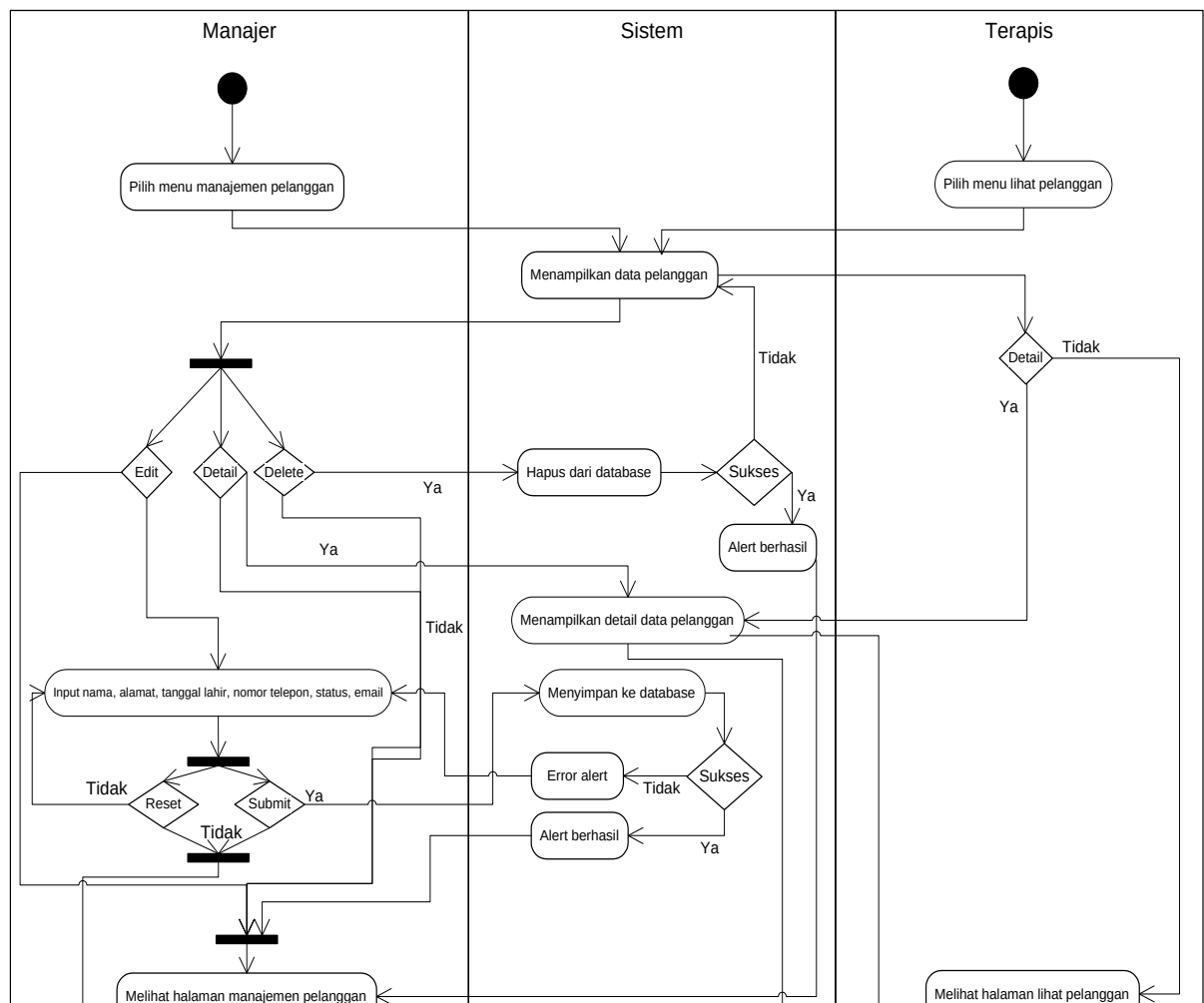
Terapis dan manager berhak untuk merubah *username* dan *password* jika dibutuhkan. Detail proses di atas digambarkan dalam gambar 3.4.



Gambar 3.4 *Activity Diagram* Ubah Password

3.5.2.4 Activity Diagram Manajemen Pelanggan

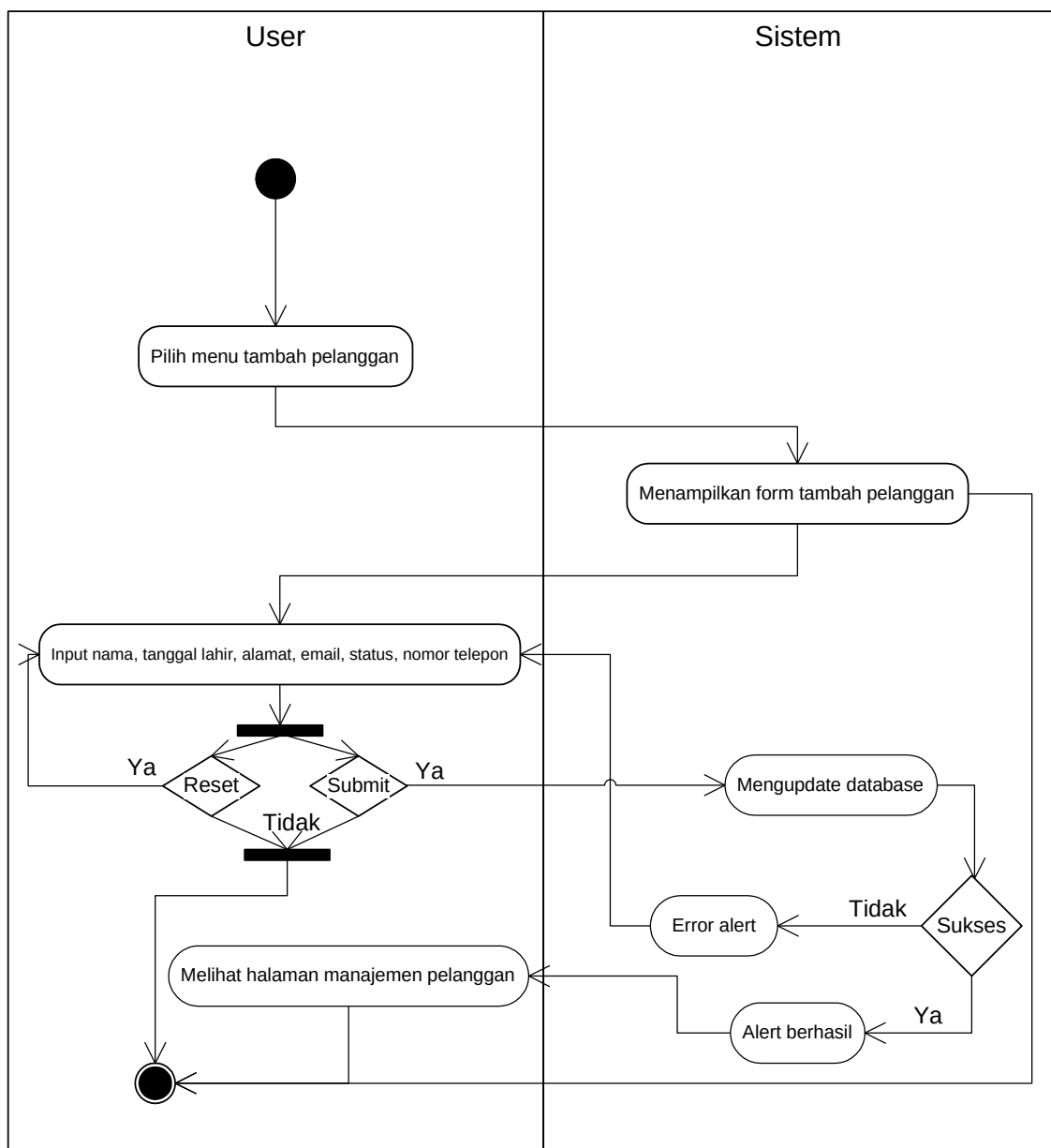
Manajemen pelanggan digunakan untuk mengelola pelanggan. Manajemen pelanggan memiliki beberapa fungsi yaitu tambah, *edit*, *delete*, dan detail. Fungsi – fungsi tersebut bisa dijalankan oleh semua *user*. Detail proses diatas digambarkan pada gambar 3.5



Gambar 3.5 Activity Diagram Manajemen Pelanggan

3.5.2.5 Activity Diagram Tambah Pelanggan

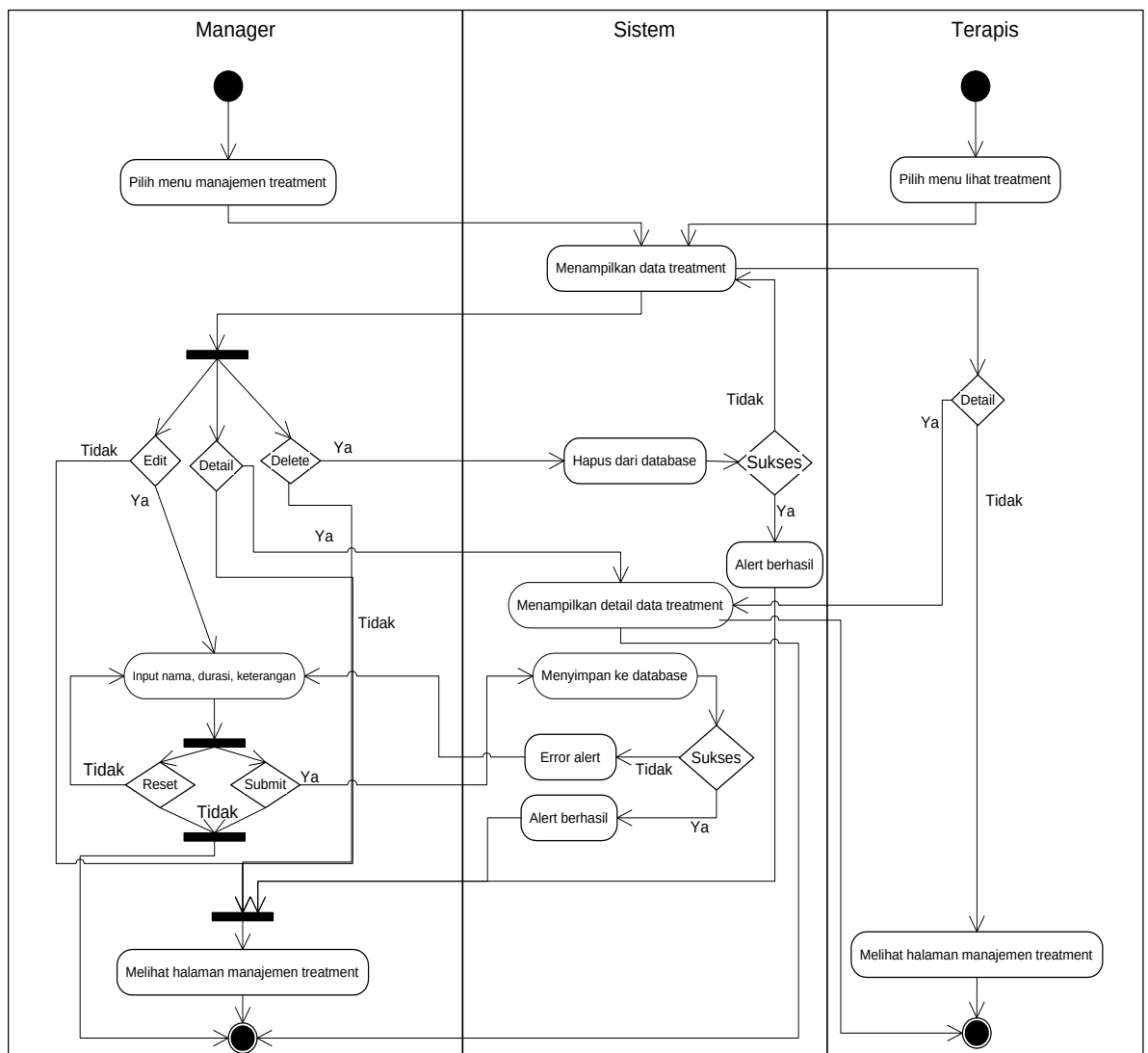
Tambah pelanggan digunakan untuk menambahkan pelanggan baru ke dalam database. Detail proses digambarkan dalam gambar 3.6.



Gambar 3.6 Activity Diagram Tambah Pelanggan

3.5.2.6 Activity Diagram Manajemen *Treatment* dan Lihat *Treatment*

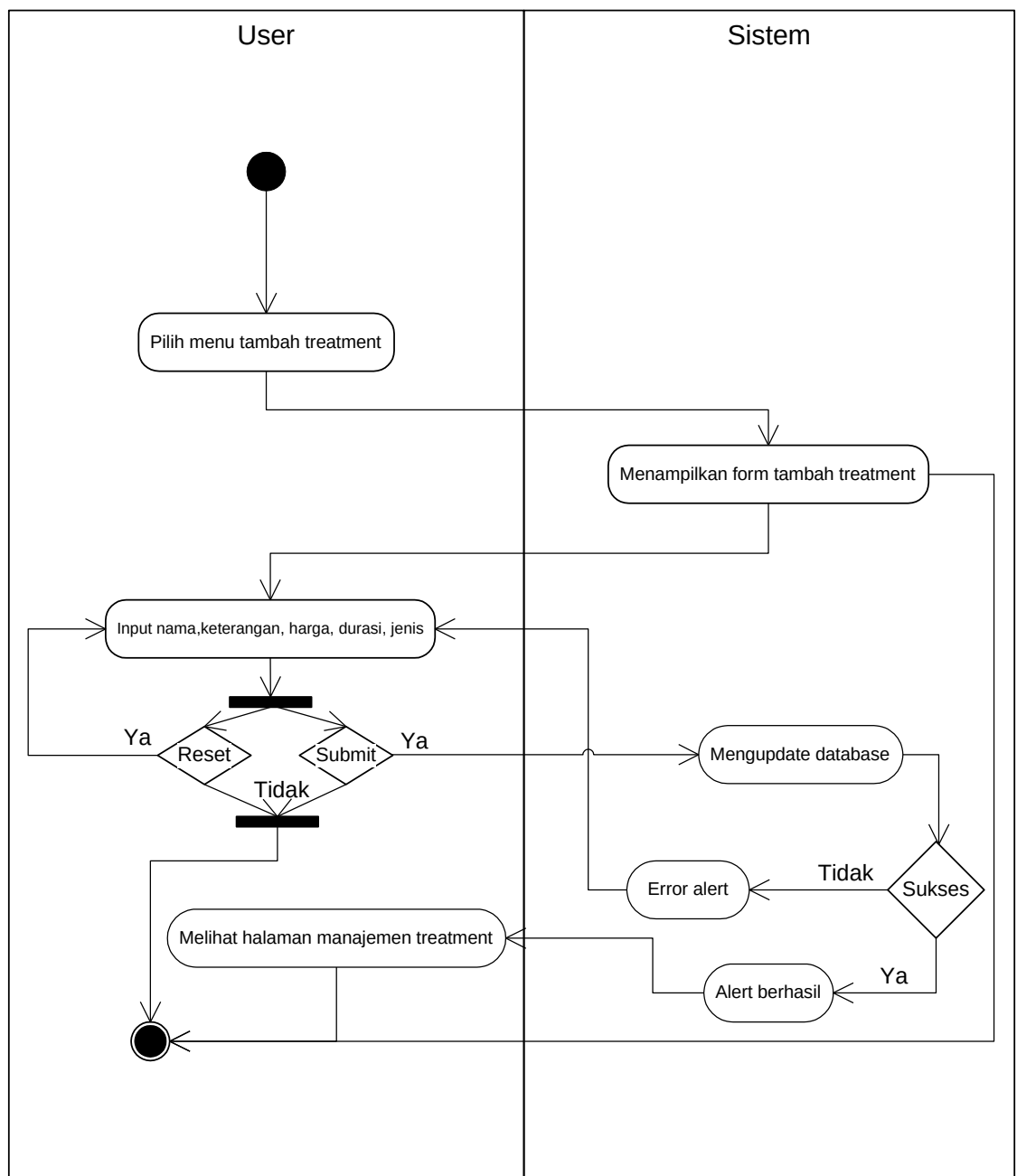
Manajemen pelanggan digunakan untuk mengelola *treatment*. Manajemen *treatment* memiliki beberapa fungsi yaitu tambah, *edit*, *delete*, dan detail. Fungsi – fungsi tersebut bisa dijalankan oleh semua *user*. Detail proses diatas digambarkan pada gambar 3.7



Gambar 3.7 Activity Diagram Manajemen Treatment

3.5.2.7 Activity Diagram Tambah Treatment

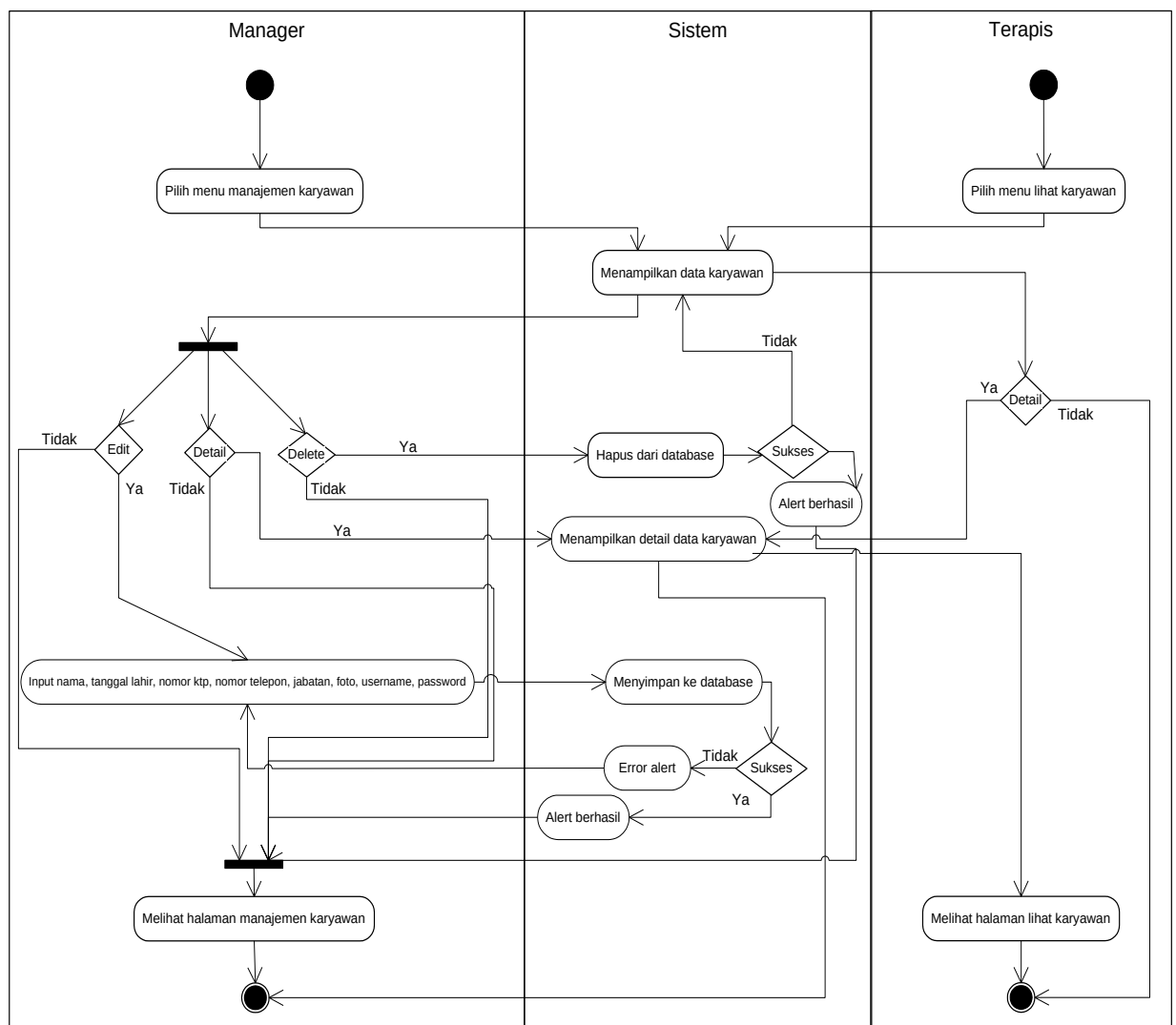
Tambah *treatment* digunakan untuk menambahkan *treatment* baru ke dalam database. Detail proses digambarkan dalam gambar 3.8.



Gambar 3.8 Activity Diagram Tambah Treatment

3.5.2.8 Activity Diagram Manajemen Karyawan dan Lihat Karyawan

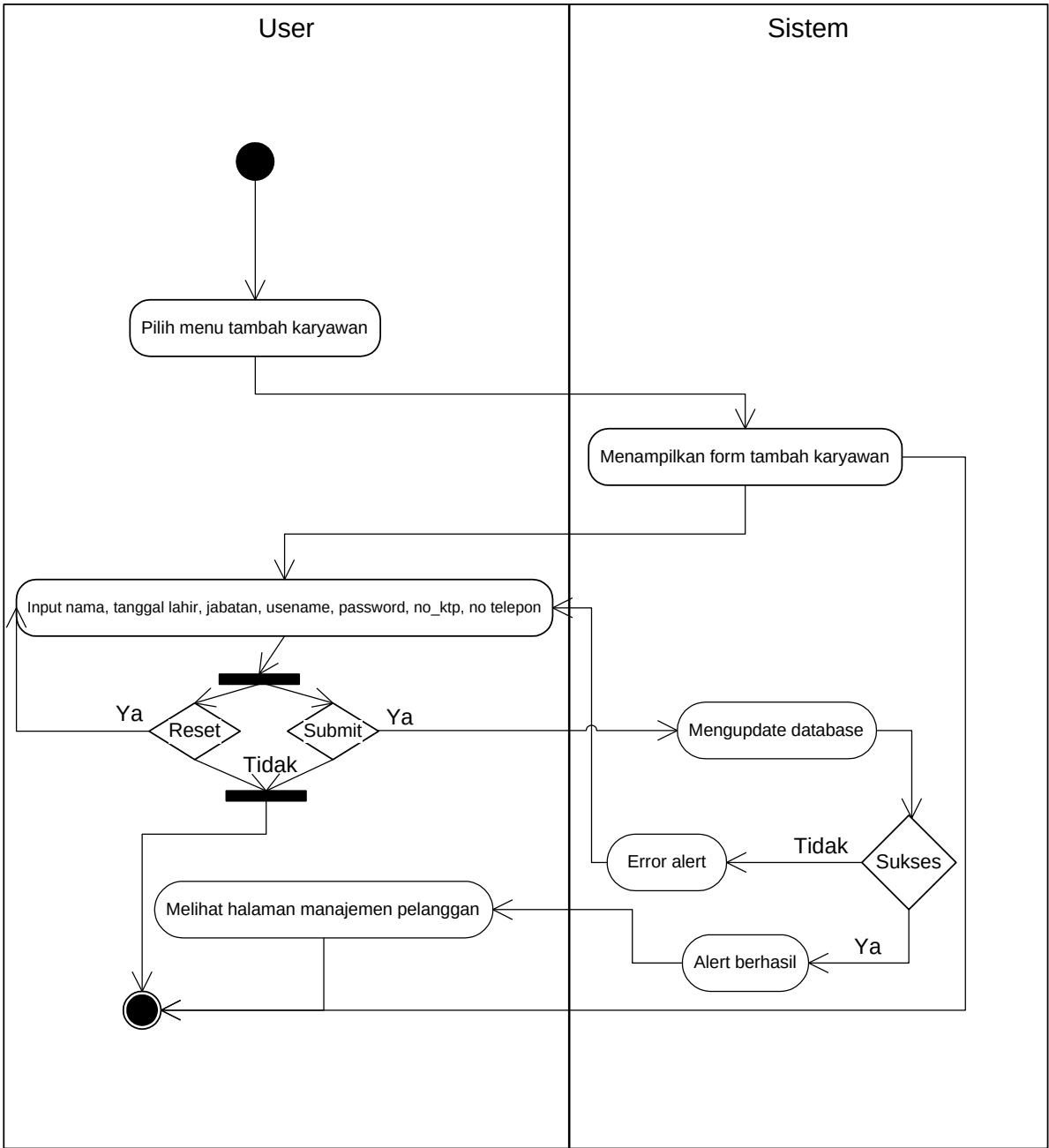
Manajemen karyawan digunakan untuk mengelola karyawan. Manajemen pelanggan memiliki beberapa fungsi yaitu tambah, *edit*, *delete*, dan detail. Namun manager memiliki wewenang lebih dalam mengelola manajemen karyawan jika diperlukan. Fungsi – fungsi tersebut bisa dijalankan oleh semua *user*. Detail proses diatas digambarkan pada gambar 3.9



Gambar 3.9 *Activity Diagram* Manajemen Karyawan

3.5.2.9 Activity Diagram Tambah Karyawan

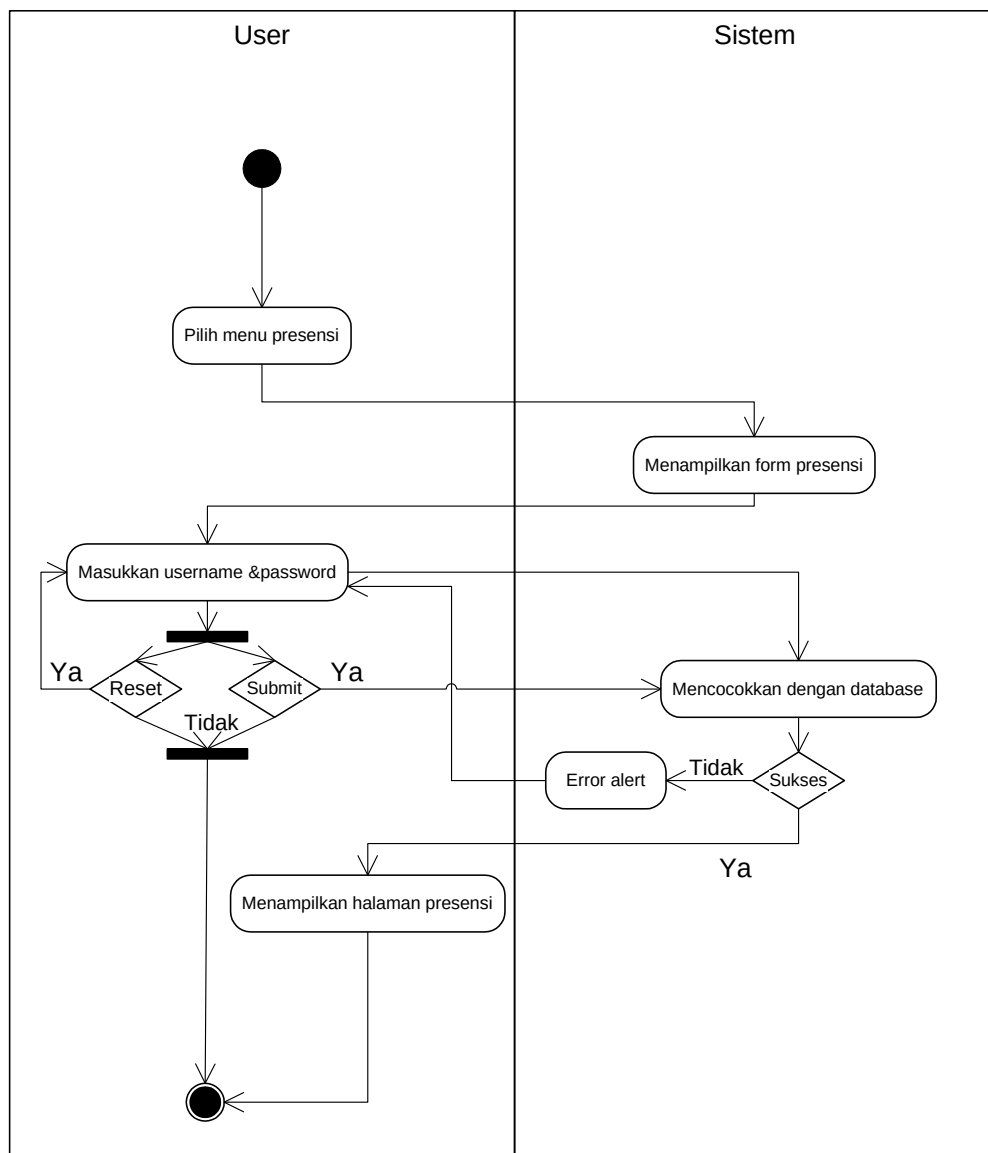
Tambah karyawan digunakan untuk menambahkan karyawan baru ke dalam database. Detail proses digambarkan dalam gambar 3.10.



Gambar 3.10 Activity Diagram Tambah Karyawan

3.5.2.10 Activity Diagram Presensi

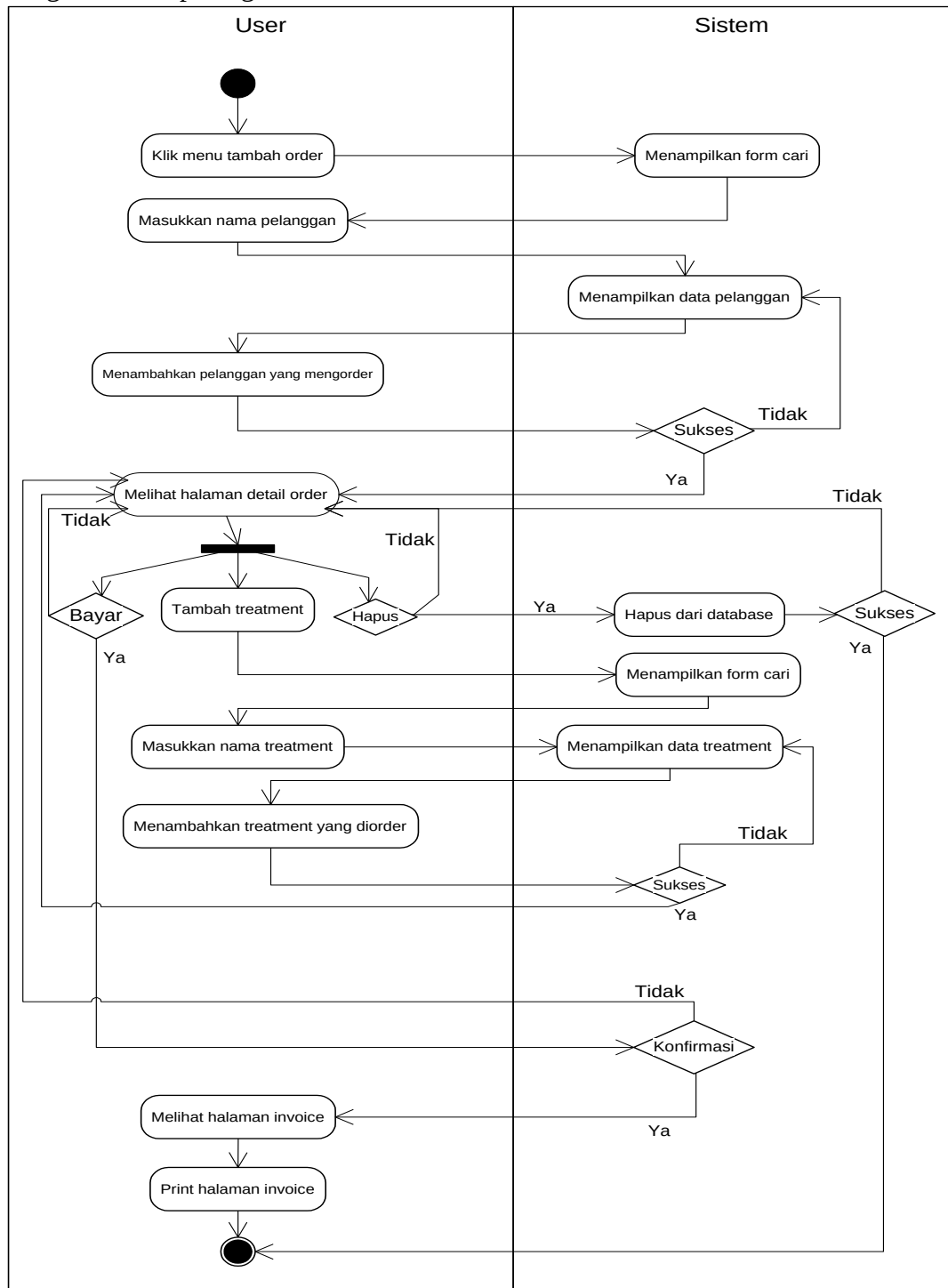
Presensi digunakan untuk melakukan presensi dan melihat data siapa saja karyawan yang sudah melakukan presensi. Presensi dilakukan oleh semua karyawan. Detail proses digambarkan pada gambar 3.11.



Gambar 3.11 Activity Diagram Presensi

3.5.2.11 Activity Diagram Manajemen Order dan Tambah Order

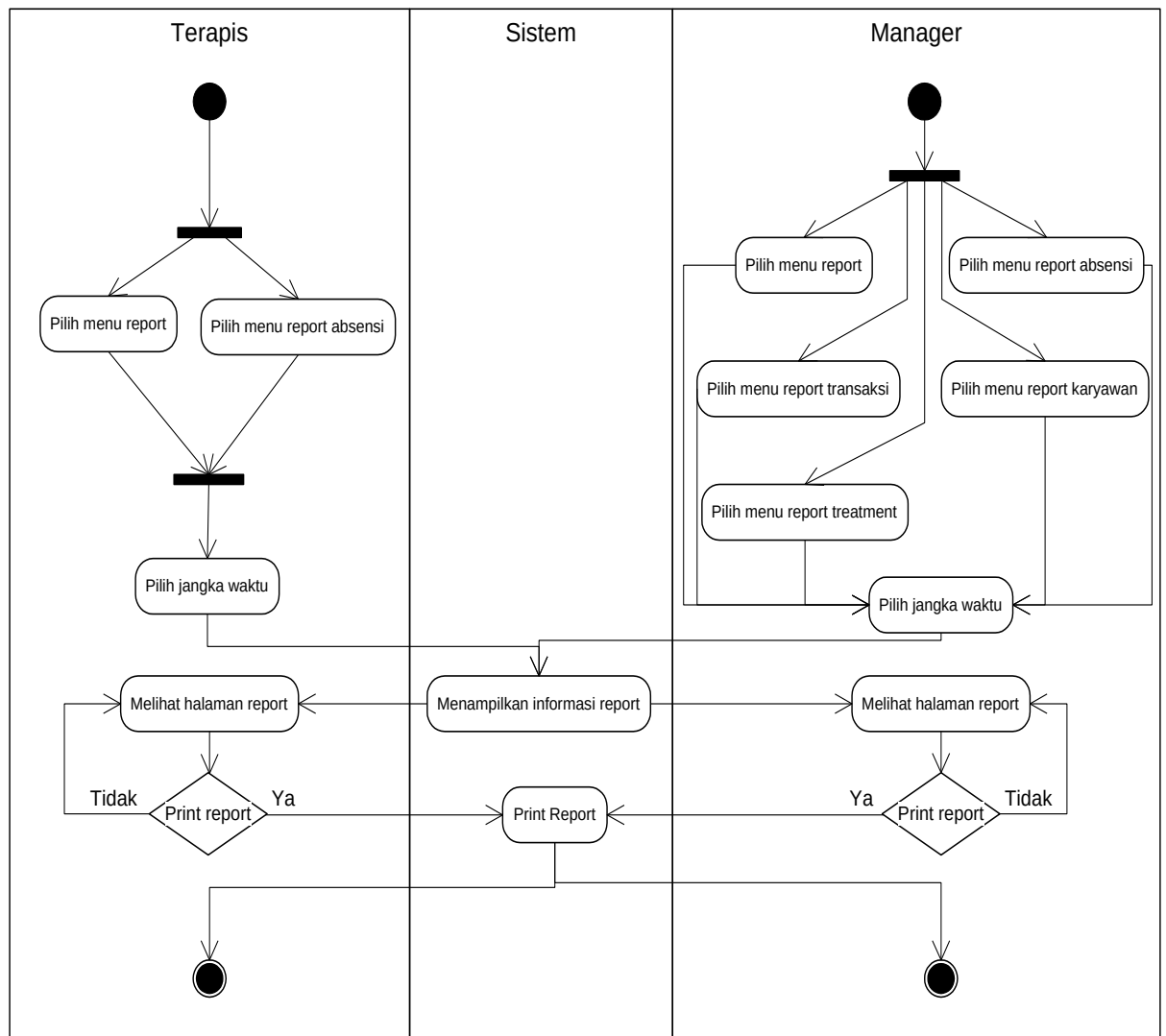
Manajemen order digunakan untuk mengelola order pelanggan. Manajemen order memiliki beberapa fungsi yaitu bayar, tambah order, dan hapus order. Fungsi – fungsi tersebut bisa dijalankan oleh semua *user*. Detail proses di atas digambarkan pada gambar 3.12



Gambar 3.12 *Activity Diagram* Manajemen Order dan Tambah Order

3.5.2.12 Activity Diagram Lihat Report

Lihat report digunakan untuk melihat report yang dibuat oleh sistem. Lihat report ini bisa dijalankan oleh semua *user*. Detail proses digambarkan pada gambar 3.13.



Gambar 3.13 *Activity Diagram* Lihat Report

3.6 Desain Antarmuka Sistem

Tahap ini dilakukan untuk mendesain tampilan dari sistem yang dibuat berdasarkan analisa dari tahap sebelumnya. Desain tampilan dibuat dengan memperhatikan kebutuhan *user* dandengan tujuan mempermudah penggunaan. Selain itu, tampilan juga dirancang dengan memperhatikan fungsi *input* dan *output* yang terdapat dalam sistem agar berjalan dengan baik.

Sistem informasi manajemen salon ini memiliki halaman utama untuk masing-masing *user*, yaitu halaman terapis dan halaman manager. Setiap *user* diharuskan melalui halaman antarmuka *login* sebelum masuk ke halaman utama tersebut.

3.6.1 Halaman Login

Halaman ini adalah halaman awal saat sistem informasi manajemen ini dijalankan. *User* menginputkan *username* dan *password* agar bisa masuk ke sistem. Ditunjukkan pada Gambar 3.14.

HEADER	
USERNAME : PASSWORD : LOGIN Klik untuk masuk ke halaman absen	
FOOTER	

Gambar 3.14 Login Site

3.6.2 Home Terapis

Halaman ini adalah halaman *home* saat terapis telah melakukan proses *login* di sistem. Halaman *home* terapis menampilkan profil terapis dan beberapa menu, yaitu *home*, edit profil, ubah *password*, manajemen order, lihat pelanggan, lihat treatment, dan report (laporan). Desain antarmuka *home* terapis ditunjukkan pada Gambar 3.15.

HEADER	
Home Edit Profil Ubah Password Manajemen Order + Tambah Order Lihat Pelanggan + Tambah Pelanggan Lihat Treatment Lihat Karyawan Report	Report Jumlah pelanggan : Jumlah pendapatan : Treatment Favorit :

Gambar 3.15 *Home Terapis*

3.6.3 *Home Manager*

Halaman ini adalah halaman *home* saat manajer telah melakukan proses *login* di sistem. Halaman *home* manager menampilkan profil manager dan semua menu yang bisa diakses oleh manager, antara lain : *home*, edit profil, ubah *password*, manajemen order, manajemen pelanggan, manajemen *treatment*, manajemen karyawan, dan report. Desain antarmuka home manager ditunjukkan pada Gambar 3.16.

HEADER	
Home Edit Profil Ubah Password Manajemen Order + Tambah Order Manajemen Pelanggan + Tambah Pelanggan Manajemen Treatment + Tambah Treatment Manajemen Karyawan + Tambah Karyawan Report - Report Presensi - Report Transaksi - Report Treatment	FOTO Nama : Alamat : Tanggal lahir : Nomor Telepon : Nomor KTP : Jabatan :
FOOTER	

Gambar 3.16 *Home Manager*

3.6.4 *Halaman Edit Profil*

Halaman ini merupakan halaman yang tampil saat *user* ingin mengedit profilnya. *User* akan mengedit dalam form edit dan kemudian mengklik tombol submit untuk mengupdate profilnya. Halaman edit profil bisa diakses oleh semua *user*. Desain antarmuka edit profil (*login* sebagai manajer) ditunjukkan pada

Gambar 3.17. Sedangkan desain antarmuka edit profil (*login* sebagai terapis) ditunjukkan pada Gambar 3.18.

HEADER	
Home Edit Profil Ubah Password Manajemen Order + Tambah Order Manajemen Pelanggan + Tambah Pelanggan Manajemen Treatment + Tambah Treatment Manajemen Karyawan + Tambah Karyawan Report - Report Presensi - Report Transaksi - Report Treatment	Edit Profil Nama : Alamat : Tanggal lahir : Nomor telepon : Nomor ktp : Submit Reset
FOOTER	

Gambar 3.17 Halaman Edit Profil(Login Sebagai Manajer)

HEADER	
Home Edit Profil Ubah Password Manajemen Order + Tambah Order Lihat Pelanggan + Tambah Pelanggan Lihat Treatment Lihat Karyawan Report - Report Presensi	Edit Profil Nama : Alamat : Tanggal lahir : Nomor telepon : Nomor ktp : Submit Reset
FOOTER	

Gambar 3.18 Halaman Edit Profil (*Login* Sebagai Terapis)

3.6.5 Halaman Ubah *Password*

Halaman ini merupakan halaman yang tampil saat *user* ingin merubah *password*. Halaman ini ditampilkan dalam bentuk form. *User* memasukkan *password* lama dan baru kemudian mengklik tombol submit untuk mengupdate *password* baru. Desain antarmuka ubah *password*(*login* sebagai manajer) ditunjukkan pada Gambar 3.19. Sedangkan desain antarmuka ubah *password* (*login* sebagai terapis) ditunjukkan pada Gambar 3.20.

HEADER	
Home Edit Profil Ubah Password Manajemen Order + Tambah Order Manajemen Pelanggan + Tambah Pelanggan Manajemen Treatment + Tambah Treatment Manajemen Karyawan + Tambah Karyawan Report - Report Presensi - Report Transaksi - Report Treatment	Ubah Password Password lama : Password baru : Konf password baru : Submit Reset

Gambar 3.19 Halaman Ubah *Password* (*Login* Sebagai Manajer)

HEADER	
Home Edit Profil Ubah Password Manajemen Order + Tambah Order Lihat Pelanggan + Tambah Pelanggan Lihat Treatment Lihat Karyawan Report - Report Presensi	Ubah Password Password lama : Password baru : Konf password baru : Submit Reset
FOOTER	

Gambar 3.20 Halaman Ubah *Password* (*Login* Sebagai Terapis)

3.6.6 Halaman Manajemen Pelanggan

Halaman ini merupakan halaman yang tampil saat *user* memilih menu manajemen pelanggan. Halaman ini menampilkan nama-nama pelanggan yang telah terdaftar di sistem. Halaman ini memiliki *action*, yaitu detail edit, dan *delete*. Halaman ini dapat diakses oleh semua *user*. Desain antarmuka manajemen pelanggan (*login* sebagai manajer) ditunjukkan pada Gambar 3.21. Sedangkan desain antarmuka manajemen pelanggan (*login* sebagai terapis) ditunjukkan pada Gambar 3.22.

HEADER																			
Home Edit Profil Ubah Password Manajemen Order + Tambah Order Manajemen Pelanggan + Tambah Pelanggan Manajemen Treatment + Tambah Treatment Manajemen Karyawan + Tambah Karyawan Report - Report Presensi - Report Transaksi - Report Treatment	Manajemen Pelanggan <table border="1"> <thead> <tr> <th>No.</th> <th>Nama</th> <th colspan="3">Action</th> </tr> <tr> <th></th> <th></th> <th>Detail</th> <th>Edit</th> <th>Delete</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table>				No.	Nama	Action					Detail	Edit	Delete					
	No.	Nama	Action																
			Detail	Edit	Delete														
Submit Reset																			
FOOTER																			

Gambar 3.21 Halaman Manajemen Pelanggan (*Login Sebagai Manager*)

HEADER																			
Home Edit Profil Ubah Password Manajemen Order + Tambah Order Lihat Pelanggan + Tambah Pelanggan Lihat Treatment Lihat Karyawan Report - Report Presensi	Lihat Pelanggan <table border="1"> <thead> <tr> <th>No.</th> <th>Nama</th> <th colspan="3">Action</th> </tr> <tr> <th></th> <th></th> <th>Detail</th> <th>Edit</th> <th></th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table>				No.	Nama	Action					Detail	Edit						
	No.	Nama	Action																
			Detail	Edit															
Submit Reset																			
FOOTER																			

Gambar 3.22 Halaman Manajemen Pelanggan(*Login sebagai Terapis*)

3.6.7 Halaman Tambah Pelanggan

Tambah pelanggan merupakan sub-menu dari menu manajemen pelanggan. Halaman ini berbentuk form, *user* akan memasukkan nama, alamat, tanggal lahir, dsb kemudian mengklik tombol submit untuk menyimpan data pelanggan baru. Desain antarmuka tambah pelanggan (*login sebagai manager*) ditunjukkan pada Gambar 3.23. Sedangkan desain antarmuka tambah pelanggan (*login sebagai terapis*) ditunjukkan pada Gambar 3.24

HEADER	
Home Edit Profil Ubah Password Manajemen Order + Tambah Order Manajemen Pelanggan + Tambah Pelanggan Manajemen Treatment + Tambah Treatment Manajemen Karyawan + Tambah Karyawan Report - Report Presensi - Report Transaksi - Report Treatment	Tambah Pelanggan
	Nama :
	Alamat :
	Tanggal lahir :
	Nomor telepon :
Submit Reset	
FOOTER	

Gambar 3.23 Halaman Tambah Pelanggan (*Login Sebagai Manager*)

HEADER	
Home Edit Profil Ubah Password Manajemen Order + Tambah Order Lihat Pelanggan + Tambah Pelanggan Lihat Treatment Lihat Karyawan Report - Report Presensi	Tambah Pelanggan Nama : Alamat : Tanggal lahir : Nomor telepon : Submit Reset
FOOTER	

Gambar 3.24 Halaman Tambah Pelanggan (*Login sebagai Terapis*)

3.6.8 Halaman Manajemen *Treatment*

Halaman ini merupakan halaman yang tampil saat *user* memilih menu manajemen *treatment*. Halaman ini menampilkan nama-nama *treatment* yang telah terdaftar di sistem. Halaman ini memiliki *action*, yaitu detail edit, dan *delete*. Halaman ini dapat diakses oleh semua *user*. Desain antarmuka manajemen *treatment*(*login* sebagai manajer) ditunjukkan pada Gambar 3.25. Sedangkan desain antarmuka manajemen *treatment*(*login* sebagai terapis) ditunjukkan pada Gambar 3.26

HEADER																			
Home Edit Profil Ubah Password Manajemen Order + Tambah Order Manajemen Pelanggan + Tambah Pelanggan Manajemen Treatment + Tambah Treatment Manajemen Karyawan + Tambah Karyawan Report - Report Presensi - Report Transaksi - Report Treatment	Manajemen Treatment <table border="1"> <thead> <tr> <th>No.</th> <th>Nama</th> <th colspan="3">Action</th> </tr> <tr> <th></th> <th></th> <th>Detail</th> <th>Edit</th> <th>Delete</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table> Submit Reset				No.	Nama	Action					Detail	Edit	Delete					
No.	Nama	Action																	
		Detail	Edit	Delete															
FOOTER																			

Gambar 3.25 Halaman Manajemen *Treatment* (Login Sebagai Manager)

HEADER			
Home Edit Profil Ubah Password Manajemen Order + Tambah Order Lihat Pelanggan + Tambah Pelanggan Lihat Treatment Lihat Karyawan Report - Report Presensi	Lihat Treatment		
	No.	Nama	Action
			Detail
FOOTER			

Gambar 3.26 Halaman Manajemen *Treatment*(Login Sebagai Terapis)

3.6.9 Halaman Tambah *Treatment*

Tambah *treatment* merupakan sub-menu dari menu manajemen *treatment*. Halaman ini berbentuk form, *user* akan memasukkan nama, durasi, harga, dan keterangan kemudian mengklik tombol submit untuk menyimpan data *treatment* baru. Desain antarmuka tambah *treatment*(login sebagai manager)ditunjukkan pada Gambar 3.27.

HEADER	
Home Edit Profil Ubah Password Manajemen Order + Tambah Order Manajemen Pelanggan + Tambah Pelanggan Manajemen Treatment + Tambah Treatment Manajemen Karyawan + Tambah Karyawan Report - Report Presensi - Report Transaksi - Report Treatment	Tambah Treatment
	Nama :
	Durasi :
	Harga :
	Keterangan :
	Submit Reset
FOOTER	

Gambar 3.27 Halaman Tambah *Treatment* (*Login* Sebagai Manager)

3.6.10 Halaman Manajemen Karyawan

Halaman ini merupakan halaman yang tampil saat *user* memilih menu manajemen *treatment*. Halaman ini menampilkan nama-nama *treatment* yang telah terdaftar di sistem. Halaman ini memiliki *action*, yaitu detail edit, dan *delete*. Halaman ini dapat diakses oleh semua *user*. Desain antarmuka manajemen karyawan(*login* sebagai manager)ditunjukkan pada Gambar 3.28. Sedangkan desain antarmuka manajemen karyawan (*login* sebagai terapis) ditunjukkan pada Gambar 3.29

HEADER				
Home Edit Profil Ubah Password Manajemen Order + Tambah Order Manajemen Pelanggan + Tambah Pelanggan Manajemen Treatment + Tambah Treatment Manajemen Karyawan + Tambah Karyawan Report - Report Presensi - Report Transaksi - Report Treatment	Manajemen Karyawan			
	No.	Nama	Action	
			Detail	Edit
				Delete
Submit Reset				
FOOTER				

Gambar 3.28 Halaman Manajemen Karyawan(*Login* Sebagai Manager)

HEADER				
Home Edit Profil Ubah Password Manajemen Order + Tambah Order Lihat Pelanggan + Tambah Pelanggan Lihat Treatment Lihat Karyawan Report - Report Presensi	Lihat Karyawan			
	No.	Nama	Action	
			Detail	
Submit Reset				
FOOTER				

Gambar 3.29 Halaman Manajemen Karyawan (*Login Sebagai Terapis*)

3.6.11 Halaman Tambah Karyawan

Halaman ini merupakan halaman yang tampil saat *user* memilih menu tambah karyawan. Tambah karyawan merupakan sub-menu dari menu manajemen karyawan. Halaman ini berbentuk form, *user* akan memasukkan nama, durasi, harga, dan keterangan kemudian mengklik tombol submit untuk menyimpan data *treatment* baru. Desain antarmuka tambah *treatment* ditunjukkan pada Gambar 3.30.

HEADER	
Home Edit Profil Ubah Password Manajemen Order + Tambah Order Manajemen Pelanggan + Tambah Pelanggan Manajemen Treatment + Tambah Treatment Manajemen Karyawan + Tambah Karyawan Report - Report Presensi - Report Transaksi - Report Treatment	Tambah Karyawan Nama : Alamat : Tanggal lahir : Nomor telepon : Submit Reset
FOOTER	

Gambar 3.30 Halaman Tambah Karyawan

3.6.12 Halaman Manajemen Order

Halaman ini merupakan halaman yang tampil saat *user* memilih menu manajemen order. Halaman ini menampilkan nama-nama pelanggan yang melakukan order. Halaman ini memiliki *action*, yaitu bayar dan *delete*. Halaman ini dapat diakses oleh semua *user*. Desain antarmuka manajemen order(*login* sebagai manajer) ditunjukkan pada Gambar 3.31. Sedangkan desain antarmuka manajemen order (*login* sebagai terapis) ditunjukkan pada Gambar 3.32.

HEADER				
Home Edit Profil Ubah Password Manajemen Order + Tambah Order Manajemen Pelanggan + Tambah Pelanggan Manajemen Treatment + Tambah Treatment Manajemen Karyawan + Tambah Karyawan Report - Report Presensi - Report Transaksi - Report Treatment	Manajemen Order			
	No.	Nama	Action	
			Delete	Bayar
FOOTER				

Gambar 3.31 Halaman Manajemen Order(Login Sebagai Manager)

HEADER				
Home Edit Profil Ubah Password Manajemen Order + Tambah Order Lihat Pelanggan + Tambah Pelanggan Lihat Treatment Lihat Karyawan Report - Report Absensi	Manajemen Order			
	No.	Nama	Action	
			Delete	Bayar
FOOTER				

Gambar 3.32 Halaman Manajemen Order (Login Sebagai Terapis)

3.6.13 Halaman Tambah Order

Halaman ini merupakan halaman yang tampil saat *user* memilih menu tambah order. Tambah karyawan merupakan sub-menu dari menu manajemen order. Halaman ini berbentuk form pencarian, *user* akan memasukkan nama pelanggan, kemudian setelah muncul ditambahkan dengan mengklik *link* tambahkan ke manajemen order. Desain antarmuka tambah order (*login* sebagai manager)ditunjukkan pada Gambar 3.33. Sedangkan desain antarmuka tambah order (*login* sebagai terapis) ditunjukkan pada Gambar 3.34.

HEADER	
Home Edit Profil Ubah Password Manajemen Order + Tambah Order Manajemen Pelanggan + Tambah Pelanggan Manajemen Treatment + Tambah Treatment Manajemen Karyawan + Tambah Karyawan Report - Report Presensi - Report Transaksi - Report Treatment	Tambah Order
	Masukkan nama pelanggan : Search
FOOTER	

Gambar 3.33 Halaman Tambah Order(Login Sebagai Manager)

HEADER	
Home Edit Profil Ubah Password Manajemen Order + Tambah Order Lihat Pelanggan + Tambah Pelanggan Lihat Treatment Lihat Karyawan Report - Report Presensi	Tambah Order Masukkan nama pelanggan : Search
FOOTER	

Gambar 3.34 Halaman Tambah Order (Login Sebagai Terapis)

3.6.14 Halaman Report

Halaman ini merupakan halaman yang tampil saat *user* memilih menu report. Halaman ini menampilkan informasi tentang report bulanan salon yang meliputi jumlah pengunjung, jumlah pendapatan, dan *treatment* favorit. Menu report memiliki beberapa sub-menu yaitu report presensi, report *treatment*, dan report transaksi. Desain antarmuka halaman report (*login* sebagai manager) ditunjukkan pada Gambar 3.35. Desain antarmuka halaman report (*login* sebagai terapis) ditunjukkan pada Gambar 3.36.

HEADER	
Home Edit Profil Ubah Password Manajemen Order + Tambah Order Manajemen Pelanggan + Tambah Pelanggan Manajemen Treatment + Tambah Treatment Manajemen Karyawan + Tambah Karyawan Report - Report Presensi - Report Transaksi - Report Treatment	Report Jumlah pelanggan : Jumlah pendapatan : Treatment Favorit :
FOOTER	

Gambar 3.35 Halaman Report(Login Sebagai Manager)

HEADER	
Home Edit Profil Ubah Password Manajemen Order + Tambah Order Manajemen Pelanggan + Tambah Pelanggan Manajemen Treatment Manajemen Karyawan Report - Report Absensi	Report Jumlah pelanggan : Jumlah pendapatan : Treatment Favorit :
FOOTER	

Gambar 3.36 Halaman Report (Login Sebagai Terapis)

BAB IV

IMPLEMENTASI DAN PENGUJIAN SISTEM

4.1 Implementasi Dan Pengujian Sistem

Tahap ini akan membahas implementasi dan pengujian sistem. Berdasarkan pembahasan pada bab sebelumnya, disimpulkan ada dua *user* dalam sistem ini, yaitu manajer dan terapis. Setiap *user* memiliki hak akses yang berbeda. Manajer bisa mengakses semua menu, sedangkan terapis hanya beberapa menu saja.

Tahap implementasi merupakan tahap penerapan hasil perancangan sistem yang sudah dibuat. Implementasi ini dibuat sesederhana mungkin agar memudahkan *user* dalam penggunaannya dan kerja sistem yang dibuat sesuai dengan proses bisnis agar dapat dipahami. *Screenshot* setiap halaman merujuk pada perancangan desain antarmuka pada Bab III.

Tahap pengujian merupakan tahap menguji hasil Sistem Informasi Manajemen Qorinah Salon and Day Spa. Pengujian dibuat dengan tujuan memastikan sistem berjalan sesuai dengan rancangan. Selain itu, tahap pengujian dibuat untuk mengurangi kesalahan yang ada pada sistem.

Pengujian dilakukan menggunakan metode *black-box testing*. Aspek yang akan digunakan dalam pengujian sistem antara lain *existing data*, *input validation*, dan fungsionalitas. Setiap pengujian data memiliki tujuan masing – masing. Uji *existing data* bertujuan mengetahui bagaimana respon sistem jika dilakukan proses *query* terhadap data. Uji *input validation* bertujuan mengetahui respon sistem jika dilakukan proses input data berdasarkan kriteria ataupun tidak. Uji fungsionalitas bertujuan mengetahui bagaimana respon sistem terhadap akses pengguna pada sistem.

4.1.1 Halaman *Login*

Halaman *login* merupakan halaman yang pertama muncul ketika sistem mulai dijalankan. Halaman *login* ini digunakan untuk *login user*, seperti terapis dan

manajer, agar dapat mengakses sistem sesuai dengan jabatannya. Implementasi halaman login merujuk pada rancangan Gambar 3.14 yang terdapat pada Bab III. Implementasi halaman *login* ditunjukkan pada Gambar 4.1.



Gambar 4.1 Halaman *Login*

Apabila *user*terapis memasukkan *username* dan *password* dengan benar, maka akan muncul halaman utama untuk *user*terapis. Halaman utama terapis terdapat beberapa menu dan sub menu yang dapat diakses oleh terapis, yaitu *home*, ubah *password*, manajemen order, tambah order, manajemen pelanggan, *report*, dan *report* presensi. Implementasi halaman *home* terapis merujuk pada rancangan Gambar 3.15 yang terdapat pada Bab III, implementasi halaman *home* terapis ditunjukkan pada Gambar 4.2.



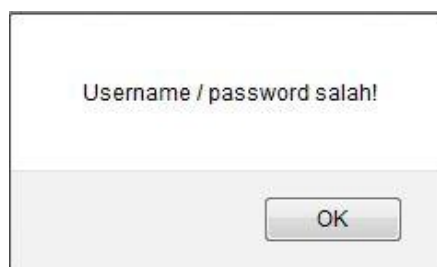
Gambar 4.2 Halaman *Home* Terapis

Berikut adalah halaman utama untuk *user* manajer. Halaman utama manajer semua menu dan sub menu dapat diakses, yaitu *home*, ubah *password*, manajemen order, tambah order, manajemen pelanggan, tambah pelanggan, manajemen treatment, tambah treatment, manajemen karyawan, tambah karyawan, report, report presensi, report treatment, dan report transaksi. Implementasi halaman *home* manajer merujuk pada rancangan Gambar 3.16 yang terdapat pada Bab III Implementasi halaman *home* manajer ditunjukkan pada Gambar 4.3



Gambar 4.3 Halaman *Home* Manajer

Proses *login* dilakukan dengan memasukkan *username* & *password* yang dimiliki oleh *user* ke dalam *form* yang terdapat pada halaman *login*. Apabila *user* memasukkan *username* dan *password* yang tidak sesuai dengan di database, akan muncul pesan *error* seperti pada Gambar 4.4



Gambar 4.4 Pesan Gagal *Login*

Berikut adalah hasil yang diperoleh dari pengujian *login* ditunjukkan pada Tabel 4.1

Tabel 4.1 Tabel Pengujian Login

Aspek	Input	Prediksi Output	Output Real
Input Validation	Username&Password (Benar)	Halaman utama user	Halaman utama user
Input Validation	Username&Password (Salah)	Muncul pesan kesalahan	Muncul pesan kesalahan

4.1.2 Halaman Edit Profil

Halaman ini merupakan halaman yang digunakan untuk mengedit profil *user* apabila diperlukan. *User* manajer dan terapis berhak mengakses halaman ini. Dalam *form* edit profil, ada beberapa *field* yang nantinya akan diisi oleh *user*. Ada dua tombol yang nantinya akan dipilih oleh *user*, yaitu Submit dan Reset. Dengan menekan tombol Submit maka data akan disimpan ke dalam *database*. Dengan menekan tombol Reset maka data pada *field* akan kembali seperti awal.

Implementasi halaman edit profil merujuk pada rancangan Gambar 3.17 yang terdapat pada Bab III. Implementasi halaman edit profil (*login* sebagai manajer) ditunjukkan pada rancangan Gambar 4.5. Sedangkan implementasi halaman edit profil (*login* sebagai terapis) merujuk pada rancangan Gambar 3.18 yang terdapat pada Bab III. Implementasi halaman edit profil (*login* sebagai terapis) ditunjukkan pada Gambar 4.6

The screenshot displays the 'Edit Profil' interface for a user logged in as a Manager. The page has a dark orange theme. On the left is a sidebar with navigation links: Home, Edit Profil, Ubah Password, Manajemen Order, + Tambah Order, Manajemen Pelanggan, + Tambah Pelanggan, Manajemen Treatment, + Tambah Treatment, Manajemen Karyawan, + Tambah Karyawan, Report, Report Presensi, Report Transaksi, and Report Treatment. The main content area is titled 'Edit Profil' and contains a form with the following fields: 'Nama' (filled with 'Danar Retno Sari'), 'Tgl lahir' (filled with '14 Maret 1992'), 'Alamat' (filled with '01. Uteap. Sumoharjo 301.9'), 'Nomor Telepon' (filled with '085643265888'), and 'Jabatan' (filled with 'Manager'). There is also a 'Foto' section with a 'Choose File' button and a 'No file chosen' label. At the bottom of the form are 'Submit' and 'Reset' buttons. The top of the page shows the system name 'Sistem Informasi Manajemen QORINAH DAY SPA & SALON' and a navigation bar with 'HOME | HELP | Danar Retno Sari, Anda login sebagai Manager | LOGOUT'.

Gambar 4.5 Halaman Edit Profil(Login Sebagai Manajer)

Gambar 4.6 Halaman Edit Profil (*Login Sebagai Terapis*)

Apabila terdapat data yang dimasukkan tidak lengkap pada *form* edit profil, atau tipe data yang diinputkan tidak sesuai, maka akan muncul pesan untuk memasukkan pemberitahuan kesalahan pada sistem. Pesan *error* ditunjukkan pada Gambar 4.7 dan pesan gagal edit profil ditunjukkan pada Gambar 4.8



Gambar 4.7 Pesan *Error* Edit Profil



Gambar 4.8 Pesan Gagal Edit Profil

Berikut adalah hasil yang diperoleh dari pengujian edit profil ditunjukkan pada Tabel 4.2

Tabel 4.2Tabel Pengujian Edit Profil

Aspek	Input	Prediksi Output	Output Real
<i>Input Validation</i>	<i>Field</i> tidak diisi	Muncul pesan data harus diisi lengkap	Muncul pesan data harus diisi lengkap
<i>Input Validation</i>	Data input benar	Menyimpan data	Menyimpan data
<i>Input Validation</i>	Data input salah	Muncul pesan gagal	Gagal menyimpan data
<i>Input Validation</i>	Penyimpanan data sukses	Muncul pesan data berhasil disimpan	Muncul pesan data berhasil disimpan
<i>Input Validation</i>	Penyimpanan data gagal	Muncul pesan data gagal disimpan	Muncul pesan data gagal disimpan

4.1.3 Halaman Ubah *Password*

Halaman ini merupakan halaman yang digunakan untuk merubah *username* dan *password* user. Setiap *user* berhak mengakses halaman ini. Dalam *form* ubah *password*, ada beberapa *field* yang nantinya akan diisi oleh *user*, yaitu *password* lama, *password* baru, dan konfirmasi *password* baru. Ada dua tombol yang nantinya akan dipilih oleh *user*, yaitu Submit dan Reset. Dengan menekan tombol Submit maka data akan disimpan ke dalam *database*. Dengan menekan tombol Reset maka data pada *field* akan kembali kosong.

Implementasi halaman ubah *password* merujuk pada rancangan Gambar 3.19 yang terdapat pada Bab III. Implementasi halaman ubah *password*(*login* sebagai manajer) ditunjukkan pada Gambar 4.9. Implementasi halaman ubah *password* merujuk pada rancangan Gambar 3.20 yang terdapat pada Bab III. Implementasi halaman ubah *password*(*login* sebagai terapis) ditunjukkan pada Gambar 4.9

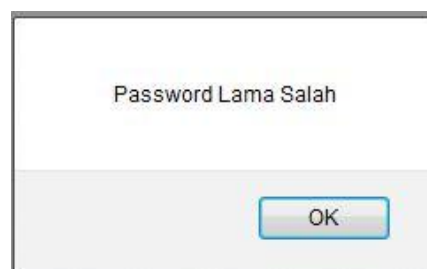
The screenshot shows the 'Ubah Password' (Change Password) page for a user logged in as a Manager. The page has a brown background with a subtle floral pattern. At the top, the title 'Sistem Informasi Manajemen QORINAH' is displayed, with 'QORINAH' in a stylized font and 'DAY SPA & SALON' below it. A navigation bar at the top contains links: HOME | HELP | Danar Retno Sari, Anda login sebagai: Manager | LOGOUT. On the left, there is a vertical menu with options: Home, Edit Profil, Ubah Password, Manajemen Order, + Tambah Order, Manajemen Pelanggan, + Tambah Pelanggan, Manajemen Treatment, + Tambah Treatment, Manajemen Karyawan, + Tambah Karyawan, Report, Report Presensi, Report Transaksi, and Report Treatment. The main content area is titled 'Ubah Password' and contains three input fields: 'Password lama', 'Password baru', and 'Konfirmasi Password baru'. Below these fields are two buttons: 'Submit' and 'Reset'.

Gambar 4.9 Halaman Ubah *Password*(Login Sebagai Manager)

The screenshot shows the 'Ubah Password' (Change Password) page for a user logged in as a Therapist. The layout is similar to the Manager page, with the same title and navigation bar. The navigation bar at the top contains links: HOME | HELP | Rabiaini Dwi Citra, Anda login sebagai: Terapis | LOGOUT. The left menu is also similar but includes 'Lihat Pelanggan' and 'Lihat Karyawan' instead of '+ Tambah'. The main content area is titled 'Ubah Password' and contains three input fields: 'Password lama', 'Password baru', and 'Konfirmasi Password baru'. Below these fields are two buttons: 'Submit' and 'Reset'.

Gambar 4.10 Halaman Ubah *Password*(Login Sebagai Terapis)

Apabila terdapat data yang belum dimasukkan pada masing-masing *field* pada *form* ubah *password*, atau data yang diinputkan tidak sesuai, maka akan muncul pesan untuk memasukkan pemberitahuan kesalahan pada sistem. Pesan *error* ditunjukkan pada Gambar 4.11, 4.12, 4.13



Gambar 4.11 Pesan *Error* Ubah *Password*



Gambar 4.12 Pesan *Error* Ubah *Password*



Gambar 4.13 Pesan *Error* Ubah *Password*

Berikut adalah hasil yang diperoleh dari pengujian ubah *password* ditunjukkan pada Tabel 4.3

Tabel 4.3Tabel Pengujian Ubah *Password*

Aspek	Input	Prediksi Output	Output Real
<i>Input Validation</i>	<i>Field</i> tidak diisi	Muncul pesan input tidak boleh kosong	Muncul pesan input tidak boleh kosong
<i>Input Validation</i>	Data input benar	Menyimpan data	Menyimpan data
<i>Input Validation</i>	Data input salah	Muncul pesan gagal	Gagal menyimpan data
<i>Input Validation</i>	Penyimpanan data sukses	Muncul pesan data berhasil disimpan	Muncul pesan data berhasil disimpan
<i>Input Validation</i>	Penyimpanan data gagal	Muncul pesan data gagal disimpan	Muncul pesan data gagal disimpan
<i>Input Validation</i>	Data tidak sesuai	Muncul pesan data tidak sesuai / salah	Muncul pesan data tidak sesuai / salah

4.1.4 Halaman Manajemen Order

Halaman ini merupakan halaman yang digunakan untuk menambah order dan melihat pelanggan yang sudah masuk dalam list order. Halaman ini dapat diakses oleh semua *user*, yaitu terapis dan manajer.

Implementasi halaman manajemen order (*login* sebagai manajer) merujuk pada rancangan Gambar 3.31 yang terdapat pada Bab III. Implementasi halaman

manajemen order (*login* sebagai manajer) ditunjukkan pada Gambar 4.14. Implementasi halaman manajemen order (*login* sebagai terapis) merujuk pada rancangan Gambar 3.32 yang terdapat pada Bab III. Implementasi halaman manajemen order (*login* sebagai terapis) ditunjukkan pada Gambar 4.15



Gambar 4.14 Halaman Manajemen Order (*Login Sebagai Manajer*)



Gambar 4.15 Halaman Manajemen Order (*Login Sebagai Terapis*)

Halaman ini merupakan halaman yang digunakan untuk menambahkan order pelanggan. Proses dimulai dengan mencari pelanggan lalu ditambahkan ke manajemen order. Halaman ini dapat diakses oleh semua *user*, yaitu terapis dan manajer.

Implementasi halaman tambah order (*login* sebagai manajer) merujuk pada rancangan Gambar 3.33 yang terdapat pada Bab III. Implementasi halaman

tambah order (*login* sebagai manajer)ditunjukkan pada Gambar 4.16. Implementasi halaman tambah order (*login* sebagai terapis) merujuk pada rancangan Gambar 3.34 yang terdapat pada Bab III. Implementasi halaman tambah order (*login* sebagai terapis) ditunjukkan pada Gambar 4.17.

Gambar 4.16Halaman Tambah Order(*Login* Sebagai Manajer)

Gambar 4.17 Halaman Tambah Order (*Login* Sebagai Terapis)

Berikut adalah hasil yang diperoleh dari pengujian tambah order ditunjukkan pada Tabel 4.4

Tabel 4.4 Tabel Pengujian Tambah Order

Aspek	Input	Prediksi Output	Output Real
<i>Fungsional</i>	Klik Tambah Order	Halaman Tambah Order	Halaman Tambah Order
<i>Existing data</i>	Masukkan nama pelanggan	Halaman nama pelanggan	Halaman nama pelanggan

Setelah menambahkan order pelanggan, akan muncul halaman tambah order treatment, dimana di halaman tersebut akan ditambahkan treatment yang akan diorder oleh pelanggan. Proses dimulai dengan mencari treatment lalu ditambahkan ke order. Halaman ini dapat diakses oleh semua *user*, yaitu terapis dan manajer. Implementasi halaman tambah order treatment (*login* sebagai manajer) ditunjukkan pada Gambar 4.18. Implementasi halaman tambah order treatment (*login* sebagai terapis) ditunjukkan pada Gambar 4.19.

Sistem Informasi Manajemen QORINAH DAY SPA & SALON

HOME | HELP | Danar Retno Sari, Anda login sebagai Manajer | LOGOUT

Tambah Order Treatment

Masukkan nama treatment: by name

No	Nama Treatment	Harga	Action
1	Total Aromatherapy Foot Spa	90,000	Tambahkan ke order
2	Half Aromatherapy Foot Spa	75,000	Tambahkan ke order
3	Half Aromatherapy Hand & Foot Spa	125,000	Tambahkan ke order
4	Creambath Traditional & Total Foot Spa	120,000	Tambahkan ke order
5	Hair Spa & Half Foot Spa	130,000	Tambahkan ke order

1 2 3 4 5 6 7 8 9 10 Next >

HOME | HELP | LOGOUT | Danar Retno Sari

Gambar 4.18 Halaman Tambah Order Treatment (*Login Sebagai Manajer*)

Sistem Informasi Manajemen QORINAH DAY SPA & SALON

HOME | HELP | Mima Masitah, Anda login sebagai Terapis | LOGOUT

Tambah Order Treatment

Masukkan nama treatment: by name

No	Nama Treatment	Harga	Action
1	Total Aromatherapy Foot Spa	90,000	Tambahkan ke order
2	Total Aromatherapy Hand Spa	82,500	Tambahkan ke order
3	Creambath Traditional & Total Foot Spa	120,000	Tambahkan ke order
4	Hair Mask Makarizo & Total Hand Spa	125,000	Tambahkan ke order

1 2 3 4 5 6 7 8 9 10 Next >

HOME | HELP | LOGOUT | Mima Masitah

Gambar 4.19 Halaman Tambah Order Treatment (*Login Sebagai Terapis*)

Berikut adalah hasil yang diperoleh dari pengujian tambah order treatment ditunjukkan pada Tabel 4.5

Tabel 4.5 Pengujian Tambah Order Treatment

Aspek	Input	Prediksi Output	Output Real
<i>Fungsional</i>	Klik tombol Tambah Treatment	Halaman Tambah Order Treatment	Halaman Tambah Order Treatment
<i>Existing Data</i>	Masukkan nama treatment	Keluar nama treatment	Keluar nama treatment

4.1.5 Halaman Manajemen Pelanggan

Halaman ini merupakan halaman yang digunakan untuk melihat pelanggan yang dimasukkan ke dalam *database* oleh *user*. Halaman ini menampilkan nama pelanggan dan *action*, yaitu detail, edit, dan hapus. Halaman ini dapat diakses oleh semua *user*, yaitu terapis dan manajer.

Implementasi halaman manajemen pelanggan (*login* sebagai manajer) merujuk pada rancangan Gambar 3.21 yang terdapat pada Bab III. Implementasi halaman manajemen pelanggan (*login* sebagai manajer) ditunjukkan pada Gambar 4.20. Implementasi halaman tambah order (*login* sebagai terapis) merujuk pada rancangan Gambar 3.22 yang terdapat pada Bab III. Implementasi halaman lihat pelanggan (*login* sebagai terapis) ditunjukkan pada gambar 4.21



Gambar 4.20 Halaman Manajemen Pelanggan (*Login* Sebagai Manajer)

Sistem Informasi Manajemen QORINAH
DAY SPA & SALON

HOME | HELP | Rabiaini Dwi Citra, Anda login sebagai Terapis | LOGOUT

Manajemen Pelanggan

Masukkan nama/id pelanggan : by nama ▼

No	Id	Nama	Action
1	4	Rina	detail edit
2	5	Rifah Tri Citra	detail edit
3	6	Eka Citra	detail edit
4	7	Putri	detail edit

HOME | HELP | Rabiaini Dwi Citra | LOGOUT

Gambar 4.21 Halaman Lihat Pelanggan (*Login Sebagai Terapis*)

Selanjutnya adalah halaman tambah pelanggan. Halaman ini merupakan halaman yang digunakan untuk menambahkan data pelanggan baru untuk dimasukkan ke dalam *database*. Halaman ini dapat diakses oleh semua *user*, yaitu terapis dan manajer. Implementasi halaman tambah pelanggan (*login sebagai manajer*) ditunjukkan pada Gambar 4.22. Implementasi halaman tambah pelanggan (*login sebagai terapis*) ditunjukkan pada Gambar 4.23.

Sistem Informasi Manajemen QORINAH
DAY SPA & SALON

HOME | HELP | Danar Retno Sari, Anda login sebagai Manajer | LOGOUT

Tambah Pelanggan

Tanggal Lahir :

Alamat :

Nomor Telepon :

Email :

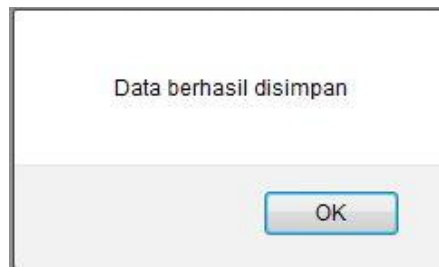
Status :

HOME | HELP | Danar Retno Sari | LOGOUT

Gambar 4.22 Halaman Tambah Pelanggan (*Login Sebagai Manajer*)

Gambar 4.23Halaman Tambah Pelanggan (*Login Sebagai Terapis*)

Apabila terdapat data yang belum dimasukkan pada masing-masing *field* pada *form* tambah pelanggan, maka akan muncul pesan untuk memasukkan pemberitahuan kesalahan pada sistem. Pesan *error* ditunjukkan pada Gambar 4.25, 4.26. Apabila *user* memasukkan data yang sesuai dan semua *field* diisi, maka akan muncul pesan bahwa data berhasil disimpan yang ditunjukkan pada Gambar 4.24.



Gambar 4.24 Pesan Berhasil Menyimpan Data



Gambar 4.25 Pesan *Error* Gagal Menyimpan Data



Gambar 4.26 Pesan *Error* Data tidak Lengkap

Berikut adalah hasil yang diperoleh dari pengujian tambah pelanggan ditunjukkan pada Tabel 4.6

Tabel 4.6 Tabel Pengujian Tambah Pelanggan

Aspek	Input	Prediksi Output	Output Real
<i>Fungsional</i>	Klik tombol Tambah Pelanggan	Halaman Tambah Pelanggan	Halaman Tambah Pelanggan
<i>Input Validation</i>	<i>Field</i> tidak diisi	Muncul pesan data harus diisi lengkap	Muncul pesan data harus diisi lengkap
<i>Input Validation</i>	Data input benar	Menyimpan data	Menyimpan data
<i>Input Validation</i>	Data input salah	Muncul pesan gagal	Gagal menyimpan data
<i>Fungsional</i>	Penyimpanan data sukses	Muncul pesan data berhasil disimpan	Muncul pesan data berhasil disimpan
<i>Fungsional</i>	Penyimpanan data gagal	Muncul pesan data gagal disimpan	Muncul pesan data gagal disimpan

Selanjutnya adalah halaman edit pelanggan. Halaman ini merupakan halaman yang digunakan untuk mengedit data pelanggan apabila diperlukan. Halaman ini dapat diakses oleh semua *user*, yaitu terapis dan manajer. Implementasi halaman edit pelanggan (*login* sebagai manajer) ditunjukkan pada Gambar 4.27. Implementasi halaman edit pelanggan (*login* sebagai terapis) ditunjukkan pada Gambar 4.27

Sistem Informasi Manajemen QORINAH
DAY SPA & SALON

HOME | HELP | Danar Retno Sani, Anda login sebagai Manager | LOGOUT

Edit pelanggan:

(*) **Wajib diisi**

Nama : Rina

Tgl lahir : 4 Maret 1983

Alamat : Jaka1

Nomor Telepon : 07070707

Email : rina@yahoo.com

Status : Pilih Status

Submit Reset

HOME | HELP | Danar Retno Sani | LOGOUT

Gambar 4.27 Halaman Edit Pelanggan (*Login Sebagai Manager*)

Sistem Informasi Manajemen QORINAH
DAY SPA & SALON

HOME | HELP | Rabiaini Dwi Citra, Anda login sebagai Terapis | LOGOUT

Edit pelanggan:

(*) **Wajib diisi**

Nama : Rina

Tgl lahir : 4 Maret 1983

Alamat : Jaka1

Nomor Telepon : 07070707

Email : rina@yahoo.com

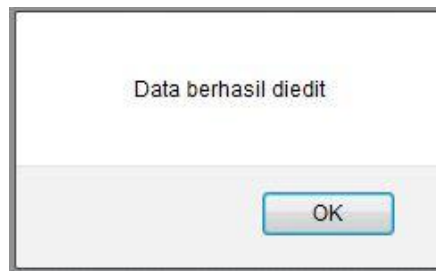
Status : Pilih Status

Submit Reset

HOME | HELP | Rabiaini Dwi Citra | LOGOUT

Gambar 4.28 Halaman Edit Pelanggan (*Login Sebagai Terapis*)

Apabila terdapat data yang belum dimasukkan pada masing-masing *field* pada *form* edit pelanggan, maka akan muncul pesan untuk memasukkan pemberitahuan kesalahan pada sistem. Pesan *error* ditunjukkan pada Gambar 4.30, 4.31. Apabila *user* memasukkan data yang sesuai dan semua *field* diisi, maka akan muncul pesan bahwa data berhasil disimpan yang ditunjukkan pada Gambar 4.29.



Gambar 4.29 Pesan Berhasil Mengedit Data



Gambar 4.30 Pesan *Error* Gagal Menyimpan Data



Gambar 4.31 Pesan *Error* Data tidak Lengkap

Berikut adalah hasil yang diperoleh dari pengujian edit pelanggan ditunjukkan pada Tabel 4.7

Tabel 4.7 Tabel Pengujian Edit Pelanggan

Aspek	Input	Prediksi Output	Output Real
<i>Fungsional</i>	Klik link Edit Pelanggan	Halaman Edit Pelanggan	Halaman Edit Pelanggan
<i>Input Validation</i>	Edit data input benar	Menyimpan data	Menyimpan data
<i>Input Validation</i>	Edit data input salah	Muncul pesan gagal	Gagal menyimpan data
<i>Fungsional</i>	Proses edit data sukses	Muncul pesan data berhasil diedit	Muncul pesan data berhasil diedit
<i>Fungsional</i>	Proses edit data gagal	Muncul pesan data gagal disimpan	Muncul pesan data gagal disimpan

4.1.6 Halaman Manajemen Treatment

Halaman ini merupakan halaman yang digunakan untuk melihat berbagai macam treatment yang dimasukkan ke dalam *database* oleh *user*. Halaman ini dapat diakses oleh manajer dan terapis. Halaman ini menampilkan informasi nama treatment dan *action*, yaitu detail, edit, dan hapus. Implementasi halaman manajemen treatment (*login* sebagai manajer) merujuk pada rancangan Gambar 3.25 yang terdapat pada Bab III. Implementasi halaman manajemen treatment (*login* sebagai manajer) ditunjukkan pada Gambar 4.32. Implementasi halaman manajemen treatment (*login* sebagai terapis) merujuk pada rancangan Gambar 3.26 yang terdapat pada Bab III. Implementasi halaman lihat treatment (*login* sebagai terapis) ditunjukkan pada Gambar 4.33.



Gambar 4.32 Halaman Manajemen Treatment (Login Sebagai Manajer)



Gambar 4.33 Halaman Manajemen Treatment (Login Sebagai Terapis)

Selanjutnya adalah halaman tambah treatment. Halaman ini merupakan halaman yang digunakan untuk menambahkan data treatment baru ke dalam *database* oleh *user*. Halaman ini dapat diakses oleh manajer dan terapis.

Implementasi halaman tambah treatment (*login* sebagai manajer) merujuk pada rancangan Gambar 3.27 yang terdapat pada Bab III. Implementasi halaman tambah treatment (*login* sebagai manajer) ditunjukkan pada gambar 4.34.

The screenshot shows a web application interface for 'Sistem Informasi Manajemen QORINAH DAY SPA & SALON'. The top navigation bar includes links for HOME, HELP, and a user status 'Danar Retno Sari, Anda login sebagai: Manajer', along with a LOGOUT button. A left sidebar menu lists various management functions: Home, Edit Profil, Ubah Password, Manajemen Order (with a sub-option '+ Tambah Order'), Manajemen Pelanggan (with sub-options '+ Tambah Pelanggan' and '+ Tambah Treatment'), Manajemen Treatment (with a sub-option '+ Tambah Treatment'), Manajemen Karyawan (with a sub-option '+ Tambah Karyawan'), and a Report section (with sub-options 'Report Presensi', 'Report Transaksi', and 'Report Treatment'). The main content area is titled 'Tambah Treatment' and contains a form with the following fields: 'Nama Treatment:' (text input), 'Jenis treatment:' (dropdown menu showing 'Aia Carte Body Treatment'), 'Durasi:' (two dropdown menus for '0 jam' and '0 menit'), 'Harga:' (text input), and 'Keterangan:' (text area). At the bottom of the form are 'Submit' and 'Reset' buttons. The footer of the page repeats the HOME, HELP, user status, and LOGOUT links.

Gambar 4.34 Halaman Tambah Treatment(*Login* Sebagai Manajer)

Apabila terdapat data yang belum dimasukkan pada masing-masing *field* pada *form* tambah treatment, maka akan muncul pesan untuk memasukkan pemberitahuan kesalahan pada sistem. Pesan *error* ditunjukkan pada Gambar 4.36, 4.37. Apabila *user* memasukkan data yang sesuai dan semua *field* diisi, maka akan muncul pesan bahwa data berhasil disimpan yang ditunjukkan pada Gambar 4.35.



Gambar 4.35 Pesan Berhasil Menyimpan Data



Gambar 4.36 Pesan *Error* Gagal Menyimpan Data



Gambar 4.37 Pesan *Error* Data tidak Lengkap

Berikut adalah hasil yang diperoleh dari pengujian tambah treatment ditunjukkan pada Tabel 4.8

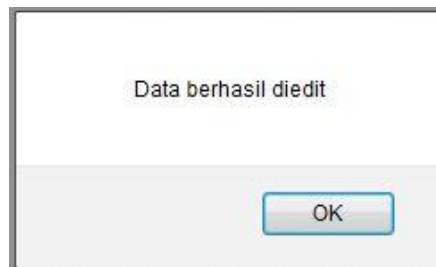
Tabel 4.8 Tabel Pengujian Tambah Treatment

Aspek	Input	Prediksi Output	Output Real
<i>Fungsional</i>	Klik tombol Tambah Treatment	Halaman Tambah Treatment	Halaman Tambah Treatment
<i>Input Validation</i>	<i>Field</i> tidak diisi	Muncul pesan data harus diisi lengkap	Muncul pesan data harus diisi lengkap
<i>Input Validation</i>	Data input benar	Menyimpan data	Menyimpan data
<i>Input Validation</i>	Data input salah	Muncul pesan gagal	Gagal menyimpan data
<i>Fungsional</i>	Penyimpanan data sukses	Muncul pesan data berhasil disimpan	Muncul pesan data berhasil disimpan
<i>Fungsional</i>	Penyimpanan data gagal	Muncul pesan data gagal disimpan	Muncul pesan data gagal disimpan

Selanjutnya adalah halaman edit treatment. Halaman ini merupakan halaman yang digunakan untuk mengedit data treatment apabila diperlukan. Halaman ini dapat diakses oleh manajer. Implementasi halaman edit pelanggan (*login* sebagai manajer) ditunjukkan pada Gambar 4.38.

Gambar 4.38 Halaman Edit Treatment (*Login Sebagai Manajer*)

Apabila terdapat data yang belum dimasukkan pada masing-masing *field* pada *form* edit treatment, maka akan muncul pesan untuk memasukkan pemberitahuan kesalahan pada sistem. Pesan *error* ditunjukkan pada Gambar 4.40, 4.41. Apabila *user* memasukkan data yang sesuai dan semua *field* diisi, maka akan muncul pesan bahwa data berhasil disimpan yang ditunjukkan pada Gambar 4.39.



Gambar 4.39 Pesan Berhasil Mengedit Data



Gambar 4.40 Pesan *Error* Gagal Menyimpan Data



Gambar 4.41 Pesan *Error* Data tidak Lengkap

Berikut adalah hasil yang diperoleh dari pengujian edit treatment ditunjukkan pada Tabel 4.9

Tabel 4.9 Tabel Pengujian Edit Pelanggan

Aspek	Input	Prediksi Output	Output Real
<i>Fungsional</i>	Klik link Edit Treatment	Halaman Edit Treatment	Halaman Edit Treatment
<i>Input Validation</i>	Edit data input benar	Menyimpan data	Menyimpan data
<i>Input Validation</i>	Edit data input salah	Muncul pesan gagal	Gagal menyimpan data
<i>Fungsional</i>	Proses edit data sukses	Muncul pesan data berhasil diedit	Muncul pesan data berhasil diedit
<i>Fungsional</i>	Proses edit data gagal	Muncul pesan data gagal disimpan	Muncul pesan data gagal disimpan

4.1.7 Halaman Manajemen Karyawan

Halaman ini merupakan halaman yang digunakan untuk melihat karyawan yang dimasukkan ke dalam *database* oleh *user*. Halaman ini dapat diakses oleh manajer dan terapis.

Implementasi halaman manajemen karyawan (*login* sebagai manajer) merujuk pada rancangan Gambar 3.28 yang terdapat pada Bab III. Implementasi halaman manajemen karyawan (*login* sebagai manajer) ditunjukkan pada Gambar 4.42. Implementasi halaman manajemen karyawan (*login* sebagai terapis) merujuk pada rancangan Gambar 3.29 yang terdapat pada Bab III. Implementasi halaman lihat karyawan (*login* sebagai terapis) ditunjukkan pada Gambar 4.43.



Gambar 4.42 Halaman Manajemen Karyawan (*Login Sebagai Manajer*)



Gambar 4.43 Halaman Lihat Karyawan (*Login Sebagai Terapis*)

Selanjutnya adalah halaman tambah karyawan. Halaman ini merupakan halaman yang digunakan untuk menambahkan data karyawan ke dalam *database* oleh *user*. Halaman ini dapat diakses oleh manajer saja.

Implementasi halaman tambah karyawan (*login* sebagai manajer) merujuk pada rancangan Gambar 3.30 yang terdapat pada Bab III. Implementasi halaman tambah karyawan ditunjukkan pada gambar 4.44.

Gambar 4.44Halaman Tambah Karyawan

Apabila terdapat data yang belum dimasukkan pada masing-masing *field* pada *form* tambah karyawan, maka akan muncul pesan untuk memasukkan pemberitahuan kesalahan pada sistem. Pesan *error* ditunjukkan pada Gambar 4.46, 4.47. Apabila *user* memasukkan data yang sesuai dan semua *field* diisi, maka akan muncul pesan bahwa data berhasil disimpan yang ditunjukkan pada Gambar 4.45.



Gambar 4.45 Pesan Berhasil Menyimpan Data



Gambar 4.46 Pesan *Error* Gagal Menyimpan Data



Gambar 4.47 Pesan *Error* Data tidak Lengkap

Berikut adalah hasil yang diperoleh dari pengujian tambah karyawan ditunjukkan pada Tabel 4.10

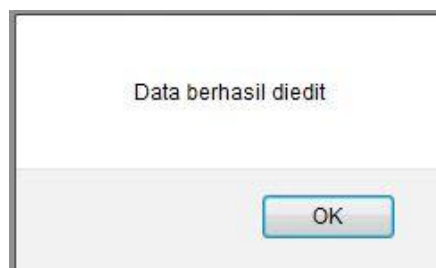
Tabel 4.10 Tabel Pengujian Tambah Karyawan

Aspek	Input	Prediksi Output	Output Real
<i>Fungsional</i>	Klik tombol Tambah karyawan	Halaman Tambah karyawan	Halaman Tambah Karyawan
<i>Input Validation</i>	<i>Field</i> tidak diisi	Muncul pesan data harus diisi lengkap	Muncul pesan data harus diisi lengkap
<i>Input Validation</i>	Data input benar	Menyimpan data	Menyimpan data
<i>Input Validation</i>	Data input salah	Muncul pesan gagal	Gagal menyimpan data
<i>Fungsional</i>	Penyimpanan data sukses	Muncul pesan data berhasil disimpan	Muncul pesan data berhasil disimpan
<i>Fungsional</i>	Penyimpanan data gagal	Muncul pesan data gagal disimpan	Muncul pesan data gagal disimpan

Selanjutnya adalah halaman edit karyawan. Halaman ini merupakan halaman yang digunakan untuk mengedit data karyawan untuk dimasukkan ke dalam *database*. Halaman ini dapat diakses oleh manajer. Implementasi halaman edit pelanggan ditunjukkan pada Gambar 4.48.

Gambar 4.48 Halaman Edit Karyawan

Apabila terdapat data yang belum dimasukkan pada masing-masing *field* pada *form* edit karyawan, maka akan muncul pesan untuk memasukkan pemberitahuan kesalahan pada sistem. Pesan *error* ditunjukkan pada Gambar 4.50, 4.51. Apabila *user* memasukkan data yang sesuai dan semua *field* diisi, maka akan muncul pesan bahwa data berhasil disimpan yang ditunjukkan pada Gambar 4.49.



Gambar 4.49 Pesan Berhasil Mengedit Data



Gambar 4.50 Pesan *Error* Gagal Menyimpan Data



Gambar 4.51 Pesan *Error* Data tidak Lengkap

Berikut adalah hasil yang diperoleh dari pengujian edit karyawan ditunjukkan pada Tabel 4.11

Tabel 4.11 Tabel Pengujian Edit Karyawan

Aspek	Input	Prediksi Output	Output Real
<i>Fungsional</i>	Klik link Edit Karyawan	Halaman Edit Karyawan	Halaman Edit Karyawan
<i>Input Validation</i>	Edit data input benar	Menyimpan data	Menyimpan data
<i>Input Validation</i>	Edit data input salah	Muncul pesan gagal	Gagal menyimpan data
<i>Fungsional</i>	Proses edit data sukses	Muncul pesan data berhasil diedit	Muncul pesan data berhasil diedit
<i>Fungsional</i>	Proses edit data gagal	Muncul pesan data gagal disimpan	Muncul pesan data gagal disimpan

4.1.8 Halaman Report

Halaman ini merupakan halaman yang menunjukkan data akumulasi tentang jumlah pelanggan, jumlah pendapatan, jumlah jam, dan treatment favorit. Halaman ini dapat diakses oleh semua *user*, yaitu terapis dan manajer. Implementasi halaman report (*login* sebagai manajer) merujuk pada rancangan Gambar 3.35 yang terdapat pada Bab III. Implementasi halaman report (*login* sebagai manajer) ditunjukkan pada Gambar 4.52. Implementasi halaman report (*login* sebagai terapis) merujuk pada rancangan Gambar 3.36 yang terdapat pada Bab III. Implementasi halaman report (*login* sebagai terapis) ditunjukkan pada Gambar 4.53.



Gambar 4.52 Halaman Report (*Login Sebagai Manajer*)



Gambar 4.53 Halaman Report (*Login Sebagai Terapis*)

Berikut adalah hasil yang diperoleh dari pengujian report ditunjukkan pada Tabel 4.12

Tabel 4.12 Pengujian Report

Aspek	Input	Prediksi Output	Output Real
<i>Fungsional</i>	Klik Report	Halaman Report	Halaman Report
<i>Fungsional</i>	Klik Print Report	Halaman Print Report	Halaman Print Report

4.1.9 Halaman Report Presensi

Halaman ini merupakan halaman yang menunjukkan data-data yang berhubungan dengan presensi karyawan di Qorinah Salon and Day Spa. Informasi

yang ditampilkan antara lain nama karyawan, jumlah hadir, dan jumlah tidak hadirnya karyawan, persentase kehadiran, dan jumlah jam kerja. Selain itu user dapat mencetak report. Halaman ini dapat diakses oleh semua user. Implementasi halaman report presensi ditunjukkan pada Gambar 4.54.

No.	Nama	Hadir	Tidak Hadir	Persentase	Jumlah Jam Kerja	Detail
1	Danar Retno Sari	0	30	0 %	hari jam menit	Detail
2	Rabaini Dwi Citra	0	30	0 %	hari jam menit	Detail
3	Rifah Tri Citra	0	30	0 %	hari jam menit	Detail
4	Mima Masitah	0	30	0 %	hari jam menit	Detail
5	Rianti	0	30	0 %	hari jam menit	Detail

Gambar 4.54 Halaman Report Presensi

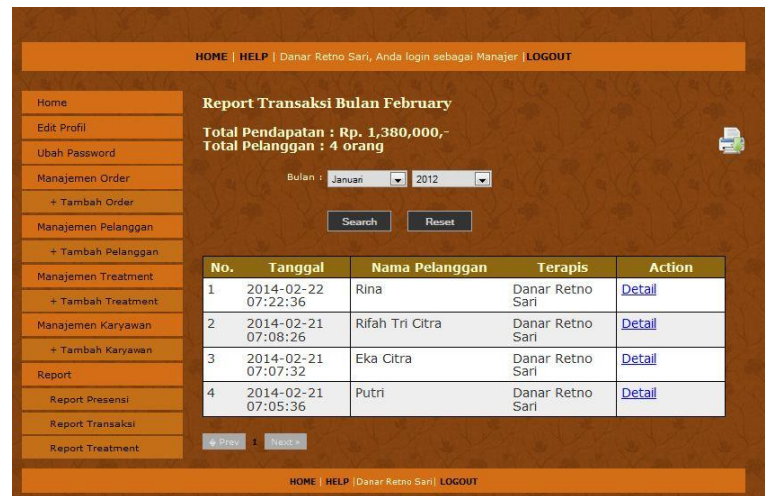
Berikut adalah hasil yang diperoleh dari pengujian report presensi ditunjukkan pada Tabel 4.13

Tabel 4.13 Pengujian Report Presensi

Aspek	Input	Prediksi Output	Output Real
<i>Fungsional</i>	Klik Report Presensi	Halaman Report Presensi	Halaman Report Presensi
<i>Existing Data</i>	Tanggal Report Presensi	Rekap Report Presensi	Rekap Report Presensi
<i>Fungsional</i>	Klik Print Report Presensi	Halaman Print Report Presensi	Halaman Print Report Presensi

4.1.10 Halaman Report Transaksi

Halaman ini merupakan halaman yang menunjukkan data-data yang berhubungan dengan transaksi di Qorinah Salon and Day Spa. Halaman ini dapat diakses oleh manajer saja. Implementasi halaman report transaksi ditunjukkan pada Gambar 4.55.



Gambar 4.55 Halaman Report Transaksi

Berikut adalah hasil yang diperoleh dari pengujian report transaksi ditunjukkan pada Tabel 4.14

Tabel 4.14 Pengujian Report Transaksi

Aspek	Input	Prediksi Output	Output Real
<i>Fungsional</i>	Klik Report Transaksi	Halaman Report Transaksi	Halaman Report Transaksi
<i>Existing Data</i>	Tanggal Report Transaksi	Rekap Report Transaksi	Rekap Report Transaksi
<i>Fungsional</i>	Klik Print Report Transaksi	Halaman Print Report Transaksi	Halaman Print Report Transaksi

4.1.11 Halaman Report Treatment

Halaman ini merupakan halaman yang menunjukkan data-data yang berhubungan dengan history treatment di Qorinah Salon and Day Spa. Halaman ini hanya dapat diakses oleh manajer saja. Implementasi halaman report treatment ditunjukkan pada Gambar 4.56

HOME | HELP | Danar Retno Sari, Anda login sebagai Manajer | LOGOUT

Home
Edit Profil
Ubah Password
Manajemen Order
+ Tambah Order
Manajemen Pelanggan
+ Tambah Pelanggan
Manajemen Treatment
+ Tambah Treatment
Manajemen Karyawan
+ Tambah Karyawan
Report
Report Presensi
Report Transaksi
Report Treatment

Report Treatment Bulan February

Treatment favourite : Javanese Body Whitening
Total jam treatment : 16 jam 00 menit

Bulan Januari 2012

Search Reset

No.	Nama Treatment
1	Javanese Body Whitening
2	Facial Biocell Algae (Rejuvenating)
3	Fruit For Antioxidan
4	Olive Body Release
5	Javanese Body Whitening

HOME | HELP | Danar Retno Sari | LOGOUT

Gambar 4.56 Halaman Report Treatment

Berikut adalah hasil yang diperoleh dari pengujian report treatment ditunjukkan pada Tabel 4.15

Tabel 4.15 Pengujian Report Treatment

Aspek	Input	Prediksi Output	Output Real
<i>Fungsional</i>	Klik Report	Halaman Report treatment	Halaman Report Treatment
<i>Existing Data</i>	Tanggal Report	Rekap Report Treatment	Rekap Report Treatment
<i>Fungsional</i>	Klik Print Report	Halaman Print Report Treatment	Halaman Print Report Treatment

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Dari serangkaian Bab I sampai pada Bab IV maka dapat disimpulkan tentang Sistem Informasi Manajemen Qorinah Day Spa and Salon, sebagai berikut:

1. Dari hasil perancangan dan pembangunan sistem informasi manajemen Qorinah Day Spa and Salon, maka diciptakan sistem yang dapat membantu kinerja karyawan di Qorinah Day Spa and Salon dalam memanajemen data pelanggan, data order, data karyawan, data presensi, dan data treatment.
2. Dalam sistem ini juga terdapat report yang digunakan dalam pemantauan presensi dan transaksi sehingga proses evaluasi dapat lebih maksimal.
3. Sistem ini dibuat sebagai pemanfaatan teknologi informasi di Qorinah Day Spa and Salon.

5.2 Saran

Hasil dari sistem ini masih belum sempurna, berikut adalah beberapa saran untuk langkah selanjutnya, yaitu :

1. Sistem informasi manajemen ini dapat dikembangkan lagi sesuai dengan kebutuhan lengkap Qorinah Day Spa and Salon yang dapat membuat sistem lebih optimal, seperti perhitungan poin pelanggan dan perhitungan gaji karyawan.
2. Sistem informasi manajemen ini perlu memasukkan perspektif pelanggan dan pengguna dalam analisis kebutuhan sistem.

DAFTAR PUSTAKA

- Kadir, Abdul. (2003). *Pengenalan Sistem Informasi*. Yogyakarta : Andi.
- Pressman, Roger S. (2001). *Software Engineering : A Practitioner's Approach - 5th ed*. New York : McGraw Hill Companies Inc.
- Fowler, Martin. (2003). *UML Distilled : A Brief Guide to the Standard Object Modelling Language (3rd edition)*. New York : Pearson Education.
- Connolly, Thomas and Begg, Carolyn. (2004). *Database Systems: A Practical Approach to Design, Implementation, and Management, Fourth Edition*. USA : Addison-Wesley Publishing Company.
- Welling, Luke, and Laura Thomson. (2001). *PHP and MySQL Web Development*. Indianapolis : Sams Publishing.
- Nugroho, Bunafit. (2008). *Membuat Sistem Penjualan Berbasis Web dengan PHP dan MySQL*. Yogyakarta : Gava Media.
- Sutabri, Tata. (2012). *Analisis Sistem Informasi*. Yogyakarta : Andi.
- MySQL. (1997). *Data Type Storage Requirement*. <http://dev.mysql.com/doc/refman/5.0/en/storage-requirements.html>. Diakses pada 8 Oktober 2013.
- Normalisasi. (2013). *Pengertian Normalisasi Database dan Contoh Normalisasi*. <http://jewyner.wordpress.com/2013/03/05/92/>. Diakses pada 26 Januari 2014
- Febrian, Jack, dan Farida Andayani. (2002). *Kamus Komputer dan Istilah Teknologi Informasi*. Bandung : Informatika.

Nama Perusahaan : Qorinah Day Spa and Salon

Nama narasumber : Lana

Jabatan : Owner dan Manager

Tanggal : 19 November 2013

N : Narasumber

P : Pewawancara

P-01 : Selamat siang Mbak, saya ingin sedikit mewawancarai Mbak Lana mengenai Qorinah

N-01 : Ya, boleh Mbak.

P-02 : Pertanyaan pertama, peran Mbak Lana di Qorinah sebagai apa?

N-02 : Saya owner sekaligus manager Mbak.

P-03 : Di Qorinah, ada berapa jabatan Mbak?

N-03 : Di Qorinah ada 2 jabatan Mbak. Yaitu manager, dan terapis.

P-04 : Oh jadi ada 2 ya Mbak. Nah kalau manager tugas seperti apa yang anda jalankan Mbak?

N-04 : Tugas saya sebagai manager adalah mengecek laporan transaksi, presensi, kinerja karyawan saya, laporan treatment. Saya juga bisa berperan sebagai terapis Mbak.

P-05 : Pada saat ini, apakah ada sistem yang mendukung kinerja Qorinah, Mbak?

N-05 : Untuk saat ini kita masih menggunakan sistem manual Mbak.

P-06 : Lalu, data apa saja yang anda butuhkan Mbak?

N-06: Data yang saya butuhkan antara lain seperti data pelanggan, presensi karyawan, data treatment, dan data karyawan. Selain itu yang utama saya butuhkan adalah laporan dari yang saya jelaskan tadi Mbak.

P-07 : Menurut Mbak, apakah Qorinah perlu database ataupun sistem yang mendukung kinerja Mbak?

N-07 : Ya menurut saya sangat perlu Mbak.

P-08 : Bagaimana pendapat Mbak kalau diadakannya sistem manajemen untuk Qorinah?

N-08 : Menurut saya itu ide yang bagus sekali Mbak, biar memudahkan pekerjaan saya dan karyawan disini.

P-09 : Menurut Mbak, apa saja yang harus ada di dalam sistem tersebut?

N-08 : Saya mengharapkan adanya pencarian data, input data, manajemen data, dan laporan Mbak.

P-10 : Ok, mungkin sekian dulu wawancaranya Mbak. Terima kasih untuk waktu dan informasinya

N-10 : Sama-sama, Mbak.

Nama Perusahaan : Qorinah Day Spa and Salon

Nama narasumber : Adel

Jabatan : Terapis

Tanggal : 19 November 2013

Keterangan : Verbatim 2

N : Narasumber

P : Pewawancara

P-01 : Selamat siang Mbak Adel, saya ingin mewawancarai Mbak Adel mengenai Qorinah

N-01 : Ya boleh Mbak.

P-02 : Peran Mbak Adel di Qorinah sebagai apa?

N-02 : Saya sebagai terapis di Qorinah, Mbak.

P-03 : Kalau sebagai terapis, tugasnya apa saja Mbak?

N-03 : Tugas saya melayani customer, yang termasuk mengurusinya, dan melakukan terapi sesuai dengan treatment yang dipesan oleh customer Mbak.

P-04 : Lalu, data apa saja yang anda butuhkan?

N-04 : Data yang saya butuhkan antara lain data pelanggan seperti nama, alamat, email, nomor telepon, tanggal lahir, dan data treatment.

P-05 : Dan laporan apa saja yang anda buat Mbak?

N-05 : Biasanya laporan tentang transaksi dan treatment apa saja yang sudah diorder Mbak.

P-06 : Apakah ada tugas lain lagi yang dilakukan terapis Mbak?

N-06 : Untuk sementara ini hanya itu saja Mbak. Oh iya, ada satu lagi yaitu melakukan presensi harian Mbak.

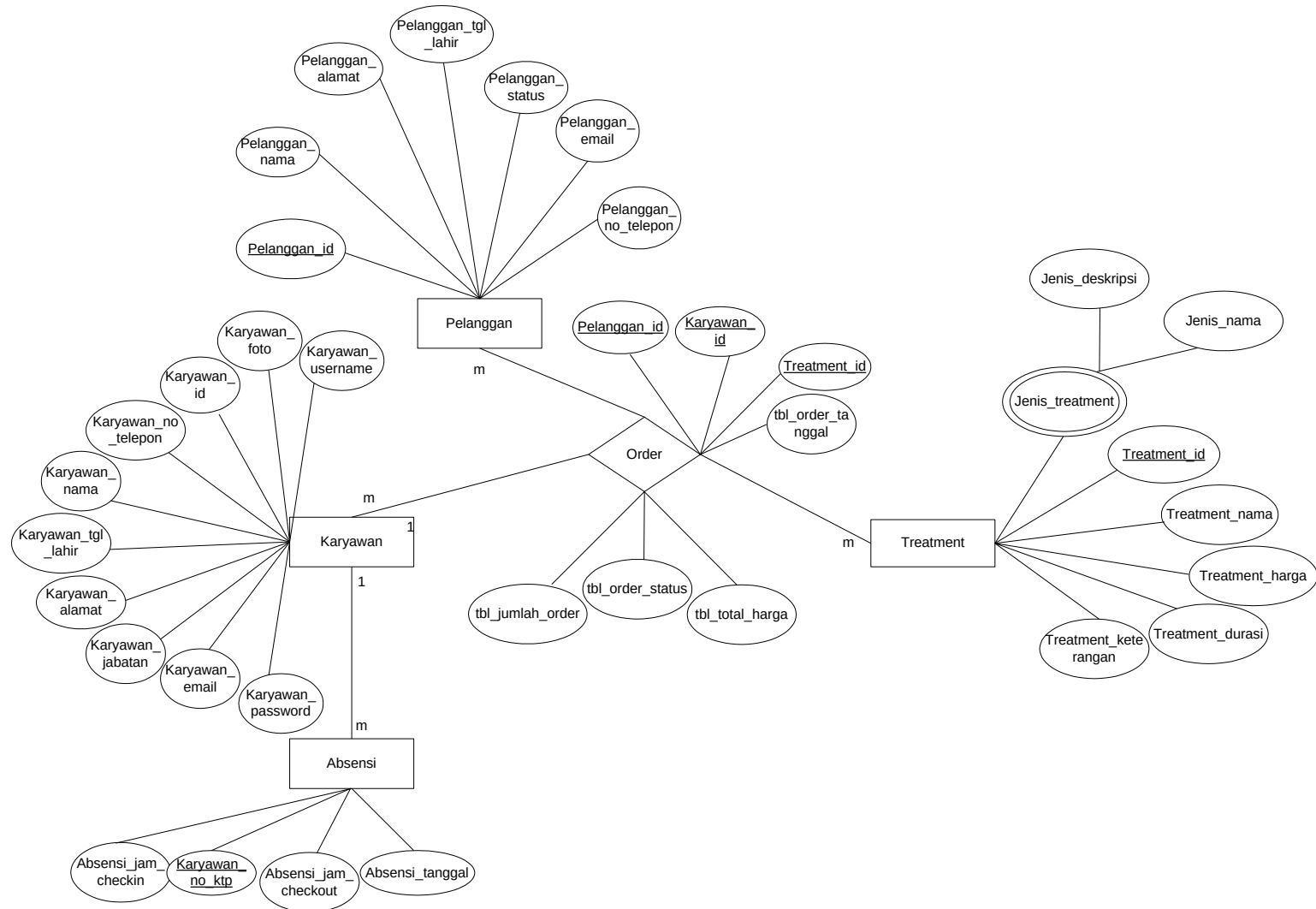
P-07 : Menurut anda bagaimana seandainya diadakan sistem manajemen di Qorinah?

N-07 : Saya sangat setuju jika ada sistem manajemen di Qorinah, Mbak. Agar memudahkan pekerjaan kami untuk melayani pelanggan, dan pekerjaan lainnya yang tadi saya sebut.

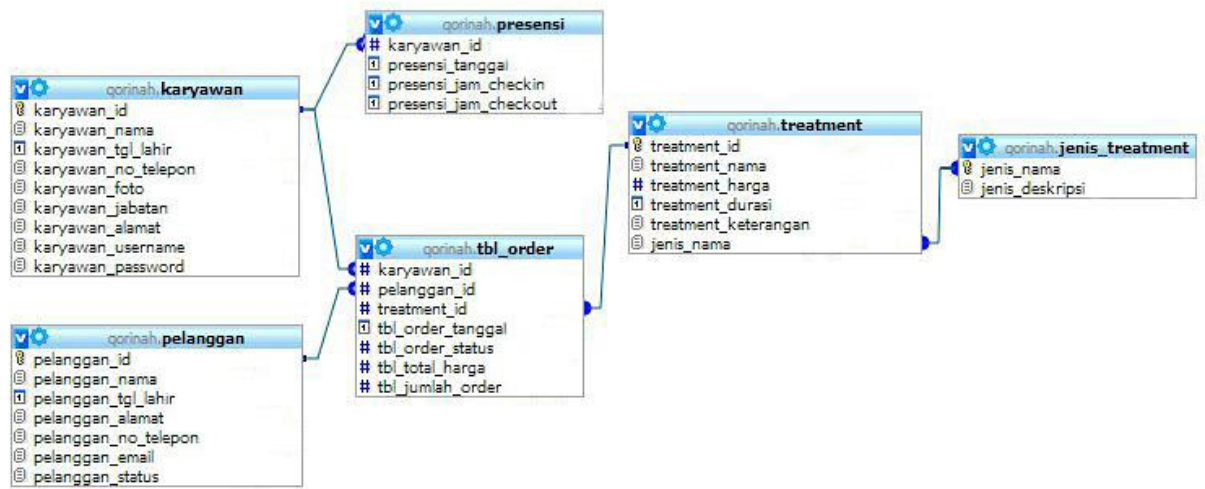
P-08 : Oke, cukup sekian wawancara saya Mbak. Terima kasih atas waktu dan informasinya Mbak.

N-08 : Sama-sama Mbak.

ENTITY RELATIONSHIP DIAGRAM (ERD)



LOGICAL SCHEMA



NORMALISASI TABEL

Tabel Karyawan

Field	Type	1NF	2NF	3NF
karyawan_id	Int (11)	Atomic	Primary key	Dependency on primary key
karyawan_nama	Varchar (50)	Atomic		
karyawan_tgl_lahir	date	Atomic		
karyawan_no_telepon	Varchar (15)	Atomic		
karyawan_foto	Varchar (200)	Atomic		
karyawan_no_ktp	Varchar (10)	Atomic		
karyawan_jabatan	Varchar (20)	Atomic		
karyawan_alamat	Varchar (150)	Atomic		
karyawan_username	Varchar(20)	Atomic		
karyawan_password	Varchar(20)	Atomic		

Tabel Pelanggan

Field	Type	1NF	2NF	3NF
pelanggan_id	Int (11)	Atomic	Primary Key	Dependency on primary key
pelanggan_nama	Varchar (50)	Atomic		
pelanggan_tgl_lahir	Date	Atomic		

pelanggan_alamat	Varchar (255)	Atomic		
pelanggan_no_telepon	Int (11)	Atomic		
pelanggan_email	Varchar (50)	Atomic		
pelanggan_status	Varchar (25)	Atomic		

Tabel Treatment

Field	Type	1NF	2NF	3NF
treatment_id	Int (11)	Atomic	Primary Key	Dependency on primary key
treatment_nama	Varchar (50)	Atomic		
treatment_harga	Float	Atomic		
jenis_nama	Varchar (50)	Atomic		
treatment_durasi	Varchar (50)	Atomic		
treatment_keterangan	Varchar (50)	Atomic		

Tabel Absensi

Field	Type	1NF	2NF	3NF
karyawan_id	Int (11)	Atomic	Primary Key	Dependency on primary key
absensi_tanggal	Date	Atomic		
absensi_jam_checkin	Time	Atomic		
absensi_jam_checkout	Time	Atomic		

Tabel Order

Field	Type	1NF	2NF	3NF
karyawan_id	Int (11)	Atomic	Foreign Key	Dependency on primary key
pelanggan_id	Int (11)	Atomic	Foreign Key	
treatment_id	Int (11)	Atomic	Foreign Key	
tbl_order_tanggal	Date	Atomic		
tbl_order_status	Tinyint(1)	Atomic		
tbl_total_harga	Float	Atomic		
tbl_jumlah_order	Int (11)	Atomic		



Use Case Diagram

