

**SISTEM INFORMASI RESERVASI
PADA HOTEL GAJAH MADA PONOROGO
GUNA MENINGKATKAN PELAYANAN
PADA KONSUMEN**

SKRIPSI

*Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Teknik*



Disusun oleh:

**DIDIEK RIADI
NIM. 0210633025 - 63**

**DEPARTEMEN PENDIDIKAN NASIONAL
UNIVERSITAS BRAWIJAYA
FAKULTAS TEKNIK
MALANG
2009**

KATA PENGANTAR

Dengan memanjatkan puji syukur kehadirat Allah Subhanahu Wata'ala atas segala rahmat-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul “SISTEM INFORMASI RESERVASI PADA HOTEL GAJAH MADA PONOROGO GUNA MENINGKATKAN PELAYANAN PADA KONSUMEN” ini.

Penyusunan skripsi ini diajukan untuk memenuhi sebagian persyaratan memperoleh gelar Sarjana Teknik, dan merupakan salah satu syarat yang harus ditempuh oleh setiap mahasiswa jurusan Teknik Elektro Universitas Brawijaya Malang.

Pada kesempatan kali ini tidak lupa saya sampaikan ucapan terima kasih yang sebesar-besarnya kepada:

- Ibu, Bapak dan seluruh keluargaku untuk seluruh do'a dan dukungannya yang telah diberikan selama studi hingga terselesaikannya skripsi ini.
- Bapak Heru Nurwarsiti, Ir,M.Kom, selaku ketua jurusan Teknik Elektro dan Bapak Rudi Yuwono, ST.,M.Sc selaku Sekertaris JURusan Teknik Elektro.
- Bapak Arief Andy Soebroto ST,M.Kom selaku KKDK Teknik Informatika dan Komputer.
- Bapak Arief Andy Soebroto ST,M.Kom dan Bapak Heru Nurwarsiti, Ir,M.Kom selaku dosen pembimbing yang telah memberikan pengarahan dan bimbingan sampai terselesaikannya penelitian ini.
- Bapak ibu Dosen serta segenap staf dan karyawan Jurusan Teknik Elektro Universitas Brawijaya.
- Irma Endriana A.md atas dukungan dan motivasinya yang tiada habisnya.
- Adikku Anggita, Dita miaw, Wida,Luvi, Ocha,Fitri,Wiwiek,Naomi, Phe, Iclang, Gendut, Anung, Gogon, Nanang, Naben, Serta temen-temen di P.Lan terima kasih atas dukungannya..

- Teman-teman paket E dan teman-teman Elektro 2002 atas masukan dan bantuannya
- Teman-teman di unyil terima kasih atas semuanya.
- Rekan-rekan sahabat yang tidak dapat penulis sebut satu persatu.

Penulis menyadari bahwa skripsi ini masih banyak kekurangan dan jauh dari kesempurnaan. Oleh karena itu kritik dan saran sangat diharapkan demi kesempurnaan skripsi ini. Akhir kata semoga skripsi ini dapat bermanfaat bagi rekan-rekan mahasiswa khususnya dan bagi pembaca pada umumnya.

Malang, Agustus 2009

Penulis



ABSTRAKSI

Didiek Riadi, Desain Sistem Informasi Reservasi Pada Hotel Gajah Mada Ponorogo Guna Meningkatkan Pelayanan pada Konsumen, Skripsi Jurusan Teknik Elektro, Fakultas Teknik, Universitas Brawijaya. Dosen Pembimbing Arief Andy Soebroto, ST,M.Kom dan Heru Nurwarsito M.Kom.

Kata kunci: Reservasi online, Pelayanan Informasi.

Sistem Informasi Reservasi pada Hotel Gajah Mada Ponorogo masih dilakukan secara offline, yaitu pelanggan datang langsung ke hotel untuk melakukan reservasi. Kekurangan system konvensional tersebut antara lain customer tidak dapat melihat secara langsung informasi mengenai fasilitas yang didapat nantinya serta dengan system tersebut akan memakan waktu yang lama. Dengan adanya web Gajahmada.com, maka akan mempermudah bagian pemesanan atau front office dalam memberikan informasi, sekaligus memudahkan proses reservasi yang dilakukan secara online.

Desain system ini bertujuan untuk mengoptimalkan reservasi Hotel Gajah Mada terutama reservasi secara online dengan menggunakan bahasa PHP dan MySQL



DAFTAR ISI

Lembar Persetujuan	i
Abstraksi	ii
Kata Pengantar.....	iii
Lembar Persembahan.....	iv
Daftar Isi.....	v
Daftar gambar.....	xi
Daftar tabel.....	xiv
BAB I Pendahuluan	
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah.....	2
1.4 Tujuan.....	3
1.5 Manfaat.....	3
1.6 Sistematika Penulisan.....	3
1.7 Mata Kuliah Penunjang.....	5
BAB II Landasan Teori	
2.1 Sistem Informasi.....	5
2.1.1 Pengertian Sistem	5
2.1.2 Pengertian Data dan Informasi.....	6
2.1.3 Fungsi Informasi Bagi Organisasi.....	6
2.1.4 Konsep Dasar Sistem Informasi.....	7
2.2 Konsep Desain Sistem.....	7



2.2.1	Pengertian Desain Sistem.....	7
2.2.2	Tujuan Desain Sistem	7
2.2.3	Perangkat Pemodelan Sistem	8
2.2.4	Sistem Prosedure Diagram	8
2.3	Context Diagram.....	9
2.3.1	Data Flow Diagram.....	9
2.3.2	Entity-Relationship Diagram.....	10
2.3.3	Flowchart Sistem	11
2.4	Data Base.....	13
2.4.1	Pengertian Data Base	13
2.4.2	Kegunaan Data Base	13
2.4.3	Sistem Data Base Internet	14
2.4.4	Pengertian Web Data base.....	14
2.5	Pengertian PHP.....	15
2.5.1	Teknik Penulisan PHP	16
1.5.2	Pengertian Editor	17
2.6	Pengenalan MySQL	17
2.6.1	Pengertian MySQL	17
2.6.2	Fungsi-fungsi MySQL	20
2.7	Pengertian Hotel	22
2.7.1	Pengertian Reservation	22
2.7.2	Departement Reservation	23
2.7.3	Alur Sistem Reservasi.....	25
2.7.4	Prosedure Reservasi.....	26
2.8	Gambaran Umum Perusahaan	26
2.8.1	Sejarah Hotel Gajah Mada	26
2.8.2	Struktur Organisasi	



BAB III Metode Penelitian	34
3.1 Observasi	35
3.2 Studi Literatur	35
3.3 Perancangan	35
3.4 Implementasi	35
3.5 Pengujian Dan Analisis	36
3.6 Pengambilan kesimpulan	36
3.7 Penulisan Laporan	36
BAB IV Perancangan	37
4.1 Daftar Kerja Reservasi	39
4.2 Analisa Masalah	41
4.3 Akibat Yang Ditimbulkan	42
4.4 Pemecahan Masalah	42
4.5 Desain Proses	43
4.5.1 Context Diagram	43
4.5.2 Dekomposisi Fungsional	44
4.5.3 Data Flow Diagram	45
4.6 ERD	51
4.6.1 Conseptual Model	52
4.6.2 Tabel	53
4.7 Desain Algoritma	56
4.7.1 Flowchart Login	56
4.7.2 Floechart input Type Kamar	57
4.7.3 Floechart Entry Data Kamar	57
4.7.4 Flowchart Edit Data kamar	60
4.7.5 Floechart Edit Data Reservasi	61
4.7.6 Flowchart Laporan kamar	64



4.7.7	Flowchart laporan Pembayaran.....	65
4.7.8	Flowchart Laporan Reservasi.....	66
4.7.9	Diagram Sekuensial.....	66
4.8	Desain	73
4.8.1	Desain Utama	73
4.8.2	Desain Room Type	74
4.8.3	Desain Room Rate	75
4.8.4	Desain Reservation	76
4.8.5	Desain laporan Pembayaran.....	77
BAB V Implementasi.....		78
5.1	Spesifikasi Hardware dan Software.....	78
5.2	Tampilan Program.....	79
5.2.1	Menu	79
5.2.1.1	Menu Pelanggan.....	79
5.2.2	Content Management System (CMS).....	79
5.2.3	Tampilan dan Logika Program.....	80
5.2.3.1	Program Admin.....	80
5.2.3.2	Program Utama	81
5.2.3.3	Program Input Type Kamar Pada CMS Admin.....	82
5.2.3.4	Program Entry Kamar	83
5.2.3.5	Menampilkan room rate pada public.....	84
5.2.3.6	Program Reservasi	86
5.2.3.7	KOnfirmasi Reservasi	90
5.2.3.8	Program Validasi Reservasi oleh Admin	92

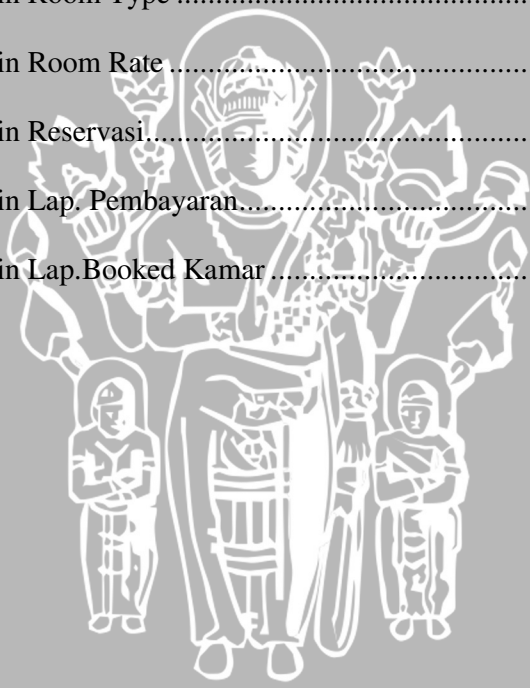
BAB VI Pengujian.....	95
6.1 Pengujian Algoritma.....	95
6.1.1 Pengujian Proses Log in.....	95
6.1.2 Pengujian Proses Data.....	97
6.1.3 Pengujian Proses Edit Data.....	99
6.1.4 Pengujian Proses Hapus Data.....	101
6.2 Pengujian Validasi.....	103
6.2.1 Kasus Uji Validasi.....	104
6.2.1.1 Kasus Uji login.....	104
6.2.1.2 Kasus Uji logout.....	105
6.2.1.3 Kasus uji tambah user.....	106
6.2.1.4 Kasus Uji melihat User.....	108
6.2.1.5 Kasus Uji Mengedit Data User.....	109
6.2.1.6 Kasus Uji Menghapus User.....	101
6.2.1.7 Kasus Uji Modifikasi data User.....	112
6.2.1.8 Kasus Uji Menambah Artikel.....	113
6.2.1.9 Kasus Uji mengedit Artikel.....	115
6.2.1.10 Kasus Uji Membuat Topik Forum.....	115
6.2.1.11 Kasus Uji Membalas Tofik Forum.....	116
6.2.1.12 Kasus Uji Hapus Tofik forum.....	118
6.2.3 Hasil Pengujian Validasi.....	118
6.2.4 Pengujian kecepatan Queri.....	121
BAB VII Penutup.....	125
7.1 Kesimpulan.....	125



DAFTAR GAMBAR

Gambar 2.1	Proses Pengolahan Data menjadi Informasi.....	6
Gambar 2.2	Struktur Organisasi Perusahaan.....	31
Gambar 4.1	Context Diagram.....	44
Gambar 4.2	Dekomposisi Fungsional.....	44
Gambar 4.3	Data Flow Diagram level1	45
Gambar 4.4	Data Floe diagram Level2 Reservasi	47
Gambar 4.5	Data Floe diagram Level2 Pembayaran.....	49
Gambar 4.6	Data Floe diagram Level2 Laporan	50
Gambar 4.7	Conseptual Model.....	52
Gambar 4.8	Physical Model	52
Gambar 4.9	Flowchart Login	56
Gambar 4.10	Flowchart Input Tipe Kamar	57
Gambar 4.11	Flowchart Entri Data Kamar	58
Gambar 4.12	Flowchart Pilih Tipe Kamar	59
Gambar 4.13	Flowchart Edit Data Kamar	60
Gambar 4.14	Flowchart Reservasi.....	61
Gambar 4.15	Flowchart Edit Data Reservasi	62
Gambar 4.16	Flowchart Inputkan Pembayaran	63
Gambar 4.17	Flowchart Laporan Kamar	64
Gambar 4.18	Flowchart Laporan Pembayaran.....	65
Gambar 4.19	Flowchart Laporan Reservasi	66
Gambar 4.20	Squence Diagram Login.....	67
Gambar 4.21	Squence Diagram log out.....	68

Gambar 4.22	Squence Diagram lihat user.....	68
Gambar 4.23	Squence Diagram add user.....	69
Gambar 4.24	Squence Diagram Delete user	70
Gambar 4.25	Squence Diagram Update Data	70
Gambar 4.26	Squence Diagram Update Data	71
Gambar 4.27	Squence Diagram Update Data	71
Gambar 4.28	Squence Diagram Update Data	72
Gambar 4.29	Desain Utama	73
Gambar 4.30	Desain Room Type	74
Gambar 4.31	Desain Room Rate	75
Gambar 4.32	Desain Reservasi.....	76
Gambar 4.33	Desain Lap. Pembayaran.....	77
Gambar 4.34	Desain Lap.Booked Kamar	77



DAFTAR TABEL

Tabel 2.1	Simbol-simbol IOFC.....	8
Tabel 2.2	Simbol-simbol DFD.....	10
Tabel 2.3	Simbol-simbol Kardinalitas.....	11
Tabel 2.4	Simbol-simbol Flowchart.....	12
Tabel 2.5	Fungsi MySQL.....	20
Tabel 2.6	Perincian Daya Tampung Hotel.....	29
Tabel 2.7	Tarif kamar masing-masing kelas.....	30
Tabel 4.1	Sistem Prosedure Diagram.....	38
Tabel 4.2	Daftar Kerja Reservasi.....	39
Tabel 4.3	User.....	53
Tabel 4.4	Tabel Kamar.....	53
Tabel 4.5	Pembayaran.....	54
Tabel 4.6	Reservasi.....	55
Tabel 5.1	Spesifikasi Hadware.....	78
Tabel 6.1	Kasus Uji untuk Pengujian Proses login.....	97
Tabel 6.2	Kasus Uji untuk Pengujian Proses tambah data.....	99
Tabel 6.3	Kasus Uji untuk Pengujian Proses edit Data.....	101
Tabel 6.4	Kasus Uji untuk Pengujian Proses Hapus data.....	103
Tabel 6.5	Kasus Uji dimana username dan paswod ada dalam system.....	104
Tabel 6.6	Kasus Uji dimana username dan paswod tidak ada dalam system.....	105
Tabel 6.7	Kasus Uji logout.....	105
Tabel 6.8	Kasus Uji tambah user dimana masukan data kosong.....	106
Tabel 6.9	Kasus Uji Melihat Daftar user dimana user sudah ada.....	108

Tabel 6.10	Kasus Uji Melihat Daftar user.....	109
Tabel 6.11	Kasus Uji Mengedit Data user.....	109
Tabel 6.12	Kasus Uji Mengedit Data user dimana username tidak boleh kosong	110
Tabel 6.13	Kasus Uji Mengedit Data user dimana username sudah ada	110
Tabel 6.14	Kasus Uji Menghapus Data user	111
Tabel 6.15	Kasus Uji Modifikasi data User.....	112
Tabel 6.16	Kasus Uji Modifikasi Data User dimana paswod aktif tidak cocok	112
Tabel 6.17	Kasus Uji Menambah Artikel.....	113
Tabel 6.18	Kasus Uji Menambah Artikel dengan masukan judul kosong	114
Tabel 6.19	Kasus Uji Mengedit Artikel	115
Tabel 6.20	Kasus Uji Menghapus Artikel	115
Tabel 6.21	Kasus Uji Membuat topik Forum	116
Tabel 6.22	Kasus Uji Membuat Topik Forum dengan Masukan judul kosong	116
Tabel 6.23	Kasus Uji Membalas Topik Forum.....	117
Tabel 6.24	Kasus Uji Membalas Topik Forum.....	118
Tabel 6.25	Hasil Pengujian Validasi.....	119
Tabel 6.26	Pengujian Waktu akses queri	122
Tabel 6.27	Perbandingan waktu akses per record dari queri data.....	123

BAB I PENDAHULUAN

1.1 Latar Belakang

Hotel-hotel yang ada di Indonesia, baik itu hotel berbintang atau kelas melati, harus menghadapi persaingan bebas pada era globalisasi saat ini. Untuk itu semua sumber daya di hotel harus dapat dikerahkan secara maksimal dan profesional untuk mendukung keberhasilan hotel tersebut. Keberhasilan manajemen suatu perusahaan dapat dilihat dari kepuasan yang konsumen dapatkan, seberapa banyak pengunjung tiap minggunya. HOTEL GAJAH MADA merupakan satu dari banyak hotel yang ada di Kota Ponorogo yang memiliki banyak pelanggan. Pelanggan mereka terdiri dari orang asli Indonesia ataupun Luar Negeri.

Sebagai Hotel yang terkenal dan memiliki banyak pelanggan, Hotel Gajah Mada sudah memiliki fasilitas-fasilitas yang memadai baik fasilitas kamar sampai pelayanan. Untuk meningkatkan pelayanan hotel dan kepuasan customer Hotel Gajah Mada membutuhkan pelayanan yang cepat, tepat dan akurat. Pelayanan yang dimaksud bukan hanya terhadap pelanggan hotel yang datang langsung melakukan transaksi tetapi juga pelanggan hotel yang menggunakan fasilitas pelayanan reservasi.

Setelah mengalami perkembangan yang cukup pesat Hotel Gajah Mada membutuhkan Sistem Informasi Reservasi dalam menunjang peningkatan pelayanan. Sistem Informasi Reservasi dalam Hotel Gajah Mada masih menggunakan media komunikasi telephone yang sehingga pelanggan tidak langsung bisa mengecek dan bertanya banyak mengenai informasi fasilitas dan akomodasi yang didapat nantinya. Masalah yang sering timbul disini adalah untuk proses validasi data reservasi diperlukan waktu yang terlalu lama karena batas waktu maksimal yang diberikan pelanggan untuk konfirmasi yaitu sehari sebelum tanggal cek in.. Pengendalian penyimpanan atau pencatatan data reservasi dalam jumlah besar masih kurang, terkadang kamar yang sudah dipesan tapi sudah ditempati orang lain. Selain itu pelanggan yang melakukan reservasi melalui via telephone tidak bisa melihat secara langsung fasilitas-fasilitas kamar yang dimiliki

oleh Hotel Gajah Mada. Sebagai akibat dari Sistem Informasi Reservasi yang lama adalah penurunan jumlah pelanggan hotel dan memperkecil pangsa pasar.

Dari masalah di atas, maka dalam tugas proyek ini dibuat sistem informasi pemesanan kamar secara online, yang mana pelanggan bisa melakukan pemesanan kamar dengan mudah, selain itu data pelanggan bisa tersimpan dengan baik yang nantinya bisa dijadikan laporan oleh Room Servis selaku pengendali ada dan tidaknya kamar. Kelebihan dari sistem ini adalah pelanggan bisa melakukan pemesanan secara online, karena dengan sistem online pelanggan mendapatkan kemudahan dalam melakukan proses reservasi dengan bisa melihat secara langsung fasilitas-fasilitas kamar yang akan dipesan tanpa harus datang langsung ke Hotel Gajah Mada. Sedangkan manfaat sistem pemesanan secara online bagi Hotel Gajah Mada adalah jumlah pelanggan akan bertambah seiring adanya kemudahan bagi pelanggan dalam melakukan proses reservasi.

1.2 Rumusan Masalah

Berdasarkan pada permasalahan yang telah dijelaskan pada bagian latar belakang, maka rumusan masalah dikhususkan pada:

1. Merancang suatu sistem pemesanan secara online dengan bahasa pemrograman PHP, untuk mempermudah proses reservasi.
2. Merancang desain sistem informasi reservasi pada Gajah Mada Ponorogo guna meningkatkan pelayanan pada konsumen.
3. Implementasi sistem ini dibuat dengan bahasa pemrograman berbasis *web* yaitu PHP dengan database MySQL serta Apache Web Server sebagai server.
4. Menguji aplikasi sistem reservasi

1.3 Batasan Masalah

Agar perancangan dan pembahasan yang dihasilkan terarah, maka perlu dilakukan pembatasan masalah sebagai berikut :

1. Desain sistem informasi.
2. Menggunakan pemrograman PHP
3. Menggunakan pemrograman MySQL

1.4 Tujuan

Tujuan penulisan skripsi ini adalah Merancang suatu sistem pemesanan secara online dengan bahasa pemrograman PHP, untuk mempermudah proses reservasi.

1.5 Manfaat

Adapun manfaat dari pembuatan sistem informasi akademik berbasis web ini antara lain :

1. Memberikan suatu solusi baru tentang proses reservasi secara online guna mempermudah proses pemesanan kamar pada Hotel Gajah Mada.
2. Mempermudah pelanggan untuk melihat fasilitas-fasilitas Hotel Gajah Mada.

1.6 Sistematika Penulisan

Pembahasan Tugas Akhir ini terdiri dari 7 bab yang disusun dengan sistematika penulisan sebagai berikut :

Bab I : Pendahuluan

Pada bab ini terdiri dari latar belakang masalah, rumusan masalah, tujuan Tugas Akhir, kegunaan Tugas Akhir serta sistematika penulisan.

Bab II : Landasan Teori

Berisi tentang pengertian dari sistem informasi, Penjelasan mengenai PHP, Penjelasan umum tentang Data Flow Diagram, Penjelasan ER-Model, Penjelasan umum tentang Flowchart, Sejarah tentang Hotel Gajah Mada Ponorogo.

Bab III : Metodologi Penelitian

Berisi tentang metode penelitian yang digunakan dalam perancangan dan pengujian sistem.

Bab IV : Perancangan

Pada bab ini dijelaskan tentang Berisi tentang analisa masalah, usulan pemecahan masalah yang meliputi Conteks Diagram, Data Flow Diagram, ER-Model, Desain Menu dan Rancangan Database.

Bab V : Implementasi Sistem

Pada bab ini akan dibahas mengenai Berisi spesifikasi program, Layout Program, dan Listing Program serta cara menjalankan program.

Bab VI : Pengujian dan Analisis

Bagian ini berisi penjelasan tentang strategi pengujian (unit, integrasi dan validasi) dan teknik pengujian yang dilakukan. Dijelaskan juga seluruh kasus uji beserta hasil pengujiannya.

Bab VII : Penutup

Bagian ini berisi kesimpulan dan saran. Kesimpulan didasarkan atas pengujian dan analisis yang dilakukan di dalam proses penelitian.



BAB II DASAR TEORI

2.1 Sistem Informasi

Sesungguhnya yang dimaksud dengan sistem informasi tidak harus melibatkan komputer. Sistem informasi yang menggunakan komputer biasa disebut sistem informasi berbasis komputer (Computer-Based Information Systems atau CBIS). Dalam prakteknya, istilah sistem informasi lebih sering dipakai tanpa embel-embel berbasis komputer walaupun dalam kenyataannya komputer merupakan bagian yang penting.

2.1.1 Pengertian Sistem

Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu [ANK-05:23-25].

Dari definisi di atas dapat disimpulkan bahwa sistem adalah kumpulan dari elemen-elemen yang saling terkait / berinteraksi untuk mencapai suatu tujuan tertentu.

Seperti yang telah disebutkan di atas, sistem terdiri dari elemen-elemen yang saling berkaitan. Elemen-elemen yang menyusun sebuah sistem tersebut terdiri dari :

1. *Tujuan*, merupakan tujuan dari sistem tersebut.
2. *Batasan*, merupakan batasan-batasan yang ada dalam mencapai tujuan dari sistem.
3. *Kontrol*, merupakan pengawas dari pelaksanaan pencapaian tujuan sistem.
4. *Input*, merupakan bagian dari sistem yang bertugas untuk menerima data masukan.
5. *Proses*, merupakan bagian yang memproses masukan data menjadi informasi sesuai dengan keinginan penerima.
6. *Output*, merupakan keluaran atau tujuan akhir dari sistem.
7. *Umpan balik*, merupakan informasi dari sistem dan menyediakan informasi bagi kontrol dimana umpan balik ini dapat berupa perbaikan, pemeliharaan dan lain-lain.

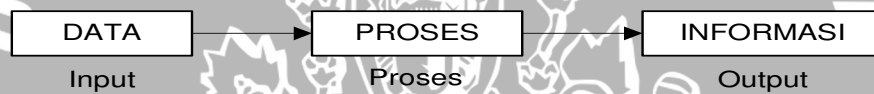
2.1.2 Pengertian Data dan Informasi

Data adalah penggambaran dari sesuatu kejadian yang kita hadapi atau kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata”.

Sedangkan kejadian (event) itu sendiri adalah sesuatu yang terjadi pada saat tertentu [KAD-04:45- 46].

Informasi adalah data yang telah diolah menjadi sebuah bentuk yang berarti bagi penerimanya dan bermanfaat dalam pengambilan keputusan saat ini atau saat mendatang. [HAD-04:63 - 67].

Adapun proses pengolahan data menjadi informasi dapat dilihat pada gambar berikut:



Gambar 5.2.1 Proses pengolahan data menjadi informasi

Sumber : [KAD-04:47].

Dari definisi diatas dapat disimpulkan bahwa data adalah fakta-fakta atau gambaran nyata yang diperoleh dari suatu kejadian. Sedangkan informasi adalah data yang telah diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi penerimanya untuk kepentingan pengambilan keputusan.

2.1.3 Fungsi Informasi Bagi Organisasi

Informasi merupakan sumber daya strategis bagi suatu organisasi. Informasi dianggap sebagai suatu entitas yang mendukung kelangsunagn hidup organisasi tersebut. Para pelaku organisasi membutuhkan informasi untuk mendefinisikan tujuan serta memberikan petunjuk dalam pengambilan keputusan organisasi tersebut. Informasi tidak hanya dibutuhkan oleh para pelaku organisasi

saja melainkan juga oleh pihak-pihak luar yang berkaitan dengan organisasi tersebut.

Oleh karena itu sistem informasi harus mampu menghasilkan informasi yang berguna bagi pengelola organisasi dalam memecahkan setiap tingkatan persoalan yang dihadapi suatu organisasi.

2.1.4 Konsep Dasar Sistem Informasi

Sistem informasi adalah jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama melakukan suatu kegiatan untuk menyelesaikan suatu sasaran tertentu [HAD-04:101].

Dari beberapa definisi diatas dapat disimpulkan bahwa sistem informasi mencakup sejumlah komponen (manusia, komputer, teknologi informasi dan prosedur kerja), ada sesuatu yang diproses (data menjadi informasi) dan dimaksudkan untuk mencapai suatu sasaran atau tujuan.

2.2 Konsep Desain Sistem

2.2.1 Pengertian Desain Sistem

Desain sistem adalah spesifikasi atau konstruksi solusi yang teknis dan berbasis komputer untuk persyaratan bisnis yang diidentifikasi dalam analisis sistem [JLK-04:38-40].

Dari definisi diatas dapat disimpulkan desain sistem adalah tahap setelah analisis sistem yang meliputi penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen terpisah ke dalam satu kesatuan yang utuh dan berfungsi.

2.2.2 Tujuan Desain Sistem

Tahap desain sistem memiliki dua tujuan utama yaitu:

1. Untuk memenuhi kebutuhan kepada pemakai sistem.

2. Untuk memberikan gambaran yang jelas dan rancang bangun yang lengkap kepada pemrogram komputer dan ahli-ahli teknik lainnya yang terlibat.

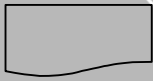

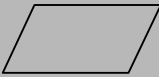




2.2.3 Perangkat Pemodelan Sistem

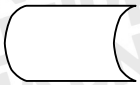





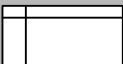

Model sistem memegang peranan penting dalam pengembangan sistem. Sebagai analis atau pengguna sistem, akan terus-menerus berhubungan dengan persoalan yang tidak terstruktur. Salah satu cara untuk menyusun persoalan tersebut adalah dengan menggambar model. Model adalah representasi kenyataan.

2.2.4 System Procedure Diagram

Pengantar Perancangan System, mendefinisikan System Procedure Diagram yaitu suatu procedure yang digunakan untuk menghubungkan antara bagian (pelaku proses), proses (manual atau berbasis komputer) dan aliran data (dalam bentuk dokumen keluaran, dan masukkan) [HAD-04:45].

Tabel 2.1 Simbol- simbol IOFC

Simbol	Nama
	<i>Document</i>
	<i>Kondisi</i>
	<i>Data</i>
	<i>Connector</i>
	<i>Off-page Connector</i>
	<i>Manual Operation</i>
	<i>Arsip Sementara</i>

	<i>Stored Data</i>
	<i>Process</i>
	<i>Penghubung</i>
	<i>Manual Input</i>
	<i>Monitor</i>
	<i>Pita Kertas</i>
	<i>Penyimpanan internal</i>
	<i>Proses yang tidak didefinisikan</i>

Sumber : [HIK-03:30]

2.3 Context Diagram

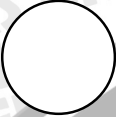


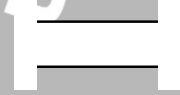
Context Diagram adalah sebuah diagram sederhana yang menggambarkan hubungan antara entity luar, masukan dan keluaran dari sistem dan direpresentasikan dengan lingkaran tunggal yang memiliki keseluruhan sistem [HAD-04:67].

2.3.1 Data Flow Diagram (DFD)

Data Flow Diagram (DFD) adalah peralatan yang berfungsi untuk menggambarkan secara rinci mengenai sistem sebagai jaringan kerja antar fungsi yang berhubungan satu sama lain dengan menunjukkan dari dan kemana data mengalir serta penyimpanannya [ABK-04:55].

Adapun simbol-simbol yang digunakan dalam penggambaran DFD dapat dilihat pada tabel 2.2

Tabel 2.2 Simbol-simbol DFD

Simbol	Nama	Keterangan
	<i>Process</i>	Simbol ini menunjukkan proses transformasi data atau dapat mewakili suatu pekerjaan atau proses
	<i>Entity</i>	Simbol ini menunjukkan sumber atau tujuan aliran data dari atau ke sistem
	<i>Data Flow</i>	Simbol ini menunjukkan aliran data
	<i>Data Store</i>	Simbol ini menunjukkan tempat penyimpanan data (file)

Sumber : [JLK-04:44]

2.3.2 Entity-Relationship Diagram (ERD)

Ada beberapa catatan mengenai pemodelan data. Model yang aktual disebut Entity Relationship Diagram (ERD). Karena model ini menjelaskan data dalam konteks entitas dan hubungan yang digambarkan oleh data tersebut. Dapat dikatakan bahwa ERD adalah gambaran awal dari sebuah data base [JLK-04:281].

Dengan demikian dapat disimpulkan bahwa ERD adalah sebuah model analisis sistem yang berbentuk diagram yang menunjukkan hubungan antar entitas atau data dari suatu sistem.

Berdasarkan relasinya, hubungan antar entitas dapat dibedakan menjadi:

1. Relasi *one to one* (1-1)

Satu record pada suatu tabel mempunyai hubungan dengan satu record pada tabel lainnya.



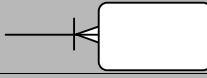


2. Relasi *one to many* (1-M)

Satu record pada suatu tabel mempunyai hubungan dengan lebih dari satu record pada tabel lainnya atau sebaliknya.

3. Relasi *many to many* (M-M)

Banyak record pada suatu tabel mempunyai hubungan dengan lebih dari satu record pada tabel lainnya. Relasi diatas memiliki kompleksitas atau tingkat tiap hubungan yang disebut dengan kardinalitas (*cardinality*).

Tabel 2.3 Sombol-simbol kardinalitas

Simbol	Nama	Keterangan
	Ke 1 hubungan	Tepat satu
	Ke 0 atau 1 hubungan	Hanya nol atau satu
	Ke banyak hubungan	Satu atau lebih
	Ke 0 atau lebih hubungan	Nol, satu atau lebih
	Ke lebih dari 1 hubungan	Lebih dari satu





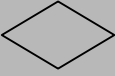
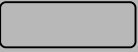
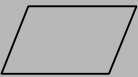

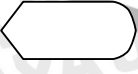
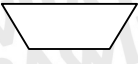
Sumber : [JLK-04:56]

2.3.3 FlowChart System

Flow Chart adalah metode yang menggambarkan tahapan dalam memecahan suatu masalah dengan mempresentasikan simbol-simbol tertentu yang mudah dimengerti, mudah digunakan dan standard [HAD-04:55].

Berikut ini adalah simbol-simbol yang digunakan dalam membuat *flowchart*.

Tabel 2.4 Simbol-simbol *flowchart*

<i>Simbol</i>	<i>Nama</i>	<i>Keterangan</i>
	<i>Dirrection</i>	Digunakan Untuk menghubungkan antara simbol yang satu dengan simbol yang lainnya.
	<i>Connector</i>	Simbol untuk keluar/masuk prosedur atau proses dalam halaman yang sama.
	<i>Off-Line Connector</i>	Simbol untuk keluar/masuk prosedur atau proses dalam halaman yang berbeda.
	<i>Process</i>	Menunjukkan pengolahan yang dilakukan oleh komputer.
	<i>Decision</i>	Simbol untuk kondisi yang akan menghasilkan beberapa kemungkinan jawaban/aksi.
	<i>Terminal</i>	Simbol untuk permulaan atau akhir dari suatu program.
	<i>Input-output</i>	Simbol yang menyatakan proses input dan output tanpa tergantung dengan jenis peralatannya.
	<i>Manual Input</i>	Simbol yang menggambarkan input manual seperti keyboard.
	<i>Display</i>	Menunjukkan output yang ditampilkan di monitor
	<i>Manual Process</i>	Menunjukkan kegiatan yang dilakukan secara manual

	<i>Sorting</i>	Menunjukkan proses pengurutan data
	<i>Magnetic Tape</i>	Sebagai media I/O berupa magnetic tape.
	<i>Magnetic Disk</i>	Magnetik disk/hardisk/fixed disk.
	<i>Mini Disk</i>	Berupa disket, floppy disk, flexible disk.
	Proses Terdefinisi	Proses yang detailnya tidak ditunjukkan, melainkan terpisah dan terdefinisi sendiri berupa program bagian/sub routine.

Sumber : [HAD-04:58]

2.4. Database

2.4.1 Pengertian Database

Database adalah skumpulan tabel-tabel (file-file) dengan pointer logis yang menghubungkan record-record pada satu tabel ke record pada tabel lain [JLK-04:541].

Dari definisi diatas dapat disimpulkan bahwa *database* adalah suatu pengorganisasian sekumpulan data yang saling terkait sehingga memudahkan aktivitas untuk memperoleh informasi.

2.4.2 Kegunaan Database

Beberapa kegunaan *database* adalah sebagai berikut:

1. Menghindari duplikasi dan inkonsistensi data.
2. Mempercepat pengaksesan data.

3. Dapat mengadakan pertukaran dan pemakaian data secara bersama antar pemakai.
4. Menjamin keamanan data.

2.4.3 Sistem Database Internet

Sistem Database internet adalah sistem database yang berjalan yang diatas jaringan kmputer internet yang menggunakan protokol komunikasi TC/IP.

2.4.4 Pengertian Web Database

Web Database adalah sistem penyampaian data yang dapat adiakses oleh bahasa pemrograman tertentu. Namun tidak seperti sistem database konvensional yang hanya ditujukan untuk platform tertentu, web database dapat diakses oleh aplikasi web yang tentunya lebih bersifat umum. Web dapat diakses oleh aplikasi web yang dikembangkan dengan HTML tag, kontrol ActiveX, pemrograman yang bersifat server-side melalui CGI, Microsoft IIS (Internet Information Server), atau skrip yang bersifat server-side seperti PHP [ANU-07].

Kemampuan untuk mengintegrasikan database ke dalam aplikasi yang dapat diakses pengguna menggunakan web browser inilah yang menjadikan suatu database biasa menjasi web database. Web database dapat digunakan untuk berbagai macam keperluan misalnya menyediakan data barang dan jasa untuk situs-situs reservasi.

Aplikasi reservasi merupakan aplikasi yang digunakan untuk melakukan bisnis dalam web. Web database merupakan komponen penting dalam membangun aplikasi reservasi. Dengan web database dapat disimpan berbagai informasi mengenai suatu produk baik barang maupun jasa, misalnya spesifikasinya, jumlah produk yang tersedia, harganya dan lain-lain. Dengan

adanya web database, maka dimungkinkan untuk membangun suatu situs web yang berisi katalog produk-produk yang akan ditawarkan.

2.5 Pengertian PHP

Ada beberapa pengertian tentang PHP. Akan tetapi, kurang lebih PHP dapat di ambil arti sebagai PHP Hypertext Preeprocessor. Ini merupakan bahasa yang hanya dapat berajalan pada server yang hasilnya dapat ditampilkan pada klien.

Interpreter PHP dalam mengeksekusi kode PHP pada sisi server (server side) berbeda dengan mesin Java yang mengeksekusi program pada sisi klien (client side). PHP merupakan bahasa standar yang digunakan dalam dunia web site. PHP adalah bahasa program yang berbentuk script yang diletakkan didalam server web. Jika kita lihat sejarah, mulanya PHP diciptakan dari ide Rasmus Lerdorf yang membuat sebuah script perl. Script tersebut sebenarnya dimaksudkan untuk digunakan sebagai program untuk dirinya sendiri. Akan tetapi, kemudian dikembangkan lagi sehingga menjadi sebuah bahasa yang disebut "Personal Home Page". Inilah awal mulanya muncul PHP sampai saat ini. Paket inilah yang menjadi cikal bakal PHP. Pada tahun 1995, Rasmus menciptakan PHP/FI versi 2. Pada versi inilah pemrograma dapat menempelkan kode terstruktur di dalam tag HTML. Yang menarik, kode PHP juga dapat berkomunikasi dengan database dan melakukan perhitungan-perhitungan yang kompleks diambil jalan [KAD-05:1]

PHP telah dicipta terutama untuk kegunaan web dan boleh menghubungkan query database dan menggunakan simple task yang boleh diluruskan dengan 3 atau 4 baris kode saja. PHP adalah bahasa programing yang

baru dibangun sekitar tahun 1994/1995. Dan penggunaannya masih baru di Malaysia dan sedang meningkat popular kegunaannya. PHP dapat menukarkan static website yang menggunakan HTML ke dinamic web pages yang berfungsi secara automatic seperti ASP, CGI, dan sebagainya.

PHP sebenarnya merupakan program yang berjalan pada platform LINUX sehingga membuat program ini menjadi freeware. Selanjutnya PHP mengalami perkembangan yakni dibuat dalam versi Windows.

2.5.1 Teknik Penulisan Script PHP

1. Setiap halaman yang mengandung script PHP hanya disimpan dengan EXSTENSI PHP sesuai dengan program PHP yang mendukungnya (nama_file.php, atau nama_file.php3, nama_file.php4).
2. Setiap Script PHP harus didahului dengan pembuka PHP (<?php dan lain-lain) dan kemudian diakhiri dengan penutup (?>).
3. Setiap baris script isi harus didahului dengan pernyataan cetak atau dalam program pascal kita sebut Write atau REM dalam basic, sedangkan dalam PHP pernyataan cetak dibedakan menjadi dua yaitu, Print dan Echo.
4. Setiap akhir baris perintah harus diakhiri dengan titik koma (;).
5. Semua bentuk variable harus diberi tanda string daloar (\$) pada penulisan awalnya.
6. Penulisan Comment atau keterangan didului dengan pembuka /* dan diakhiri */, biasanya kita menggunakan ini untuk memberikan comment yang berbentuk kalimat.

7. Selain menggunakan tanda /*, penulisan command juga dapat menggunakan tanda slash dobel (//). Tanda ini hanya digunakan untuk menuliskan pesan yang hanya berisi satu baris saja.

8. Semua script HTML, yang akan digabungkan dalam script PHP harus dihilangkan tanda petiknya (“”). Sebagai contoh, jika kita memberikan warna background pada halaman kita, maka perintah yang kita gunakan dalam HTML adalah `<BODY bgcolor=#009966>`. Akan tetapi jika diletakkan dalam PHP, maka script tersebut harus diubah menjadi `<BODY bgcolor=#009966>`.

2.5.2 Program Editor

Editor yang digunakan adalah Dreamweaver MX, adalah suatu bentuk editor web yang dibuat oleh makromedia, editor tersebut merupakan editor yang komplit hingga dapat digunakan untuk membuat animasi sederhana yang berbentuk layer, dengan adanya program ini akan memudahkan kita dalam menulis sript dengan format HTML, PHP, ASP maupun bentuk-bentuk yang lainnya, dan seperti program editor-editor yang lain, dreamweaver juga memiliki dua bentuk layer, yaitu bentuk halaman desain dan halaman code, hal ini akan memudahkan kita dalam menambahkan script yang berbasis PHP, ASP, javascript, dan lain-lain.

2.6 Pengenalan MySQL

2.6.1 Pengertian MySQL

MySQL (My Structure Query Language) atau yang biasa dibaca “mai-se-kuel:adalah sebuah program pembuat database yang bersifat open source, artinya siap saja boleh menggunakannya dan tidak dicekal. Saat kita mendengar open

source, kita ingat dengan system operasi handal keturunan Unix, yaitu Linux [KAD-04:99].

MySQL sebenarnya produk yang berjalan pada platform Linux. Karena sifatnya yang open source, dia dapat dijalankan pada semua platform baik Windows maupun Linux. Selain itu, MySQL juga merupakan program pengakses database yang bersifat jaringan sehingga dapat digunakan untuk aplikasi Multi User (Banyak Pengguna). Saat ini database MySQL telah digunakan hampir oleh semua programmer database, apalagi dalam pemrograman web.

Kelebihan lain dari MySQL adalah ia menggunakan bahasa query standart yang dimiliki SQL (Structure Query Language). SQL adalah suatu bahasa permintaan yang terstruktur yang telah distandardkan untuk semua program pengakses database seperti Oracle, Posgres SQL, SQL Server, dan lain-lain.

Sebagai sebuah program penghasil database, MySQL tidak dapat berjalan sendiri tanpa adanya sebuah aplikasi lain (Interface). MySQL dapat didukung oleh hampir semua program aplikasi baik yang open source seperti PHP maupun yang tidak, yang ada pada platform Windows seperti Visual Basic, Delphi, dan lainnya.

Program-program yang digunakan bahasa SQL, antara lain:

- MySQL.
- Posgres SQL.
- Oracle.
- SQL Server 97, 2000.
- Interbase.

Program-program aplikasi pendukung MySQL, antara lain:

- PHP.
- Visual Delphi.
- Visual Basic.
- Cold Fusion, dll.

Hal yang mungkin sulit bagi anda yang menyukai grafis pada MySQL adalah layer program yang berbasis DOS. MySQL memiliki layer utama yang seperti layer DOS yaitu memiliki prompt utama yang disebut `mysql>` sehingga bagi orang yang baru pertama dan belum mengenal perintah DOS akan mengalami kesulitan. Akan tetapi, bagi anda yang baru pertama kali masuk dalam dunia database jangan berkecil hati karena sekarang ada suatu program `dimp` yang dibuat seperti web dan berjalan dibawah server database yang disebut `PHPMyAdmin`.

Dengan menggunakan `PHPMyAdmin` dapat melakukan seluruh Query yang ada sehingga tidak akan jenuh dalam memprogram. Jika Anda terbiasa menggunakan `PHPMyAdmin`, Anda tak akan paham bagaimana teori pembuatan database beserta perintah-perintahnya karena semua Query yang digunakan akan digunakan pada saat kita melakukan Query pada program PHP kita.

Pada pengolahan database `MySQL` pada `Visual Basic` menggunakan `Connection class` yaitu sebuah kelas yang digunakan untuk dapat mengakses database server seperti `MySQL` atau `PgSql`, yang terdiri dari beberapa `Property` dan `Method`. Untuk dapat menggunakan `Connection Class`, kita harus mendeklarasikan sebuah variabel objek dan menunjuk `Connection Class` ke variabel tersebut, kemudian memberi nilai untuk properti-properti yang diperlukan dan menjalankan metod. Openuntuk membuka koneksi ke database server.

Setelah koneksi terbentuk, kita dapat mengolah data pada satu tabel atau lebih dengan menggunakan metode `Exec` (`sql_statement`).

`Exec` (`sql_statement`) Metode `.Exec1` akan mengembalikan hasil eksekusinya ke sebuah variabel object bertipe `Result`, variabel tersebut boleh dikatakan sebagai variabel object yang bertipe `Recordset` di VB.

2.6.2 Fungsi-Fungsi MySQL

Untuk dapat menggunakan semua perintah SQL pada PHP dan Visual Basic, kita memerlukan suatu fungsi MySQL. Fungsi ini berguna untuk mengantarkan perintah SQL pada PHP dan Visual Basic menuju server sehingga perintah tersebut dapat dieksekusi oleh server MySQL.

Tabel 2.5 Fungsi fungsi MySql

Fungsi MySQL	Penggunaan
<code>mysql_change_user()</code>	Menganti user pada koneksi yang sedang aktif
<code>mysql_close()</code>	Digunakan untuk menutup koneksi dengan MySQL
<code>mysql_affected_rows()</code>	mendapatkan jumlah baris dari operasi MySQL sebelumnya
<code>mysql_connect()</code>	Untuk membuka koneksi dengan menggunakan database MySQL Server
<code>mysql_create_db()</code>	Membuat database baru
<code>mysql_data_seek()</code>	Mencari data
<code>mysql_db_name()</code>	Untuk menghasilkan hasil data
<code>mysql_db_query()</code>	Untuk mengantarkan query MySQL
<code>mysql_drop_db()</code>	Untuk menghapus database
<code>mysql_errno()</code>	Untuk menampilkan pesan kesalahan dalam bentuk

	nomer dari server MySQL
mysql_error()	Untuk menampilkan pesan kesalahan dalam bentuk text dari server MySQL
mysql_fetch_array()	Menghasilkan data berupa array dalam bentuk angka dari isi table MySQL
mysql_fetch_assoc()	Menghasilkan data berupa array dalam bentuk field dari isi tabel MySQL
mysql_fetch_field()	Menghasilkan informasi kolom dari hasil yang kemudian dikembalikan sebagai suatu obyek
mysql_fetch_lengths()	Menghasilkan besar file dari hasil tabel MySQL
mysql_fetch_object()	Menghasilkan sebuah baris sebagai suatu obyek
mysql_fetch_row()	Menghasilkan array atau baris dengan menggunakan nama field pada tabel MySQL
mysql_field_name()	Menghasilkan nama field khusus pada database MySQL
mysql_field_len()	Mengembalikan panjang dari field yang sudah ditentukan
mysql_field_seek()	Mengeset hasil pointer dari field yang sudah dispesifikasi
mysql_field_table()	Untuk mendapatkan nama table dalam field
mysql_field_type()	Untuk mendapatkan type di dalam hasil field yang sudah dispesifikasi
mysql_free_result()	Hasil memori yang kosong
mysql_insert_id()	Menghasilkan identitas/id dari operasi INSERT

	sebelumnya
mysql_list_dbs()	List database diketahui di dalam MySQL Server
mysql_list_field()	List hasil dari field MySQL
mysql_list_table()	List tabel dalam MySQL database
mysql_num_field()	Mendapatkan number untuk hasil dalam field
mysql_num_row()	Mendapatkan number untuk hasil dalam row
mysql_pconnect()	Membuka koneksi langsung dengan MySQL
mysql_query()	Dugunakan untuk mengirimkan perintah MySQL
mysql_result()	Untuk menghasilkan data
mysql_select_db()	Digunakan untuk masuk pada database MySQL
mysql_table_name()	Mendapatkan nama tabel

Sumber : [UTD-02].

2.7 Pengertian Hotel

Hotel adalah Bangunan yang bersifat bisnis untuk penginapan atau tempat tinggal sementara dalam waktu tertentu dengan tarif tertentu [BAM-05:25].

2.7.1 Pengertian Reservation

Yang dimaksud dengan Reservation adalah Pemesanan tempat (di restoran, night club, theater, show, tempat duduk dalam pesawat udara, kereta api, intercity bus, atau kamar dalam hotel dan lain sebagainya).

Salah satu cara yang dapat memudahkan konsumen dalam melakukan reservation, yaitu dengan jalan melakukan pesenan atau reservation tempat melalui internet. Dengan demikian pelanggan tidak harus bersusah payah

mendatangi tempat lokasi perusahaan tersebut, tentang spesifikasi produk atau fasilitas yang ditawarkan.

Melalui internet pelanggan dapat melihat informasi tentang fasilitas, type dan harga barang yang diinginkan. Selanjutnya memungkinkan pula untuk melakukan reservation atau pemesanan dari jarak jauh sekalipun.

2.7.2 Departement Resevation

Departemen Resevation adalah departemen yang berfungsi untuk melayani pemesanan kamar di hotel. Departemen ini mempunyai hak untuk menerima atau menolak pemesanan kamar yang dipesan oleh pelanggan [RIS-O6:10].

Adapun tugas dan tanggung jawab departemen reservation adalah sebagai berikut:

1. Menangani pemesanan kamar yang masuk melalui surat, telephone, faxmail, internet, dan pelanggan yang datang secara langsung.
2. Mengontrol reservation dan mengisi format reservation serta slipnya berdasarkan pesanan kamar yang diterima.
3. Mengontrol reservation chart sheer dan suite chart sheer tiap hari.
4. Menyimpan semua correspondence yang berkenaan dengan pesanan kamar sesuai dengan cara penyimpanannya (file) yang telah ditentukan.
5. Menjaga agar daftar/buku special company rate selalu up to date.
6. Menghubungi perusahaan/ pelanggan yang memesan kamar tetapi tidak datang tanpa pemberitahuan.
7. Menyimpan kartu registrasi (R/C) yang check in dan check out.
8. Menyimpan group information untuk dikirim ke Departemen lain.
9. Menyiapkan Expected Arrival list untuk ke esokan harinya sesuai dengan reservation yang ada.

10. Menyiapkan daily statistic.

Setelah Melakukan pemesanan kamar, customer akan mendatangi hotel sesuai dengan tanggal dan hari yang telah ditentukan. Pelanggan akan melakukan konfirmasi ulang mengenai pemesanannya kepada bagian front office, dengan memperlihatkan bukti pemesanan kamar.

Front office merupakan bagian yang paling penting dalam pengoperasian sebuah hotel. Front office hotel menurut sudut pandang tamu merupakan cermin keseluruhan dari hotel karena segala pusat kegiatan hotel mula-mula di mulai dari bagian ini, yakni mulai dijalankan peranannya sebagai penghubung antar tamu dan hotel yaitu mulai tamu datang, sampai tamu menginap dan akhirnya meninggalkan hotel selalu berhubungan dengan bagian ini. Dengan demikian tugas utama dari front office adalah memberikan pelayanan yang sebaik mungkin dalam meningkatkan penjualan kamar hotel. Dan salah satu fungsi dari front office adalah menerima segala sesuatu pemesanan kamar hotel, baik untuk keperluan perseorangan ataupun kelompok.

Berikut ini adalah beberapa seksi yang ada didalam front office hotel.

1. Pemesanan Kamar (Reservation)

Yaitu segala kegiatan yang menyangkut segala sesuatu atas pemesanan kamar hotel, baik untuk keperluan perseorangan maupun kelompok.

2. Penerimaan Tamu (Reservation Regeistrtion)

Yaitu seluruh kegiatan yang menyangkut tata cara dalam menerima tamu yang akan menginap serta mendistribusikan informasi mengenai kedatangan tamu tersebut kepada bagian-bagian lain di hotel.

3. Information and Keys

Penangan informasi, surat-surat dan kunci kamar hotel dan lain-lain dari dan untuk hotel serta kepentingan tamu, baik mengenai informasi hotel itu sendiri atau bersifat umum.

4. Pelayanan Telepon (Telephone Service)

Yaitu kegiatan yang menyangkut seluruh pelayanan telepon dari dan ke hotel baik untuk kepentingan hotel maupun tamu.

5. Uniform Service Constege

Yaitu seksi yang memberikan pelayanan “service” kepada tamu hotel sepanjang saat dibutuhkan.

6. Front Office Center

Pelayanan pembayaran rekening pada saat tamu hotel akan menyelesaikan pembayaran rekening hotel.

2.7.3 Alur Sistem Reservasi

1. Reservation Clerk menerima reservasi dari tamu.
2. Reservation Clerk menanyakan nama tamu yang akan menginap, tanggal berapa tiba, dan tanggal berangkat, serta jumlah dan tipe kamar.
3. Resevation Clerk memeriksa Reservation Chart untuk memenuhi kondisi kamar.
4. Apabila permintaan telah terpenuhi, Reservation Clerk menyampaikan informasi tipe kamar, harga kamar dan apabila diperlukan, dimintakan uang muka (deposit).
5. Reservation Clerk mengisi formulir pemesanan kamar yang lengkap.
6. Reservation Clerk kembali menyampaikan seluruh pemesanan kamar untuk menghindari terjadinya kesalahan.

2.7.4 Prosedure Reservasi

1. Pelanggan mengisi data yang lengkap pada formulir reservasi.
2. Pada waktu cek in pelanggan membawa identitas yang sama dengan data yang ada pada saat reservasi dan tidak diwakilkan.
3. Dimintakan membayar uang muka (DP) 50% dari harga kamar.
4. Membawa slip bukti pembayaran uang muka pada saat cek in, itu ditujukan pada pelanggan yang datang langsung ke hotel untuk reservasi atau slip bukti transfer dari bank, ditujukan pelanggan dari luar kota yang melakukan reservasi lewat telphon.
5. Jika slip bukti pembayaran uang muka atau slip bukti transfer dari bank tidak dibawa maka uang muka akan hilang pada saat customer melakukan cek in.
6. Bila ada pembatalan atau perubahan pesanan kamar pelanggan harus segera mengkonfirmasi secepatnya kepada pihak hotel.
7. Batas waktu yang diberikan pihak hotel kepada pelanggan yang melakukan reservasi untuk melakukan konfirmasi lebih lanjut maksimal 3 jam dari waktu yang ditetapkan selebihnya kamar akan diberikan pelanggan lain.

2.8 Gambaran Umum Perusahaan

2.8.1 Sejarah Hotel Gajah Mada

Hotel Gajah Mada bergerak dibidang jasa perhotelan. Sebelum menjadi Hotel Gajah Mada yang ada sekarang ini, Hotel Gajah Mada memiliki pergantian nama dan pemilik. Pada tahun 1916 saat belanda masuk ke Indonesia, Hotel

Gajah Mada dikelola oleh biro arsitektur AIA dengan nama Palace Hotel. Bangsa Jepang masuk pada tahun 1942 dan menggantikan nama pengelolanya, nama Palace Hotel diganti dengan nama Azzuna Hotel.

Setelah masuk pada zaman kemerdekaan, maka Hotel Gajah Mada ini berganti nama lagi dan dikelola oleh warga keturunan cina (1945). Oleh pemiliknya, Azuma Hotel diganti dengan nama menjadi semula pada saat dikelola Pemerintah Belanda yaitu Palace Hotel. Pada tahun 1962 terjadi pengalihan pemilik lagi dan pemiliknya orang pribumi asli dan namanya berganti lagi namanya menjadi Gajah Mada Hotel sampai sekarang. Pemilik ini adalah Bpk. SJACHRAN HOESANI (alm). Adapun pemegang perusahaan Hotel Gajah Mada adalah keluarga Bpk. SJACHRAN HOESNI (alm) sampai saat ini jabatan direktur dipercayakan kepada bapak ROBENSJAH SJACHRAN, SH dan sebagai dewan komisaris adalah keluarga Bpk. H. SJACHRAN HOESNI (alm) atau putra putrinya.

Pembenahan dilakukan disegala bidang terutama pembenahan-pembenahan fisik banguyan secara bertahap dan dilengkapi oleh berbagai fasilitas yang memadai dan sesuai dengan perkembang zaman. Hal tersebut dilakukan agar dapat memberi rasa aman, nyaman, dan konsumen merasa kerasan sebagai tempat untuk istirahat, sebagai tempat singgah sementara waktu atau untuk keperluan untuk lainnya. Pihak Hotel Gajah Mada selalu memperhatikan tentang pemenuhan seluruh kebutuhan dan service yang baik untuk keputusan pelayanan konsumen. Berbnagai layanan disuguihkan secara professional dan selalu menjaga kredibilitas dan image yang baik.

Untuk mendapat pembenahan yang sah sebagai usaha, maka pimpinan mendaftarkan perusahaan ini pada departemen perdagangan dan pariwisata:

- a. Departemen perdagangan (keputusan menteri perdagangan)

No :1587/kp/13/3/84

Surat ijin perdagangan

No :1806/ml/84/BR

Tanggal :14 maret 1984

- b. Departemen perdagangan

No :1308560166

Perihal :tanda daftar perusahaan

Tanggal :19 februari 1992

Hotel Gajah Mada didirikan dengan fasilitas akomodasi sebanyak 80 kamar 5 macam tipe kamar, dimana tipe tersebut sudah diklafikasikan oleh manajemen Hotel Gajah Mada antara lain:

- a. Ekonomi

Kamar yang dilengkapi dengan kamar mandi dan shower (air panas dan dingin), serta kipas angin.

- b. Tipe Suite

Kamar yang dilengkapi kamar mandi dan shower (air panas dan dingin), AC, TV, dan Telephone IDD.

- c. Tipe Standart

Kamar yang dilengkapi dengan kamar mandi dan shower (air panas dan dingin), AC, Walto wall carpeted, program TV parabola dengan 3 chanel, musik dan telephone IDD.

d. Tipe Superwin

Kamar yang difasilitasi kamar mandi dan shower (panas dan dingin), AC, Wal to wall carpeted, program TV parabola dengan 3 chanel, musik, Telephone IDD, dan lemari es.

e. Tipe duluxe

Kamar yang difasilitasi dengan kamar mandi dan shower (air dingin dan panas), AC, Wal to wall carpeted, program TV parabola dengan 3 chanel, musik, Telephone IDD system, dan Lemari es

Tabel 2.6 Perincian daya tampung hunian kamar Hotel Gajah Mada

Jenis kamar	Jumlah kamar	Daya tampung kamar	Daya tampung per bulan
Deluxe	14	14	420
Superwin	16	16	480
Standart	18	18	540
Suite	16	16	480
Ekonomi	14	16	480

Sumber : [Hotel Gajah Mada]

Kelas	Tarif
-------	-------

Tabel 2.7 kamar masing- kelas	Deluxe	Rp. 200.000,00	Tarif pada masing
	Superwin	Rp. 165.000,00	
	Standart	Rp. 150.000,00	
	Suite	Rp. 55.000,00	
	Ekonomi	Rp. 37.500,00	

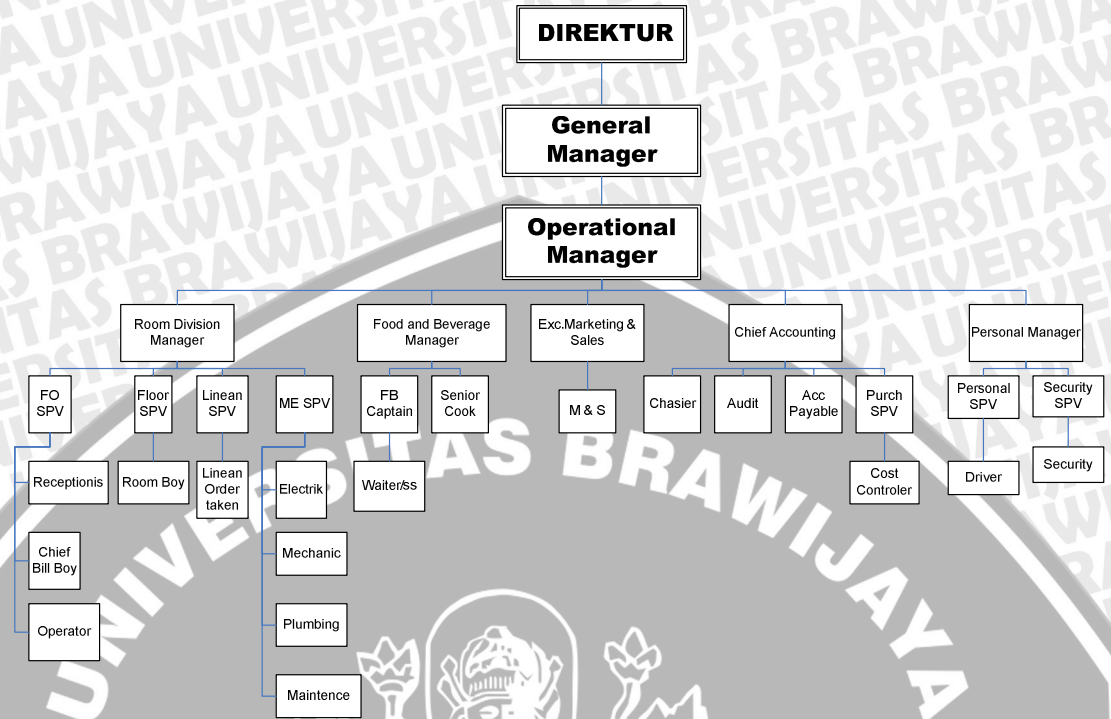
Sumber :

[Hotel Gajah Mada]

2.8.2 Struktur Organisasi

Struktur organisasi merupakan kerangka kerja yang menunjukkan segenap tugas untuk mencapai tujuan organisasi, hubungan antar bagian, wewenang serta tanggung jawab tiap-tiap anggota organisasi yang memikul setiap tugas dan pekerjaannya. Dengan adanya struktur organisasi ini diharapkan para personil dapat berkerja dengan teratur dan produktif.

Bila dilihat dari susunan organisasi yang mengatur tugas-tugas dan tanggung jawab yang ada dalam organisasi Hotel Gajah Mada, tipe organisasi yang digunakan adalah organisasi garis. Penentuan dalam sistem organisasi garis ini dimaksudkan agar keputusan dapat dilaksanakan dengan tepat dan biaya dapat dihemat. Sebab pengawasan dari berbagai kegiatan operasional hanya dilakukan oleh seorang saja. Untuk lebih jelasnya maka struktur organisasi yang terdapat pada Hotel Gajah Mada pada gambar sebagai berikut:



Gambar 2.1 Struktur Organisasi Perusahaan

Sumber : [Hotel gajah mada]

Job Discription

Berikut ini adalah uraian tugas dan tanggung jawab masing-masing jabatan sebagai berikut:

a. General Manager

- Mengadakan perencanaan umum dalam semua bidang.
- Menentukan kebijakan perusahaan baik intern maupun ekstern dan memimpin aktivitas perusahaan secara keseluruhan.
- Mengkordinasikan dan mengawasi seluruh kegiatan operasional dari masing-masing departemen.

b. Front office Departement

- Memeriksa pembuatan dan pengisian form-form yang telah dilakukan oleh receptionis.
- Membuat atau mengisi laporan harian, status kamar, data pribadi tamu, termasuk mengecek identitas tamu.
- Menkoordinir dan bertanggung jawab atas kebersihan, keindahan, keutuhan gedung dan bangunan yang ada disekitar front office.

c. Housekeeping Departement

- Bertanggung jawab atas kebersihan, kerapihan kamar-kamar maupun ruangan yang ada dihotel baik disekitar area hotel maupun luar area.
- Serta pencucian dan dry cleaning pakaian-pakaian tamu dan klien hotel.

d. Engineering Departement.

- Bertanggung jawab atas kegiatan mekanik dan elektrik yang ada di Hotel Gajah Mada baik perbaikan maupun kerusakan maupun maintenance.

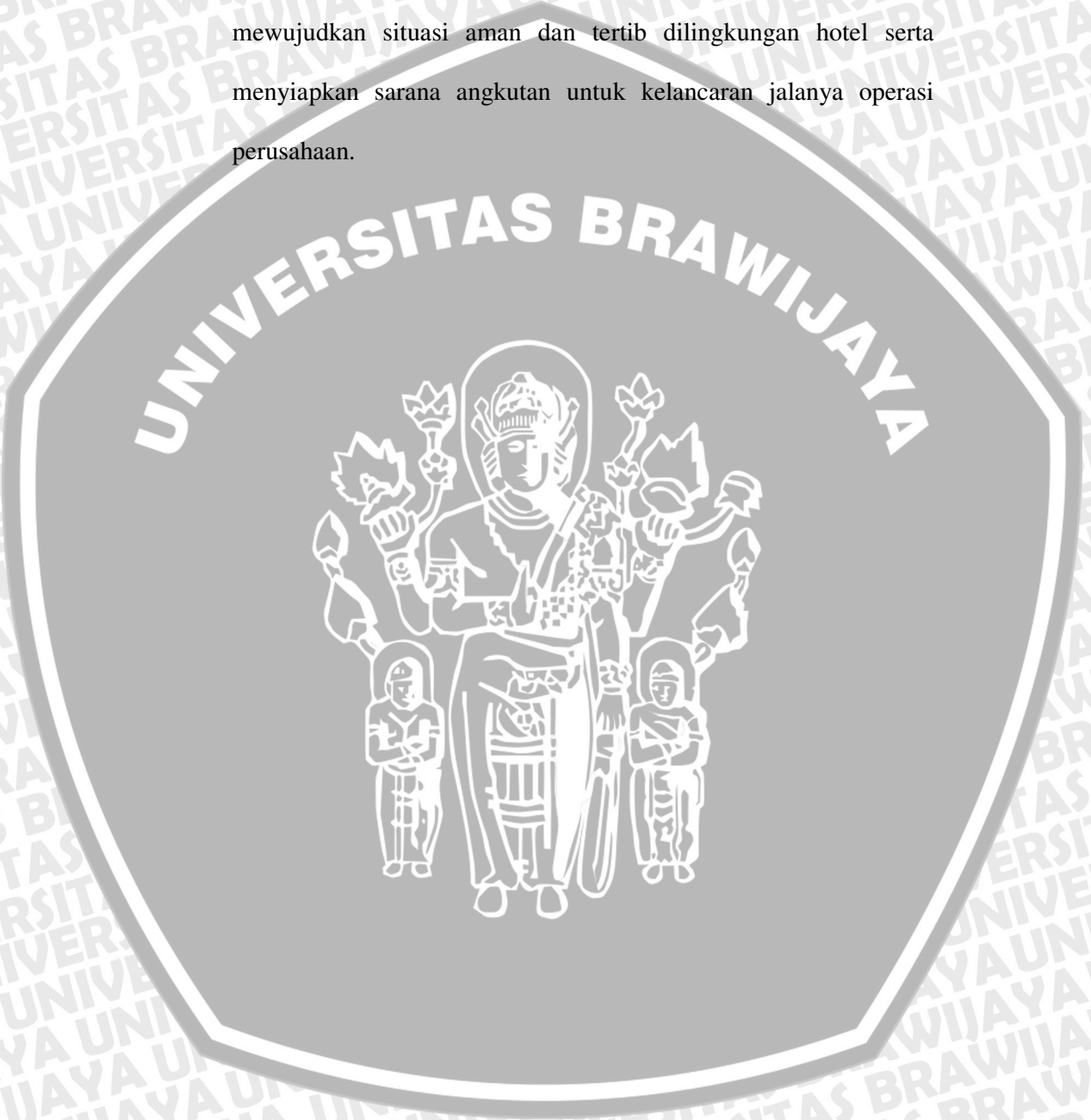
e. Marketing Departement

- Menadakan kerjasama dengan pihak tertentu untuk promosi baik didalam maupun diluar negeri.
- Mengatur kebijaksanaan pemasaran berkaitan dengan produk, harga, promosi dan saluran distribusi yang diterapkan.

f. Personnel Departement.

- Mengawasi dan malayani pelaksanaan hak dan kewajiban pegawai.

- Menyelenggarakan pembinaan dan training untuk para karyawan agar gairah dan peran sertanya lebih meningkat.
- Melakukan koordinasi dalam pelaksanaan tugas operasi untuk mewujudkan situasi aman dan tertib dilingkungan hotel serta menyiapkan sarana angkutan untuk kelancaran jalannya operasi perusahaan.



BAB III METODE PENELITIAN

Pada tahap ini dijelaskan mengenai langkah-langkah yang akan dilakukan untuk mengimplementasikan aplikasi perangkat lunak yang akan dibuat. Dalam mengimplementasikan perangkat lunak tersebut diperlukan data, informasi dan metode dalam penyelesaiannya. Langkah – langkah yang perlu dilakukan untuk dapat merealisasikan perangkat lunak yang akan dibuat adalah sebagai berikut :

- Observasi
- Studi literatur
- Perancangan
- Implementasi
- Pengujian dan Analisis
- Pengambilan Kesimpulan
- Penulisan laporan

3.1 Observasi

Observasi dilakukan dengan pengumpulan informasi mengenai aplikasi sejenis yang sudah pernah ada baik melalui internet maupun buku. Selain itu observasi dilakukan dengan pengamatan secara langsung pada obyek penelitian untuk mendapatkan informasi tambahan yang berguna.



3.2 Studi Literatur

Studi literatur yang dilakukan bertujuan untuk mengkaji hal-hal yang berhubungan dengan teori-teori yang mendukung dalam perancangan dan perealisasiian aplikasi, diantaranya adalah:

- Pengertian Hotel.
- Pengertian Reservasi.
- Sistem Informasi.
- Basis Data
- *Structured Query Language (SQL)*
- *MySQL*
- Teori Perancangan Perangkat Lunak
- Internet
- Pemrograman Web

3.3 Perancangan

Perancangan perangkat lunak terdiri dari analisis kebutuhan perangkat lunak, perancangan sistem, perancangan basis data, perancangan proses, dan perancangan antarmuka. Perancangan didasarkan pada teori-teori yang nantinya akan diaplikasikan pada sistem. Perancangan perangkat lunak dilakukan pada tiap-tiap blok aplikasi sistem untuk mempermudah analisis terstruktur, perancangan, dan pemrograman.

3.4 Implementasi

Implementasi perangkat lunak terdiri dari implementasi basis data dan implementasi antarmuka perangkat lunak. Implementasi perangkat lunak dilakukan berdasarkan pada perancangan perangkat lunak. Melakukan implementasi dari hasil perancangan yang telah didefinisikan sebelumnya

kedalam bentuk kode (coding) dengan bahasa pemrograman PHP serta MySQL sebagai sarana untuk penyimpanan databasenya.

3.5 Pengujian dan Analisis

Melakukan pengujian perangkat lunak yang telah dibuat yang ditujukan untuk menemukan kesalahan, sehingga perangkat lunak akan terbebas dari segala kesalahan pada saat dieksekusi oleh pengguna. Pengujian meliputi pengujian basis data, pengujian algoritma dengan teknik *white box testing* dan *black box testing*, pengujian sistem secara *online* setelah dilakukan pengujian maka dilakukan analisis dari keseluruhan sistem.

3.6 Pengambilan Kesimpulan

Pada tahap ini dilakukan pengambilan kesimpulan dari aplikasi yang telah dibuat. Pengambilan kesimpulan dilakukan setelah semua tahapan perancangan dan pengujian sistem aplikasi telah selesai dilakukan dan didasarkan pada kesesuaian antara teori dan praktek. Kesimpulan ini merupakan informasi akhir dari perancangan aplikasi yang berisi mengenai berhasil atau tidaknya aplikasi tersebut dijalankan.

Tahap terakhir dari penulisan adalah saran yang dimaksudkan untuk memperbaiki kesalahan-kesalahan yang terjadi serta menyempurnakan penulisan.

3.7 Penulisan Laporan

Melakukan dokumentasi skripsi yang sudah dibuat, sehingga dokumentasi ini dapat berguna dan diharapkan dapat dikembangkan menjadi lebih baik.

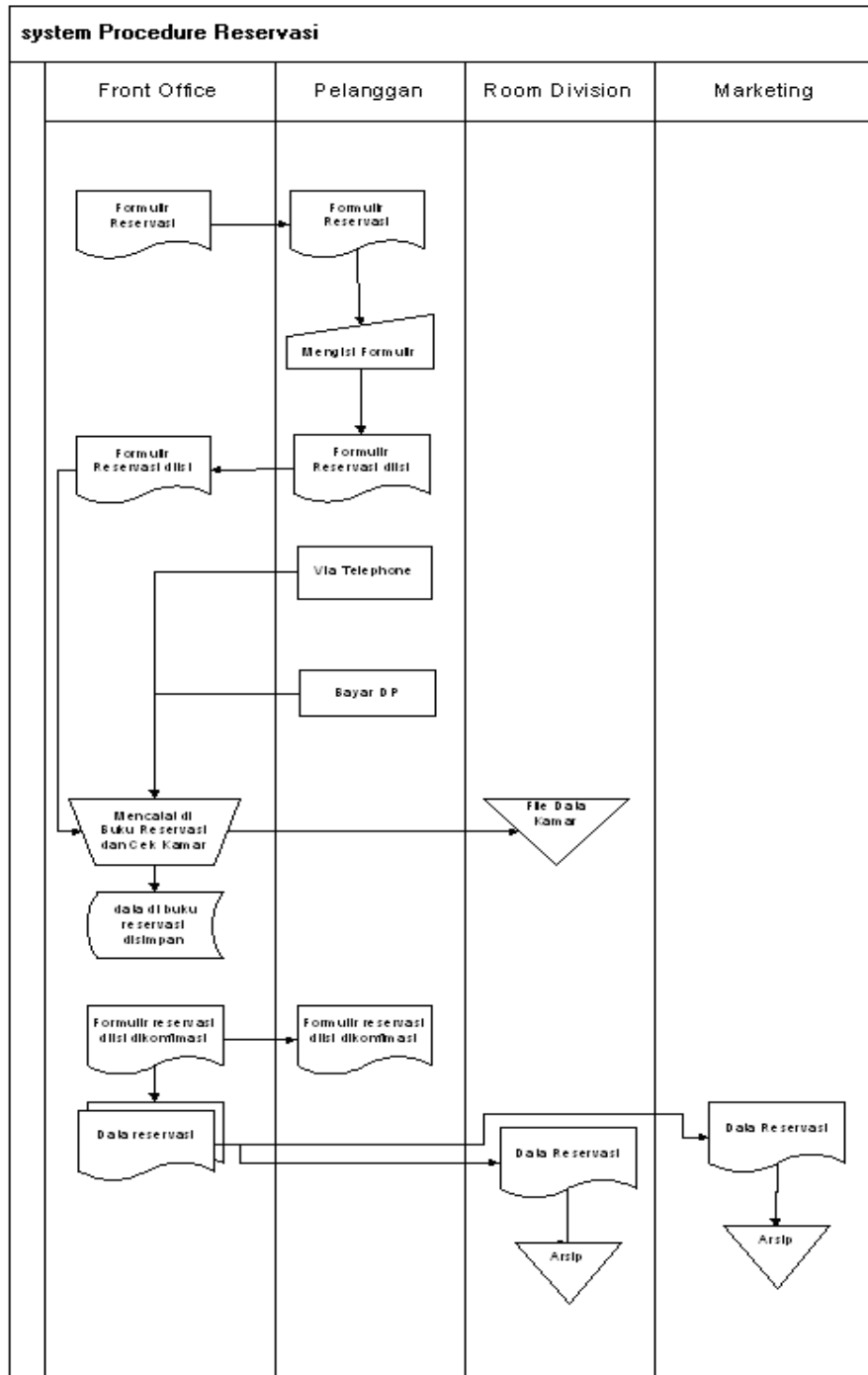
BAB IV PERANCANGAN

Setelah melakukan penelitian pada Hotel Gajah Mada tentang proses reservasi, walaupun secara garis besar sistem yang digunakan sudah sangat bagus, namun masih dapat dilihat beberapa kelemahan yang ada dalam proses tersebut, yang nantinya mungkin dapat berpengaruh pada tingkat kepuasan pelanggan.

Dalam melaksanakan operasionalnya, Hotel Gajah Mada memerlukan standarisasi teknologi informasi dan pola kerja yang baik, guna menghadapi persaingan yang ketat dan guna meningkatkan pangsa pasar. Berdasarkan bab II yang merupakan prosedur reservasi, dapat disimpulkan bahwa, customer melakukan reservasi dengan cara datang langsung atau melalui via telephon maupun via email. Pada waktu check in pelanggan harus membawa identitas beserta slip bukti pembayaran atau slip bukti transfer uang muka, jika tidak dibawa maka uang muka akan hilang. Bila ada perubahan atau pembatalan pelanggan harus mengkonfirmasi secepatnya, batas waktu yang diberikan hotel untuk konfirmasi ulang sehari sebelum hari pemesanan.

Sesuai dengan keinginan Hotel Gajah Mada, dimana ingin meningkatkan kualitas pelayanan pada pelanggan, maka kenyamanan dan kepuasan customer adalah yang utama, dengan demikian mungkin Aplikasi reservasi secara online guna mempermudah proses pemesanan kamar dapat menjadi salah satu solusi untuk meningkatkan kepuasan pelanggan, dan juga untuk memperluas pangsa pasar Hotel Gajah Mada hingga keluar daerah Ponorogo.

Tabel 4.1. Tabel System Procedure Diagram



Sumber : [Perancangan]

Keterangan:

1. Front Office memberikan slip reservasi atau menerima telephone dari pelanggan yang akan melakukan reservasi.
2. Pelanggan mengisi formulir reservasi dengan langkap sedangkan pelanggan yang melalui via telephone akan dicatat oleh Front Office dan akan dikonfirmasi ulang selama 24 jam.
3. Front office mengecek file data kamar berdasarkan waktu pemesanan pada bagian Room Division serta mencatat data reservasi pada buku reservasi.
4. Setelah ada kamar Front office menyimpan data reservasi, Front office mengkonfirmasi ulang ke pelanggan. Apabila permintaan sudah dipenuhi maka Front office menyampaikan harga kamar, type kamar, dan apabila perlu dimintakan uang muka (deposit).
5. Front office melakukan pengisian formulir reservasi dengan langkap agar tidak terjadi kesalahan catatan.
6. Pelanggan membayar uang muka baik tunai langsung atau transfer bank kalau pemesan dari luar kota.
7. Bagian manager dan bagian room division meminta data reservasi.

4.1 Daftar Kerja Reservasi.

Tabel 4.2 Tabel Daftar Kerja Reservasi

Bagian	Daftar Kerja
1. Front Office	<ul style="list-style-type: none"> • Melayani semua transaksi hotel serta memberikan informasi tentang hotel. • Menangani pemesanan kamar yang

	<p>masuk melalui surat, telephone, dan pelanggan datang langsung maupun via email.</p> <ul style="list-style-type: none"> • Mengontrol reservasi dan mengisi format atau list reservation serta slipnya berdasarkan pemesanan yang diterima. • Menjaga agar list atau buku reservasi selalu up to date. • Mengisi serta menyerahkan form reservasi ke bagian marketing dan room division. • Menghubungi perusahaan atau pemesan yang memesan kamar tetapi tidak datang tanpa pemberitahuan. • Menyimpan data reservasi.
<p>2. Room Division</p>	<ul style="list-style-type: none"> • Menyiapkan kamar yang akan dipesan. • Menyiapkan data kamar secara up to date. • Mengontrol reservasi chart atau file data kamar untuk memenuhi kondisi kamar. • Meminta data reservasi dari front office.

	<ul style="list-style-type: none"> • Menjaga dan mengkoordinir atas kebersihan semua kamar di hotel.
3. Manager	<ul style="list-style-type: none"> • Mengadakan kerjasama dengan pihak tertentu untuk promrsi baik didalam maupun diluar negeri. • Mengontrol dan mengatur kebijaksanaan pemasaran berkaitan dengan produk harga, promosi dan saluran distribusi yang diterapkan. • Meminta data resevasi pada bagian front office untuk dijadikan laporan sebagai pertimbangan dalam meningkatkan pemasaran.

Sumber : [Perancangan]

4.2 Analisa Masalah

Setelah dilakukan penelitian mengenai sistem reservasi seperti diatas ternyata memiliki beberapa kelemahan, yaitu :

1. Untuk proses validasi data reservasi diperlukan waktu yang terlalu lama karena batas waktu maksimal yang diberikan pelanggan untuk konfirmasi yaitu sehari sebelum tanggal cek in.
2. Waktu yang lama untuk melakukan reservasi jika pelanggan datang langsung ke Hotel Gajah Mada.

3. Bagi pelanggan yang melalui via telephone tidak bisa melihat secara langsung fasilitas dan akomodasi yang didapat nantinya dikarenakan keterbatasan waktu dan wilayah.
4. Pemesanan hanya terbatas oleh wilayah karena promosi di Hotel Gajah Mada hanya melalui spanduk disekitar wilayah Ponorogo.

4.3 Akibat yang ditimbulkan

Setelah dilakukan penelitian mengenai order pada sistem penjualan diatas ternyata memiliki beberapa akibat yang ditimbulkan, yaitu;

1. Pelanggan merasa kurang puas karena kurangnya informasi tentang Kamar dan fasilitas yang ditawarkan dikarenakan keterbatasan waktu.
2. Proses transaksi yang cukup lama apabila pelanggan datang langsung. Pemesanan tidak bisa meningkat, karena keterbatasan wilayah.

4.4 Pemecahan Masalah

Dengan semakin majunya teknologi maka proses pemesanan kamar bisa dilaksanakan dengan sistem terkomputerisasi dan bisa dilakukan online yang menggunakan PHP. Hal tersebut dipilih karena keuntungan yang diperoleh juga semakin besar yaitu:

1. Tingkat kepuasan pelanggan naik sehingga pelanggan tidak ingin pindah ke hotel lain.
2. Jaringan pemasaran menjadi lebih luas sehingga pelanggan pun menjadi banyak.

Adapun prosedur sistem pemesanan yang digunakan adalah sebagai berikut :

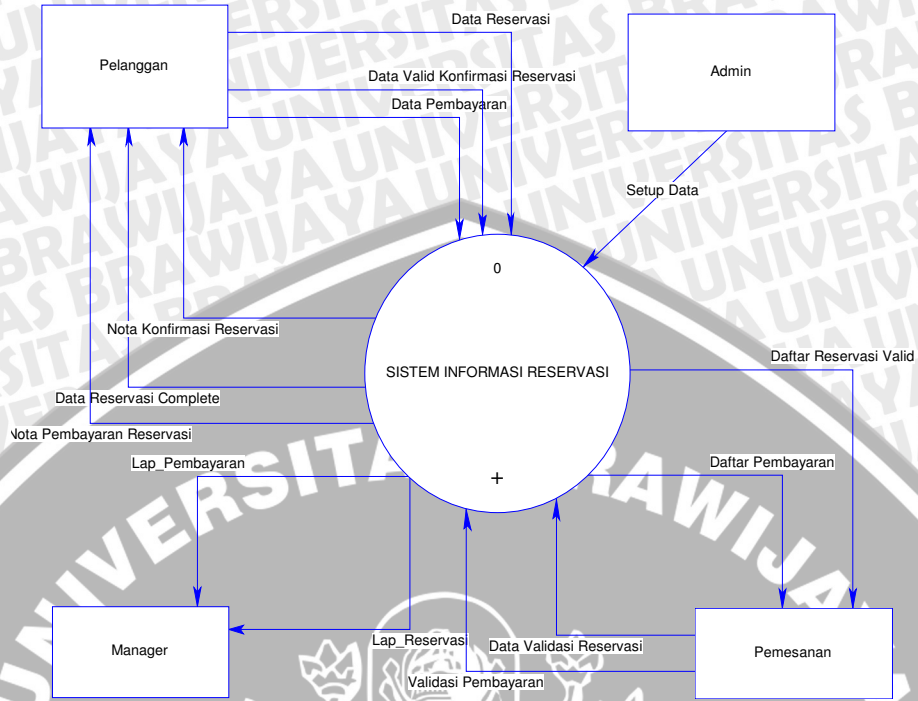
1. Pelanggan membuka web site hotel melalui warnet, internet rumahan, atau datang langsung.
2. Pelanggan melihat data kamar yang tersedia kemudian melakukan pemesanan kamar.
3. Pelanggan mengisi foormulir reservasi dengan lengkap beserta jenis kamar yang diinginkan dan akan langsung dikirim validasi reservasi via email.
4. Apabila permintaan sudah dipenuhi maka pelanggan melakukan pembayaran harga kamar yang diinginkan.

4.5 Desain Proses

Untuk menggambarkan secara rinci proses yang ada serta jaringan antar proses yang berhubungan satu dengan yang lain yang menunjukkan dari dan ke mana suatu data mengalir beserta penyimpanannya, secara keseluruhan dimodelkan dalam beberapa tahapan berikut.

4.5.1 Contex Diagram

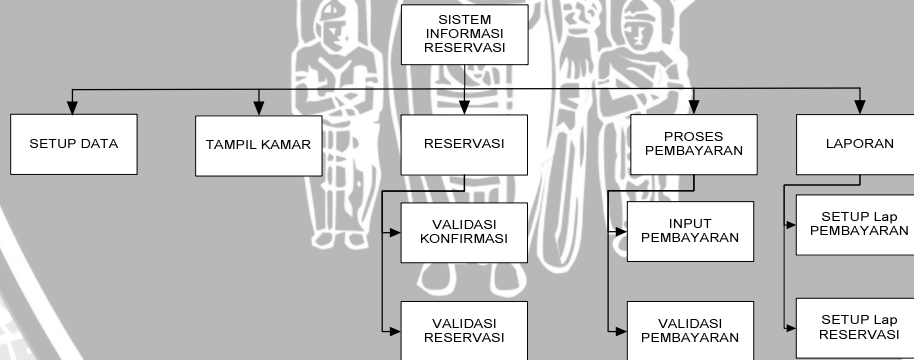
Pengolahan semua data dilakukan oleh bidang layanan dimana nanti semua data akan dilaporkan keManager hotel . Untuk lebih jelasnya lihat gambar 4.1 Context diagram.



Gambar 4.1 Context Diagram

Sumber : [Perancangan]

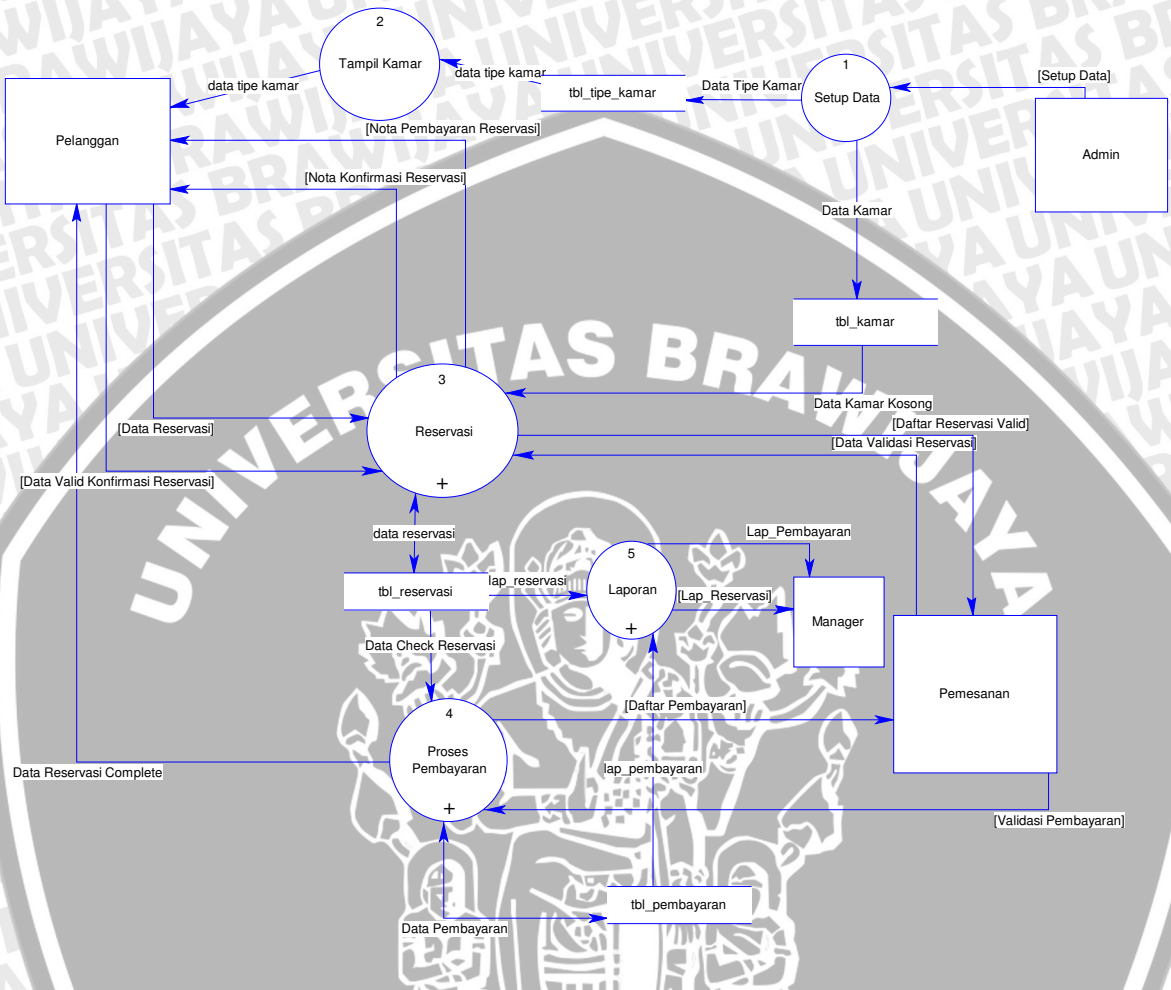
4.5.2 Dekomposisi Fungsional



Gambar 4.2 Dekomposisi Fungsional

Sumber : [Perancangan]

4.5.3 Data Flow Diagram Level 1 Reservasi



Gambar 4.3 Data Flow Diagram level 1

Sumber : [Perancangan]

Gambar 4.3 merupakan DFD level 1 dari system informasi dari data reservasi dimana sistem tersebut dibagi menjadi 5 proses, yaitu proses tampil kamar, proses reservasi, proses pembayaran, proses setup data dan proses laporan. Dari diagram tersebut, maka dapat dijabarkan lebih detail proses-proses tersebut sebagai berikut:

1. Proses tampilan kamar

Proses ini merupakan proses yang ditampilkan untuk mengetahui fasilitas hotel kepada user.

Proses ini mempunyai data masukan:

- Data tipe kamar.

Proses ini mempunyai data keluaran :

- Data tipe kamar.

2. Proses reservasi

Proses ini adalah proses data pemesanan kamar.

Proses ini mempunyai data masukan:

- Data kamar kosong.
- Data validasi reservasi.
- Data reservasi.
- Data valid konfirmasi reservasi.

Proses ini mempunyai data keluaran :

- Nota konfirmasi Reservasi.
- Nota pembayaran Reservasi.

3. Proses pembayaran

Proses ini merupakan proses data pembayaran kamar.

Proses ini mempunyai data masukan:

- Laporan pembayaran.
- Data pembayaran.
- Data reservasi complete.
- Data check reservasi.

Proses ini mempunyai data keluaran :

- Data pembayaran.

4. Proses setup data

Proses ini merupakan proses penyimpanan data reservasi yang dimasukan oleh admin.

Proses ini mempunyai data masukan:

- Setup data..

Proses ini mempunyai data keluaran :

- Data kamar.
- Data tipe kamar.

5. Proses laporan

Proses ini berisi data laporan hasil reservasi.

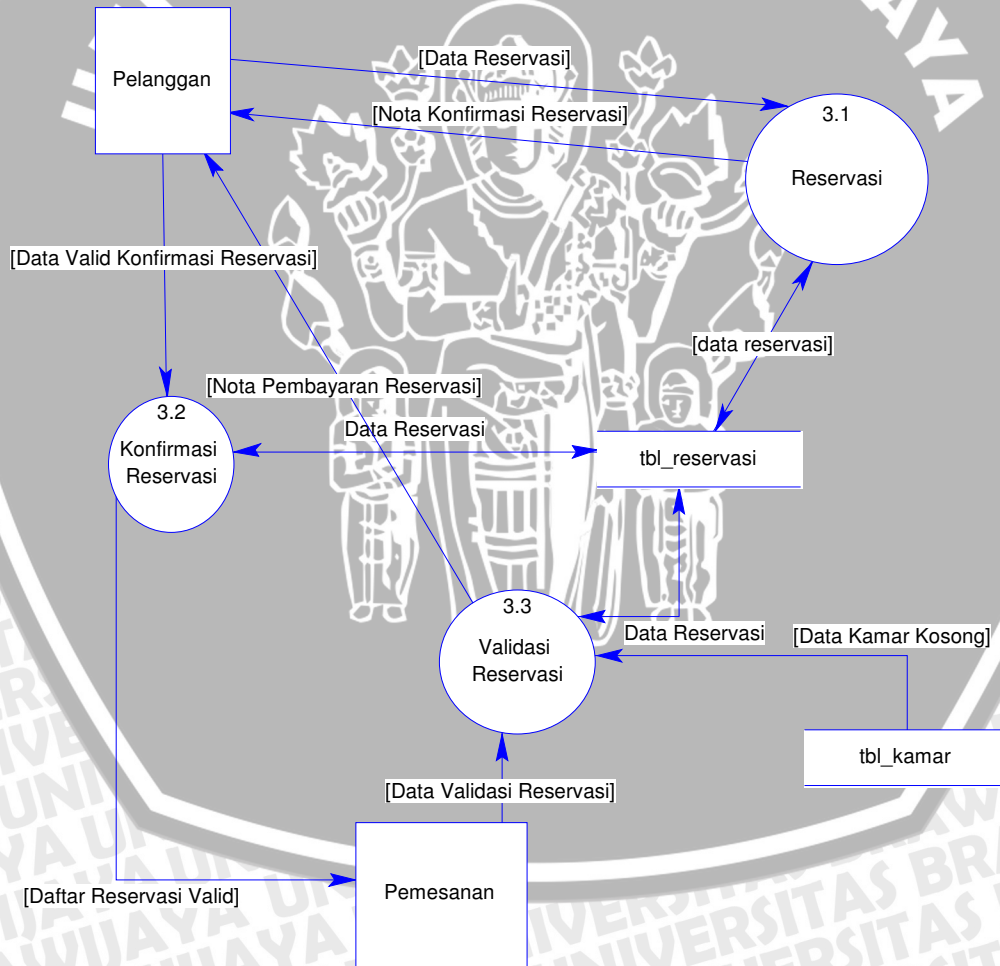
Proses ini mempunyai data masukan:

- Laporan reservasi.
- Laporan pembayaran.

Proses ini mempunyai data keluaran :

- Laporan reservasi.
- Laporan pembayaran.

4.5.4 Data Flow Diagram Level 2 Reservasi



Gambar 4.4. Data Flow Diagram level 2Reservasi

Sumber : [Perancangan]

Gambar 4.4 merupakan DFD level 2 dari system informasi dari data reservasi dimana sistem tersebut dibagi menjadi 3 proses, yaitu konfirmasi reservasi, proses validasi reservasi, dan proses reservasi. Dari diagram tersebut, maka dapat dijabarkan lebih detail proses-proses tersebut sebagai berikut:

1. Proses Konfirmasi Reservasi

Proses ini berisi data reservasi pemesanan kamar.

Proses ini mempunyai data masukan :

- Data valid konfirmasi reservasi.
- Data reservasi.

2. Proses validasi reservasi

Proses ini berisi data pengesahan kamar oleh user dan pihak hotel.

Proses ini mempunyai data masukan:

- Data validasi reservasi.
- Data reservasi.

Proses ini mempunyai data :

- Nota pembayaran reservasi.

3. Proses Reservasi

Proses ini berisi proses data pemesanan kamar.

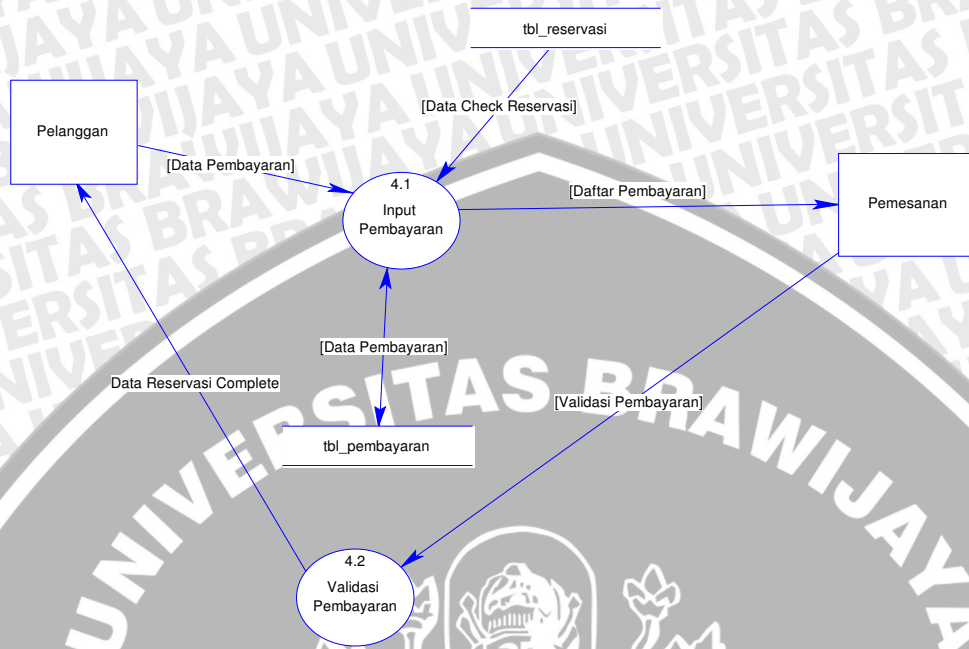
Proses ini mempunyai data masukan:

- Data reservasi.

Proses ini mempunyai data keluaran:

- Nota konfirmasi reservasi.
- Data reservasi.

4.5.5 Data Flow Diagram Level 2 Pembayaran



Gambar 4.5. Data Flow Diagram level 2 Pembayaran

Sumber : [Perancangan]

Gambar 4.5 merupakan DFD level 2 dari system informasi pembayaran dimana sistem tersebut dibagi menjadi 2, yaitu input pembayaran dan validasi pembayaran. Dari diagram tersebut, maka dapat dijabarkan lebih detail proses-proses tersebut sebagai berikut:

1. Proses input pembayaran

Proses ini berisi data pembayaran validasi pemesanan kamar

Proses ini mempunyai data masukan:

- Data pembayaran.
- Data check reservasi.

Proses ini mempunyai data keluaran:

- Daftar pembayaran.

2. proses validasi

proses ini berisi data proses pembayaran pemesanan kamar.

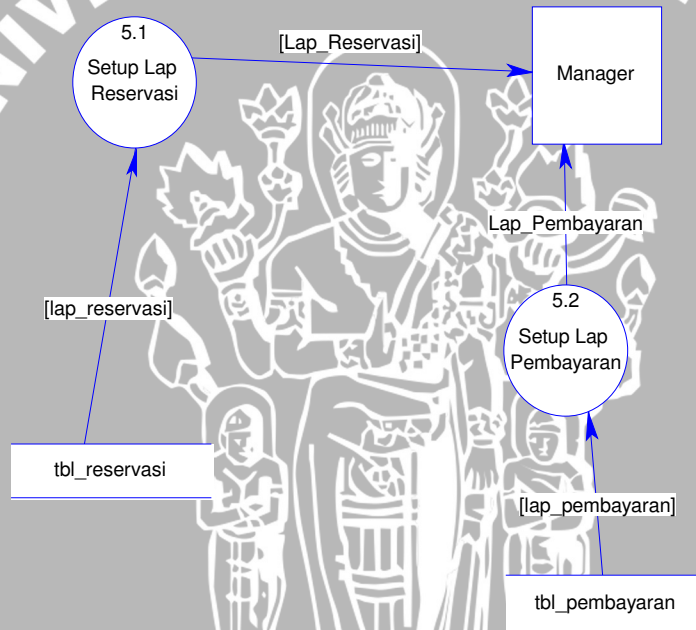
Proses ini mempunyai data masukan:

- Validasi pembayaran

Proses ini mempunyai data keluaran:

- Data reservasi complete

4.5.6 Data Flow Diagram Level 2 Laporan



Gambar 4.6. Data Flow Diagram level 2 Laporan

Sumber : [Perancangan]

Gambar 4.6 merupakan DFD level 2 dari system informasi laporan reservasi pembayaran kamar, dimana sistem tersebut dibagi menjadi 2, yaitu:

1. Setup laporan reservasi

Proses ini berisi data laporan pengaturan reservasi pemesanan kamar.

Proses ini mempunyai data masukan:

- Laporan reservasi.

Proses ini mempunyai data keluaran:

- Laporan reservasi.

2. Setup laporan pembayaran

Proses ini berisi data laporan pengaturan pembayaran pemesanan kamar

Proses ini mempunyai data masukan:

- Laporan pembayaran

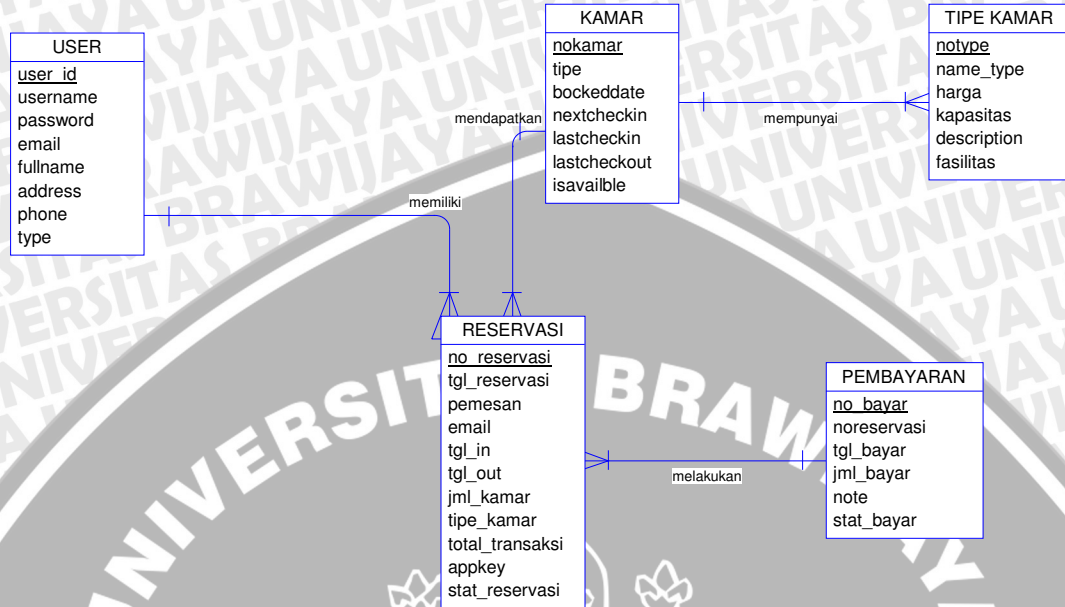
Proses ini mempunyai data keluaran:

- Laporan pembayaran

4.6 ERD

Untuk memodelkan struktur data, hubungan antar data dengan antar penyimpanan dalam DFD maka diperlukan pemodelan data. Setelah digambarkan aliran arah dan tujuan data serta proses yang terjadi, maka dibuat *Entity Relationship Diagram*. Diagram ini disesuaikan untuk menggambarkan hubungan antar entitas yang ada pada sistem yang dibuat.

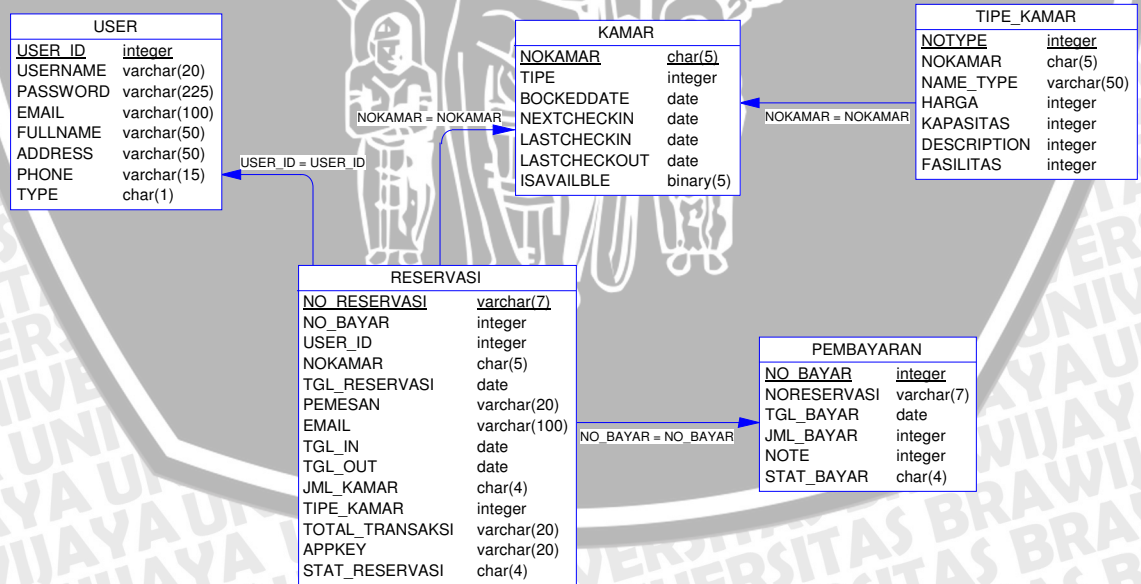
4.6.1 Conceptual Model



Gambar 4.7. Conceptual Model

Sumber : [Perancangan]

4.6.2 Physical Model



Gambar 4.8. Physical Model

Sumber : [Perancangan]

4.6.3 Tabel

Nama Tabel : User

Primary Key : User_Id

Tabel 4.3. Tabel User

No	Nama	Type	Panjang	Keterangan
1.	User_Id	Integer	11	No. Identitas User
2.	Username	Varchar	20	Nama depan User
3.	Password	Varchar	225	Password User
4.	Email	Varchar	50	Email User
5.	Fullname	Varchar	50	Nama Lengkap
6.	address	Varchar	50	Alamat User
7.	Phone	Varchar	50	Telepon User
	Type	Varchar	15	Type User

Sumber : [Perancangan]

Nama Tabel : Kamar

Primary Key : No_Kamar

Tabel 4.4. Tabel Kamar

No	Nama	Type	Panjang	Keterangan
1.	nokamar	Char	5	No. Kamar
2.	tipe	integer	20	Tipe/ Jenis Kamar
3.	bockeddate	date	-	Tanggal Booking Kamar
4.	nextcheckin	date	-	Tanggal Cek In Selanjutnya
5.	lastcheckin	date	-	Tanggal Terakhir Cek In
6.	lastchekout	date	-	Tanggal Terakhir Cek Out

7.	isavailable	benary	5	Status Kamar
----	-------------	--------	---	--------------

Sumber : [Perancangan]

Nama Tabel : Pembayaran

Primary Key : No_Bayar

Foreign Key : No_Resevasi

Tabel 4.5. Tabel Pembayaran

No	Nama	Type	Panjang	Keterngan
1.	No_Bayar	Integer	11	No Bayar
2.	NoReservasi	Varchar	7	No. Reservasi
3.	Tgl_Bayar	Date	-	Tanggal Pembayaran
4.	Jml_Bayar	Integer	-	Jumlah Pembayaran
5.	Note	Integer	-	Catatan Pembayaran
6.	Stat_Bayar	Integer	-	Status Pembayaran

Sumber : [Perancangan]



Nama Tabel : Reservasi
 Primary Key : No_Reservasi
 Foreign Key : CID, No_Kamar, No_Bayar

Tabel 4.6. Tabel Reservasi

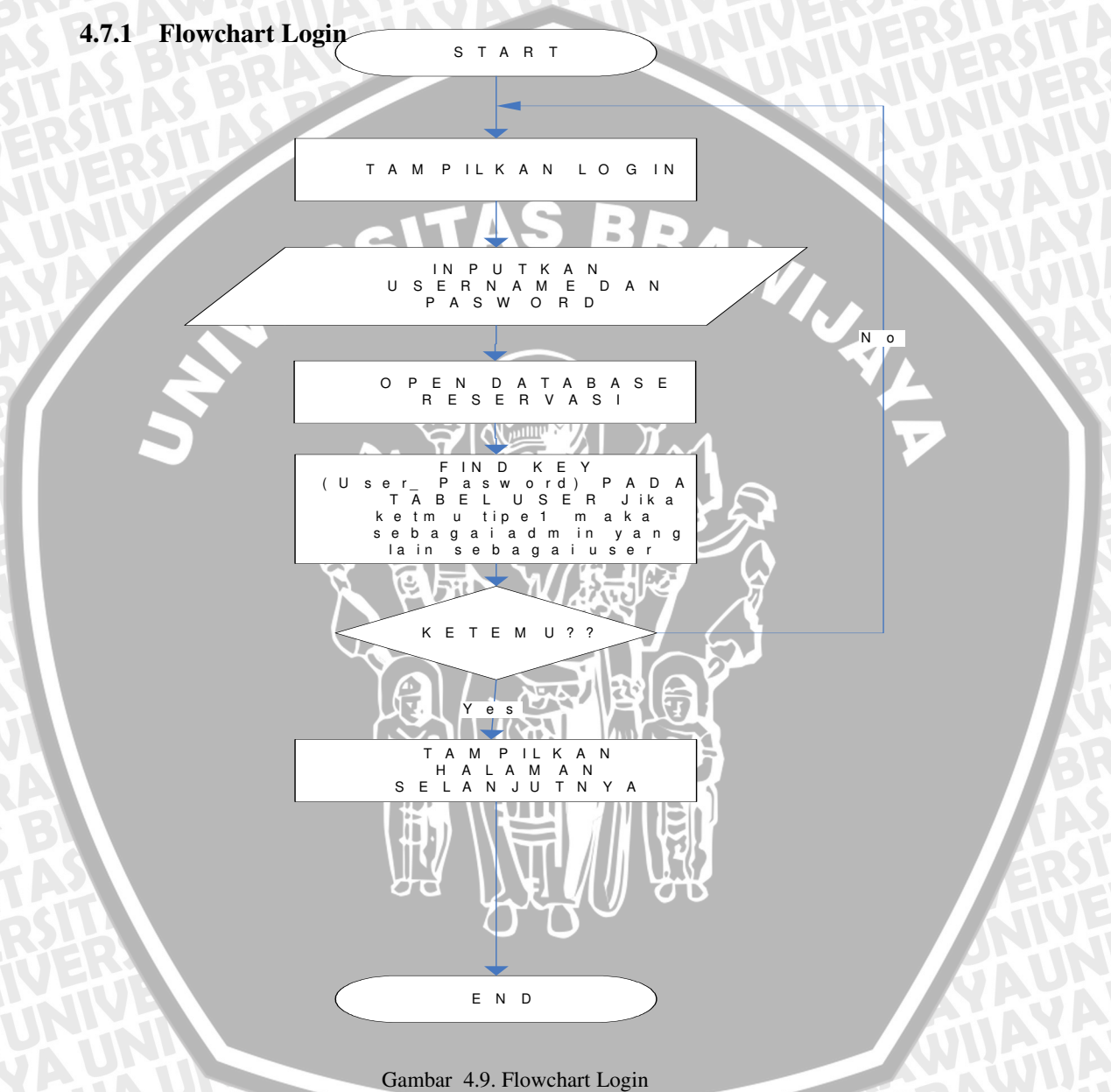
No	Nama	Type	Panjang	Keterangan
1.	No_reservasi	Char	6	No reservasi
2.	Tgl_Reservasi	Date	-	Tanggal Reservasi
3.	Pemesan	Varchar	50	Nama Yang Reservasi
4.	Email	Varchar	100	Email Yang Reservasi
5.	Tgl_In	Date	-	Tgl Cek In
6.	Tgl_Out	Date	-	Tgl Cek Out
7.	Jml_Kamar	Char	4	Jml. Kamar Yg Dipesan
8.	Tipe_Kamar	Integer	11	Type Kamar Yg Dipesan
9.	Total_Transaksi	Integer	11	Total Reservasi
10.	App_Key	Varchar	20	Kunci Reservasi
11.	Stat_Reservasi	Char	4	Status Reservasi

Sumber : [Perancangan]

4.7 Desain Algoritma

Program dapat digambarkan melalui diagram yang disebut *flowchar* Diantara lain dibawah ini:

4.7.1 Flowchart Login

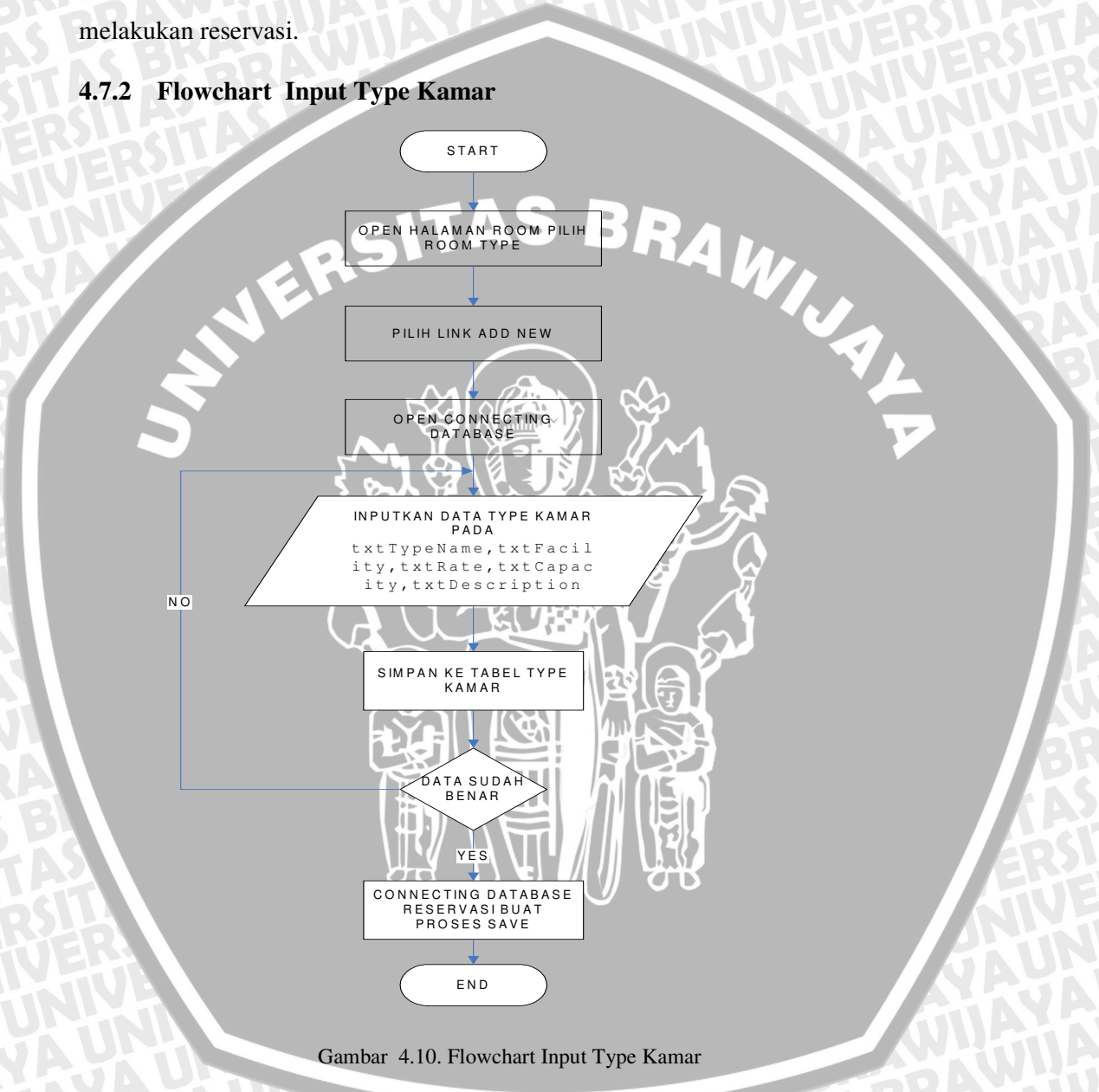


Gambar 4.9. Flowchart Login

Sumber : [Perancangan]

Pada gambar 4.9 merupakan flowchart tampilan login. Sistem ini merancang user akan melakukan reservasi pemesanan hotel. Selanjutnya sistem ini akan meminta user untuk memasukkan input username dan paswod untuk melakukan reservasi.

4.7.2 Flowchart Input Type Kamar

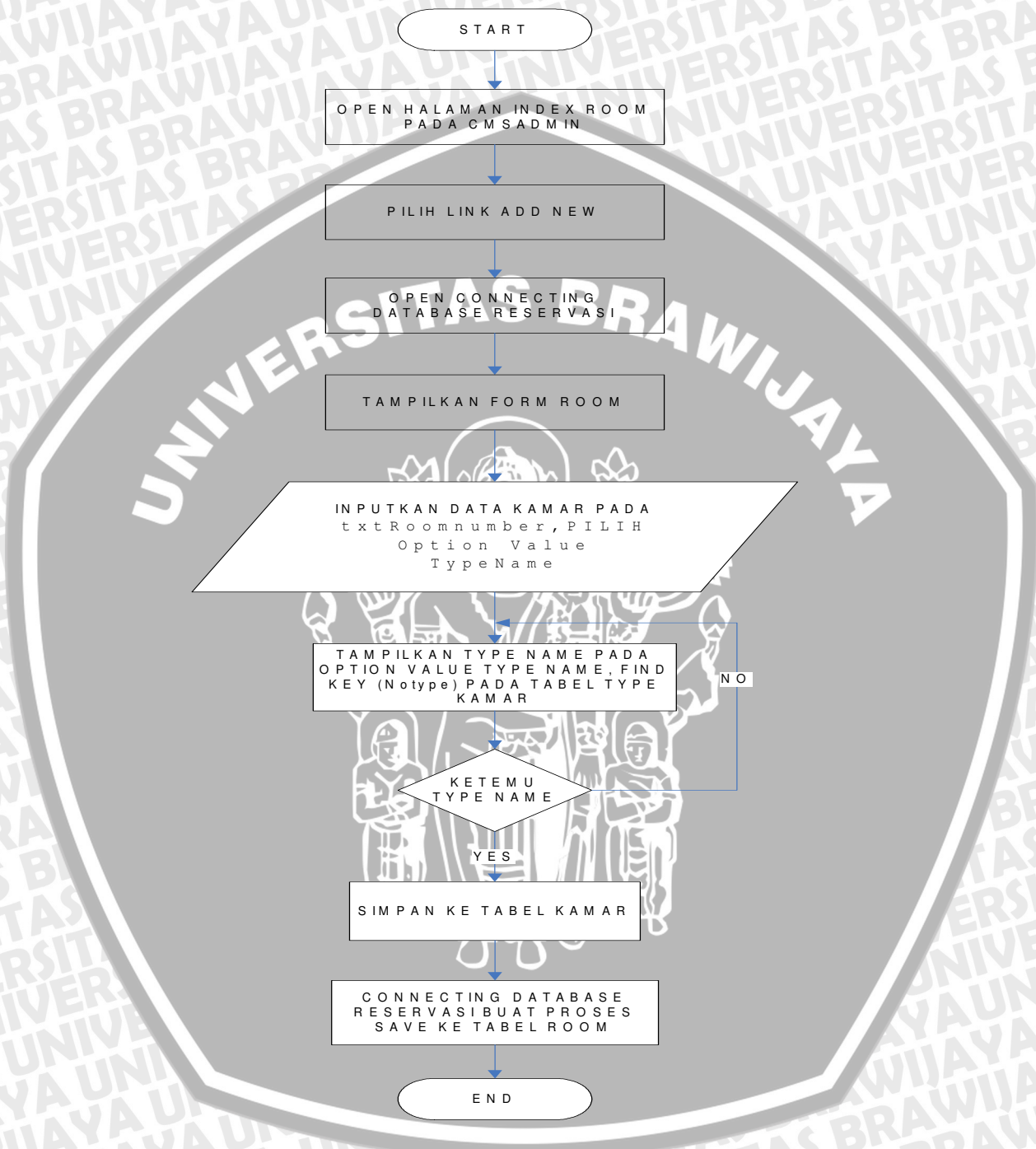


Gambar 4.10. Flowchart Input Type Kamar

Sumber : [Perancangan]

Pada gambar 4.10 merupakan flowchart input type kamar, dimana akan dirancang user akan melakukan pemilihan kamar sesuai yang diinginkannya.

4.7.3 Flowchart Entry Data Kamar

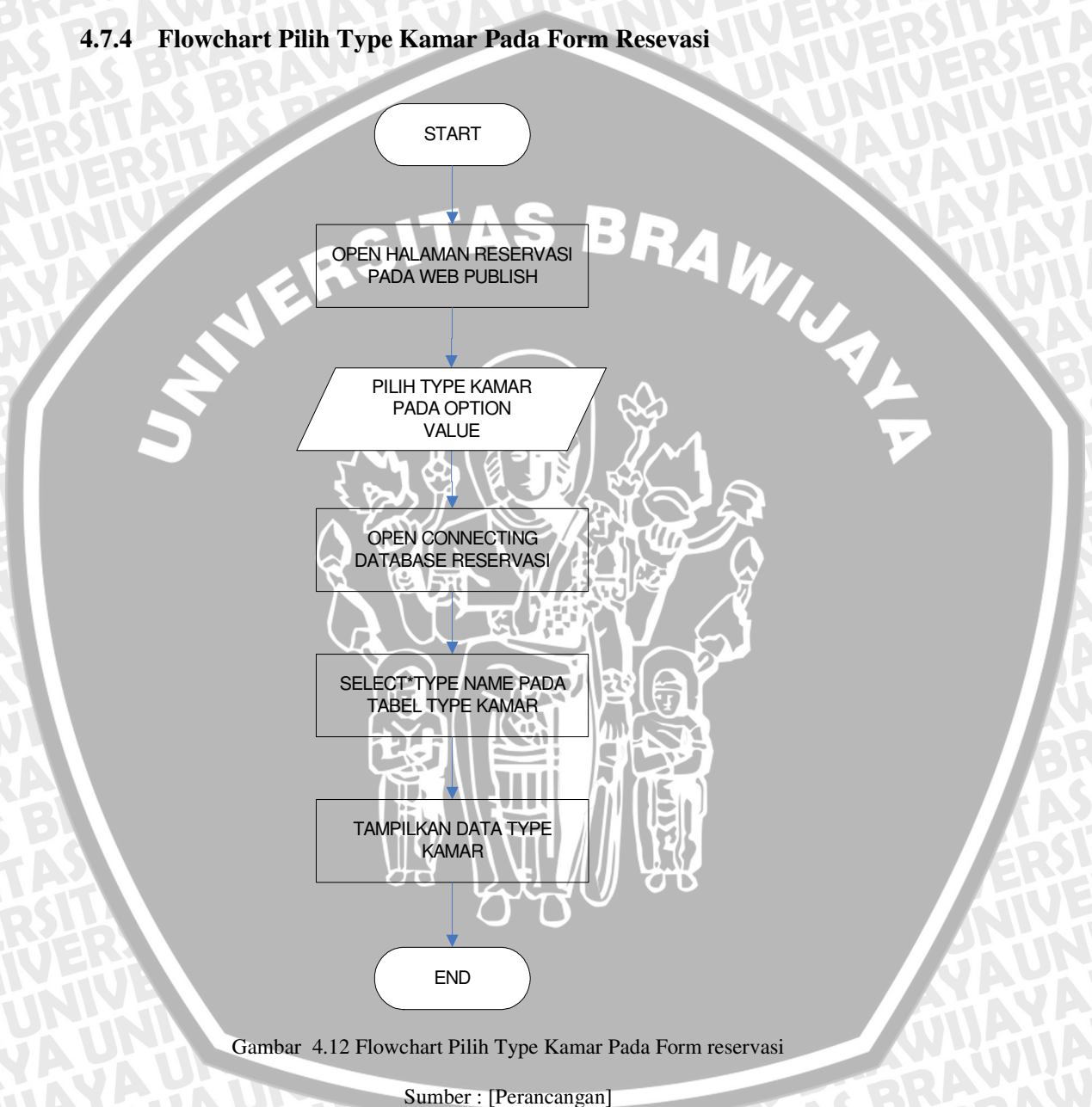


Gambar 4.11. Flowchart Entry Data Kamar

Sumber : [Perancangan]

Gambar 4.11 Flowchart entry data kamar, yang akan merancang sistem dimana user akan memilih memilih tipe kamar yang diinginkannya. Setelah itu user menentukan tipe kamar sesuai yang diinginkannya.

4.7.4 Flowchart Pilih Type Kamar Pada Form Reservasi

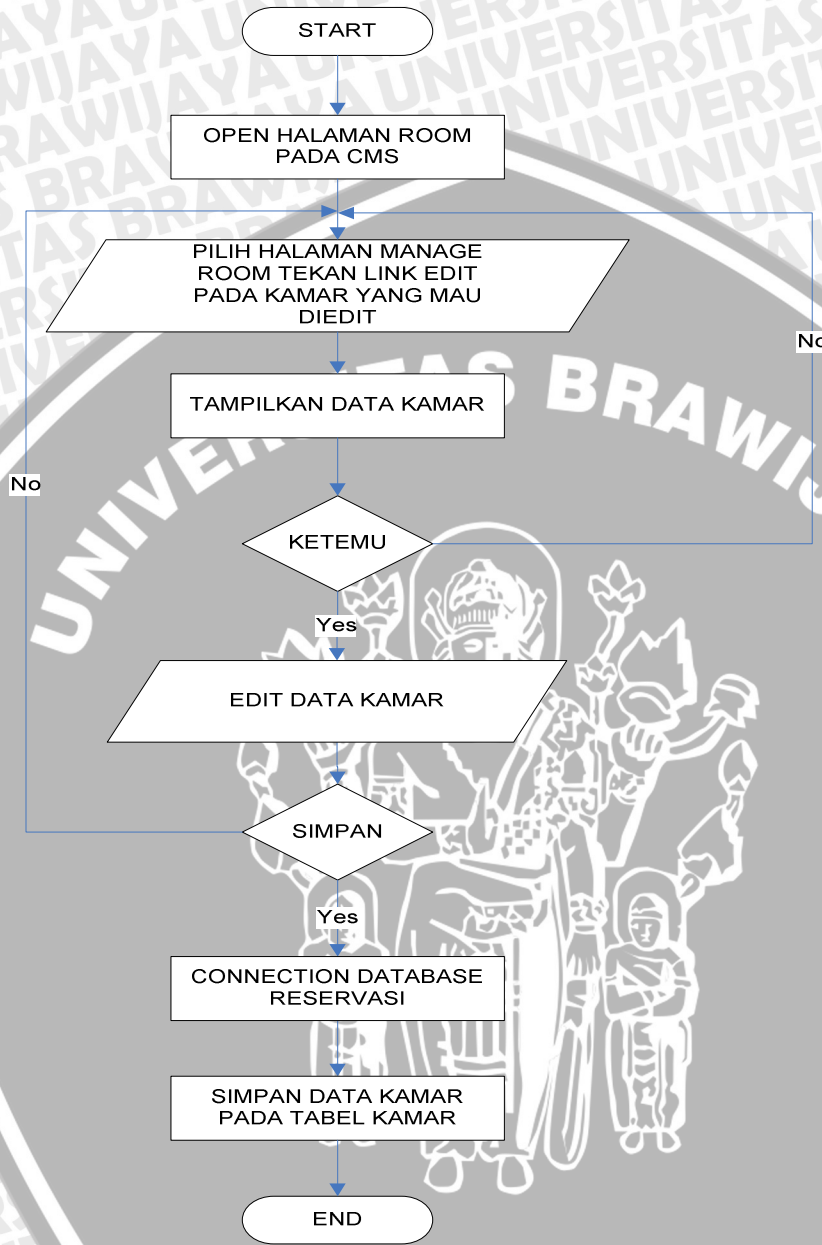


Gambar 4.12 Flowchart Pilih Type Kamar Pada Form reservasi

Sumber : [Perancangan]

Pada gambar 4.12 merupakan perancangan dimana user akan melakukan pengisian form reservasi untuk melakukan reservasi sesuai yang diinginkannya.

4.7.5 Flowchart Edit Data Kamar

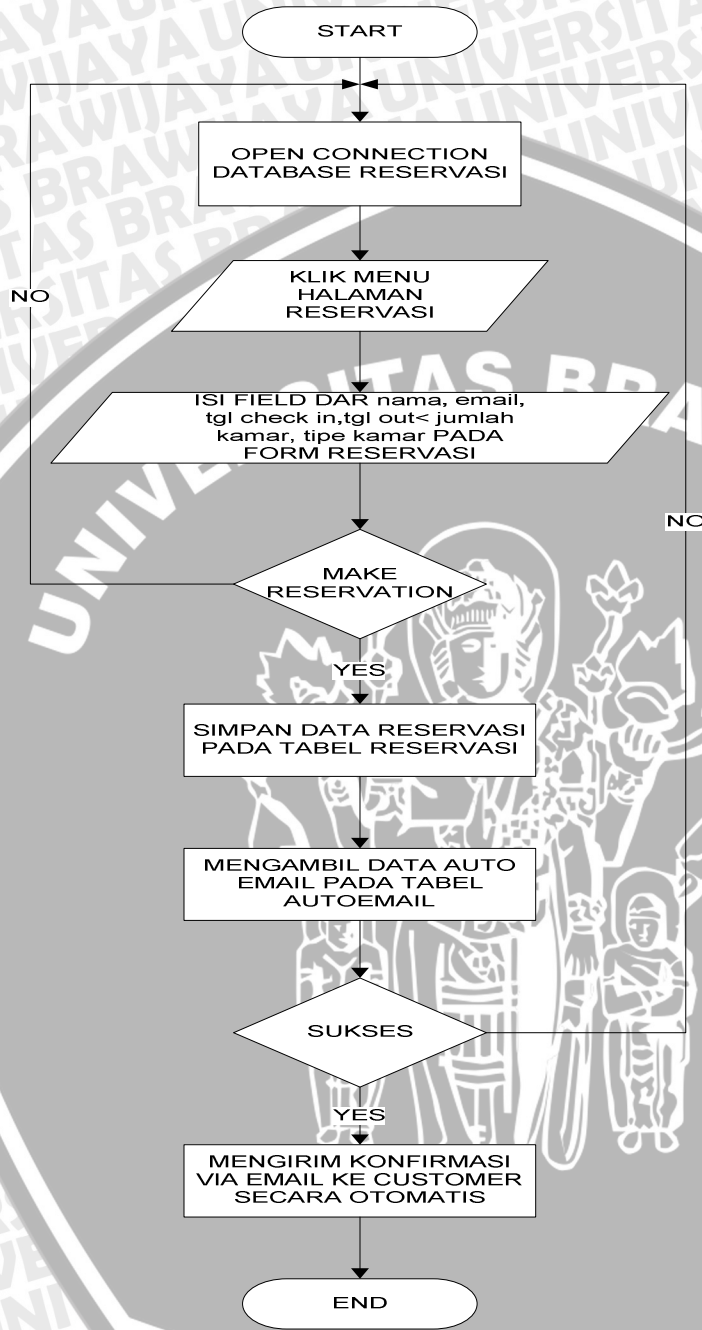


Gambar 4.13 Flowchart Edit Data Kamar

Sumber : [Perancangan]

Gambar 4.13 merupakan rancangan apabila user melakukan perubahan pemesanan kamar.

4.7.7 Flowchart Reservasi

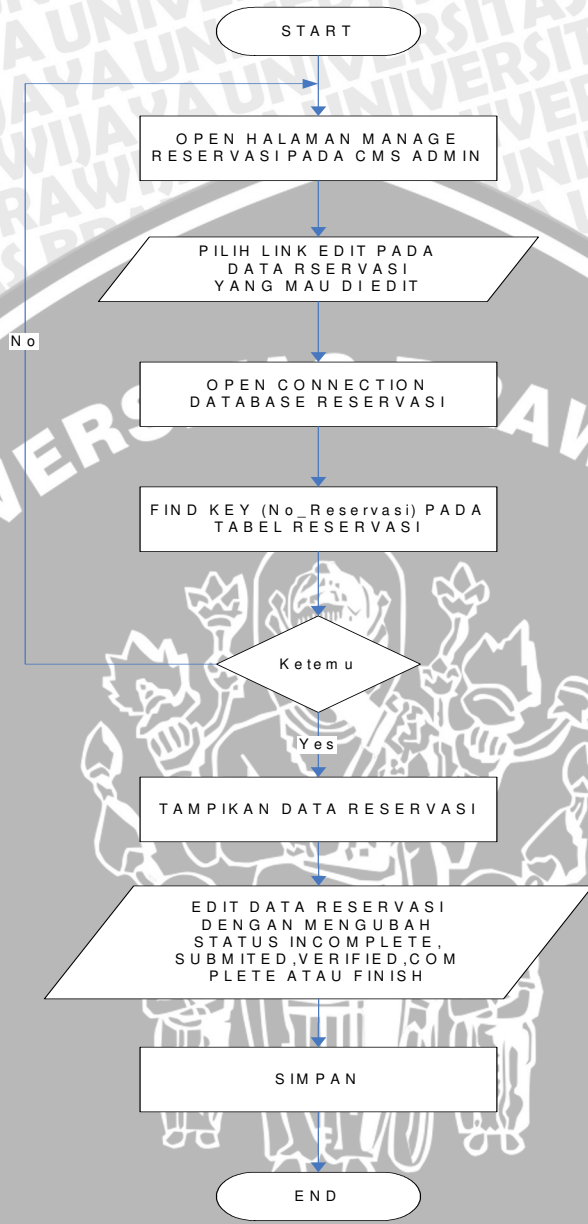


Gambar 4.14 Flowchart Simpan Data Reservasi

Sumber : [Perancangan]

Pada gambar 4.14 merupakan rancangan penyimpanan data sesuai keinginan user pada waktu melakukan reservasi.

4.7.6 Flowchart Edit Data Reservasi



Gambar 4.15. Flowchart Edit Data Reservasi

Sumber : [Perancangan]

Gambar 4.15. Flowchart Edit Data Reservasi merupakan rancangan dimana admin akan melakukan perubahan data pada waktu user melakukan perubahan pemesanan kamar.

4.7.7 Flowchart Inputkan Pembayaran

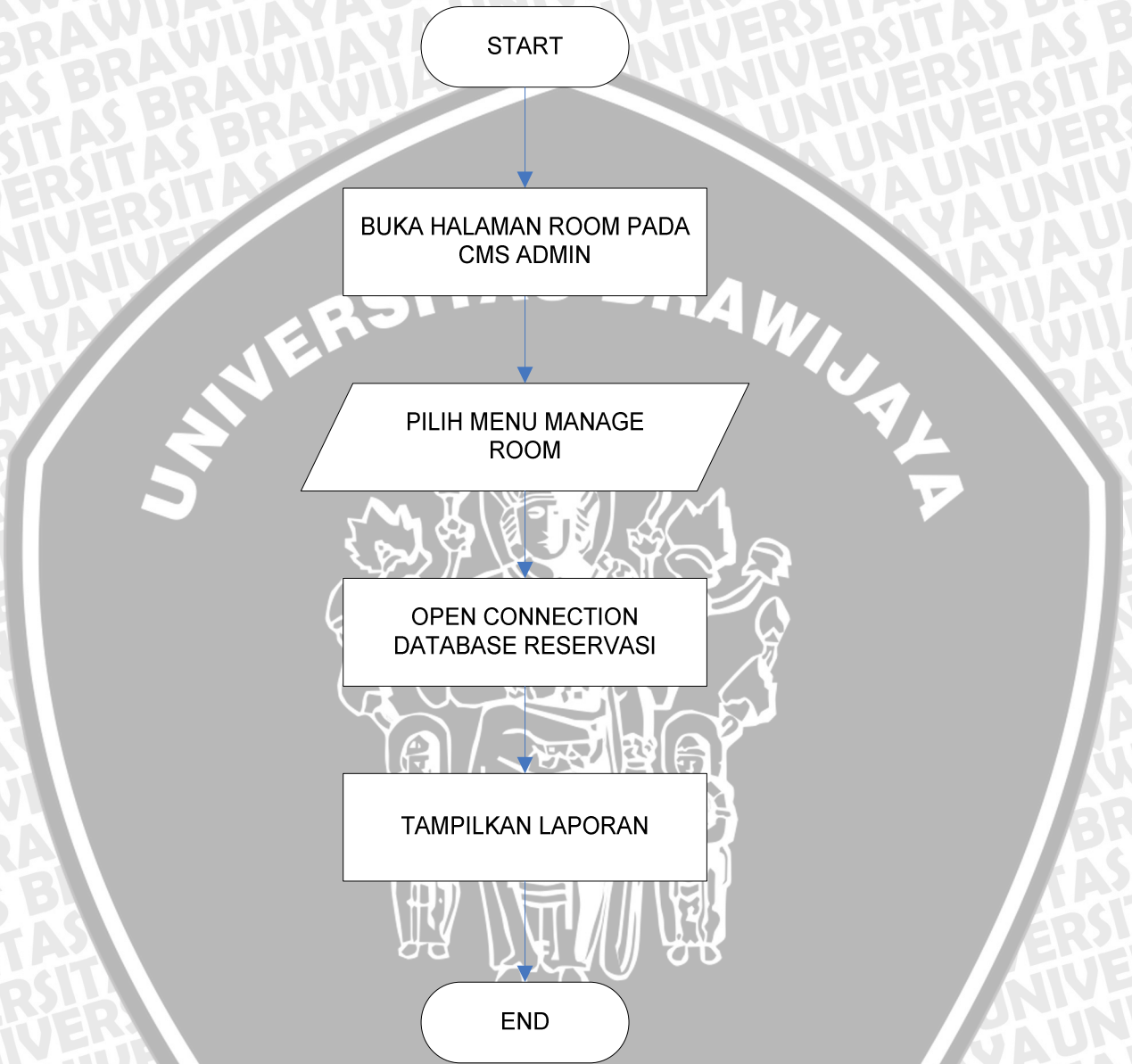


Gambar 4.16. Flowchart Input Pembayaran

Sumber : [Perancangan]

Pada gambar 4.16 Merupakan rancangan proses pemasukan data hasil pembayaran dan reservasi oleh customer.

4.7.8 Flowchart Laporan Kamar



Gambar 4.17 Flowchart Laporan Kamar

Sumber : [Perancangan]

Pada gambar 4.17 merupakan suatu sistem yang dirancang untuk menampilkan report pemesanan kamar ketika melakukan transaksi reservasi.

4.7.9 Flowchart Laporan Pembayaran

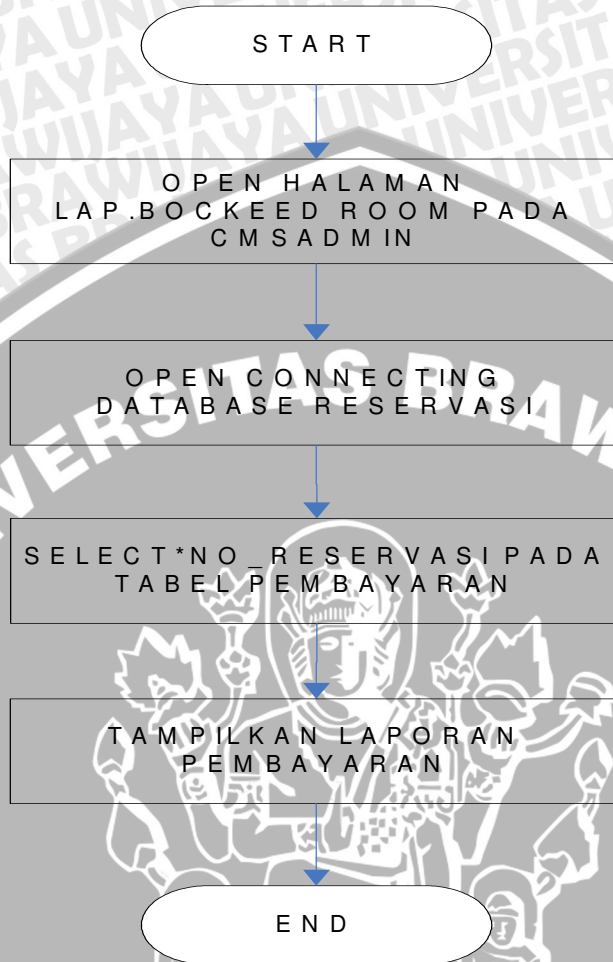


Gambar 4.18 Flowchart Laporan Pembayaran

Sumber : [Perancangan]

Pada gambar 4.18 merupakan suatu sistem yang dirancang untuk menampilkan laporan pembayaran ketika customer melakukan transaksi pembayaran.

4.7.10 Flowchart Laporan Reservasi



Gambar 4.19 Flowchart Laporan Reservasi

Sumber : [Perancangan]

Pada gambar 4.19 flowchart Laporan Reservasi. Sistem ini merancang hasil dari reservasi yang dilakukan oleh user.

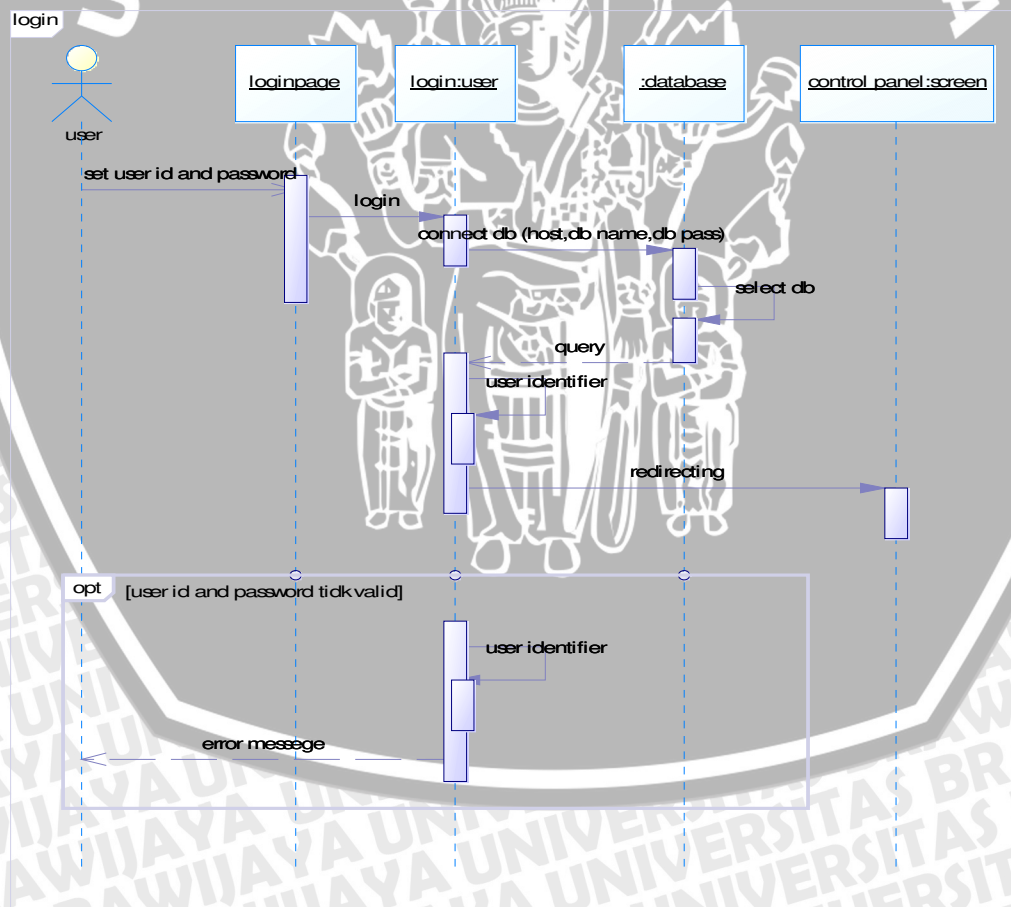
4.7.11 Diagram Sekuensial (*Sequence Diagram*)

Pemodelan aliran jalannya proses interaksi antar objek atau kelas yang disusun berdasarkan urutan waktu ditunjukkan oleh *sequence diagram*. *Sequence diagram* digunakan untuk menggambarkan skenario atau rangkaian langkah-

langkah yang dilakukan sebagai respons dari sebuah *event* untuk menghasilkan *output* tertentu. Diawali dari apa yang men-*trigger* aktivitas tersebut, proses dan perubahan apa saja yang terjadi secara internal dan *output* apa yang dihasilkan. Diagram sekuensial (*sequence diagram*) disusun dengan mengambil acuan pada *use case* dan kelas-kelas yang membentuk fungsionalitas yang digambarkan pada *use case* tersebut.

- **Diagram sekuensial *use case login***

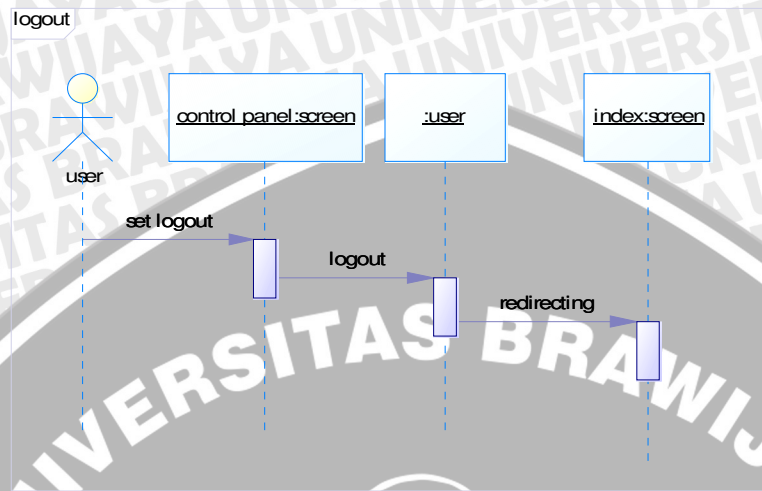
Gambar 4.20 merupakan diagram sekuensial untuk *use case log in*. Diagram ini terdiri dari sebuah aluran utama dan 2 buah aliaran opsional yaitu *invalid input* dan *undevined user*.



Gambar 4.20 *sequence diagram login*
 Sumber:[Perancangan]

- **Diagram sekuensial use case log out**

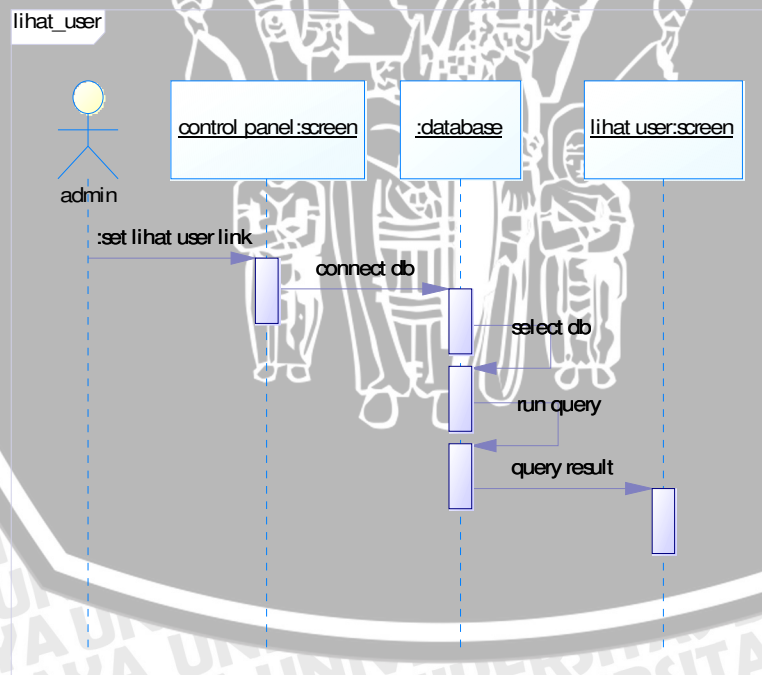
Gambar 4.21 merupakan diagram sekuensial untuk use case log out.



Gambar 4.21 sequence diagram log out
Sumber: [Perancangan]

- **Diagram sekuensial use case lihat user**

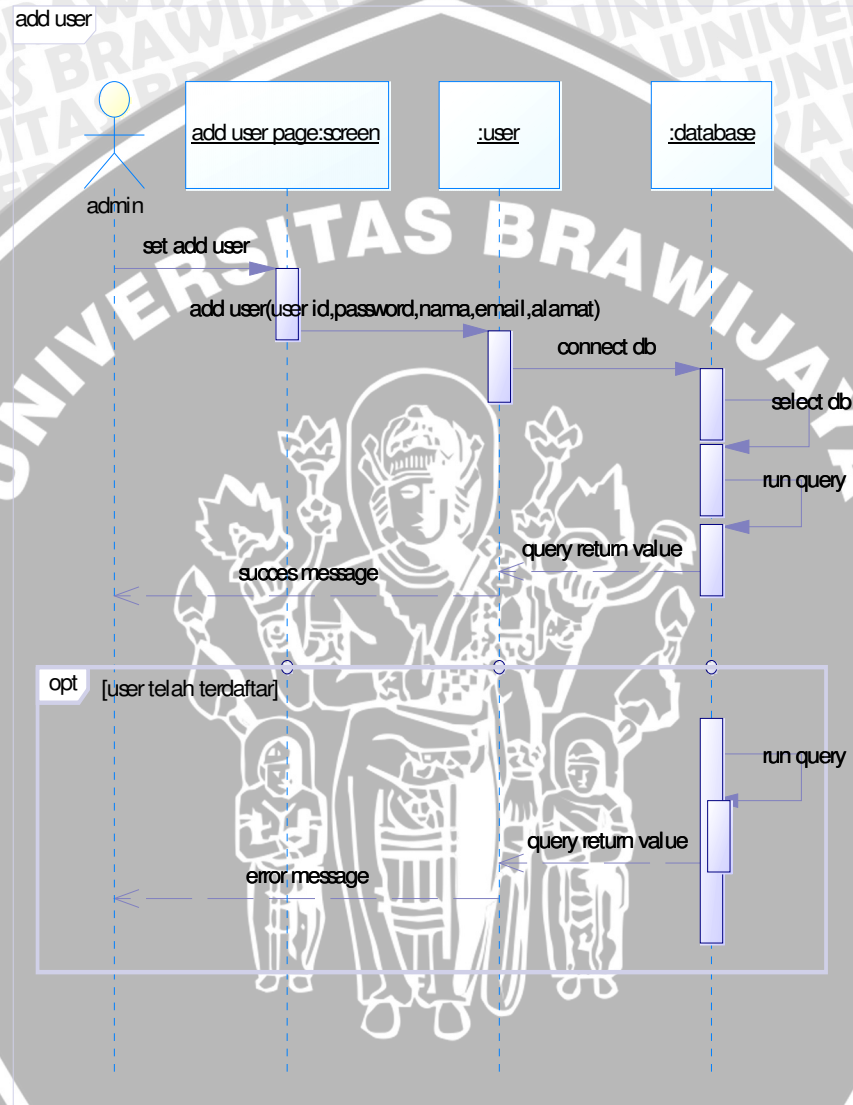
Gambar 4.22 merupakan diagram sekuensial untuk use case lihat user.



Gambar 4.22 sequence diagram lihat user.
Sumber: [Perancangan]

- **Diagram sekuensial *use case add user***

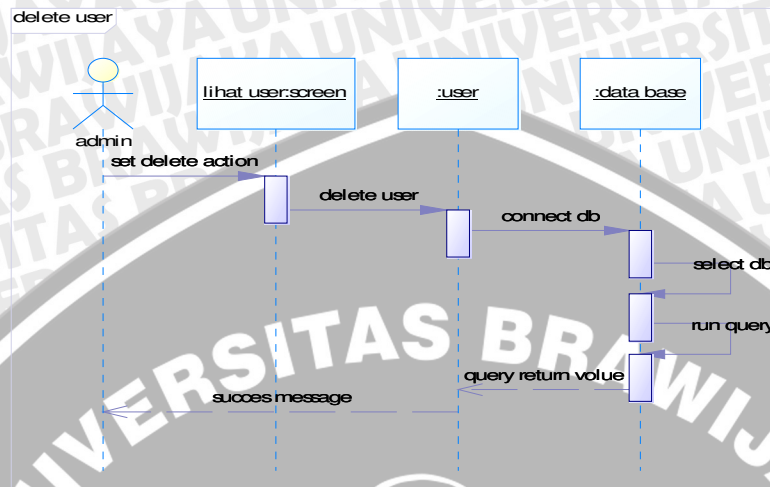
Gambar 4.23 merupakan diagram sekuensial untuk *use case add user*. Diagram ini terdiri dari sebuah aliran utama dan sebuah aliran opsional yaitu (negative) user sudah terdaftar.



Gambar 4.23 *sequence diagram add user*
Sumber: [Perancangan]

- **Diagram sekuensial *use case delete user***

Gambar 4.24 merupakan diagram sekuensial untuk *use case delete user*.

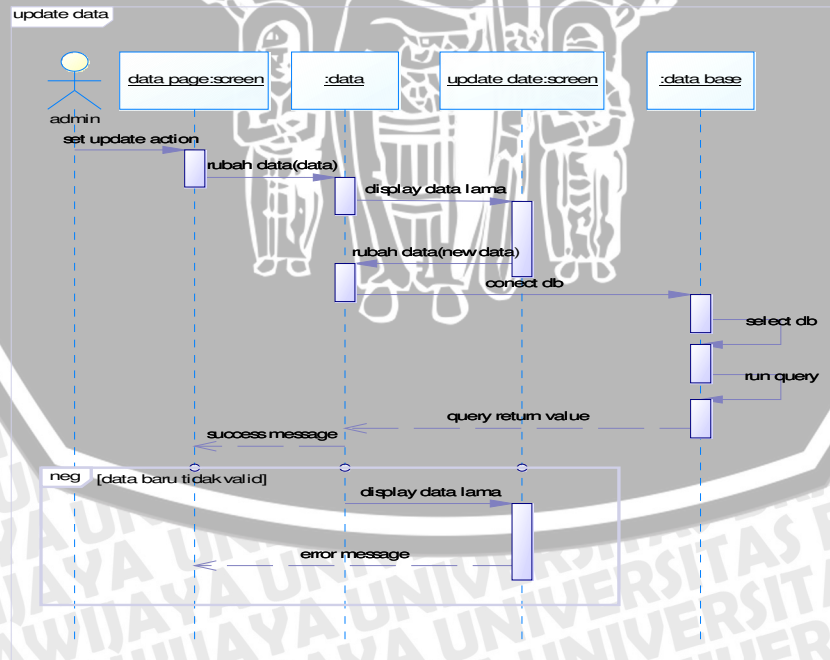


Gambar 4.24 *sequence diagram delete user*
 Sumber: [Perancangan]

- **Diagram sekuensial *use case add user***

Gambar 4.25 merupakan diagram sekuensial untuk *use case update data*.

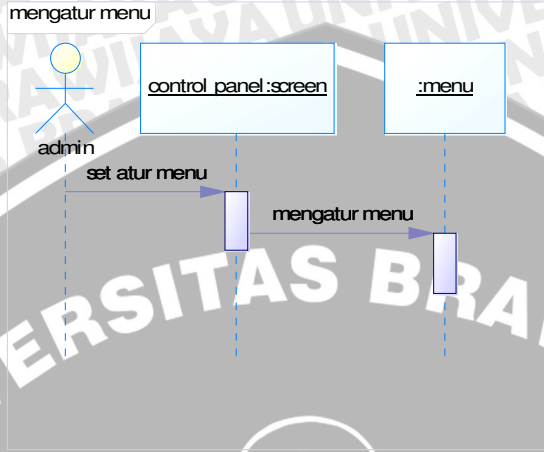
Diagram ini terdiri dari sebuah aliran utama dan sebuah aliran opsional yaitu (negative) data baru tidak valid.



Gambar 4.25 *sequence diagram update data*
 Sumber: [Perancangan]

- **Diagram sekuensial *use case mengatur menu***

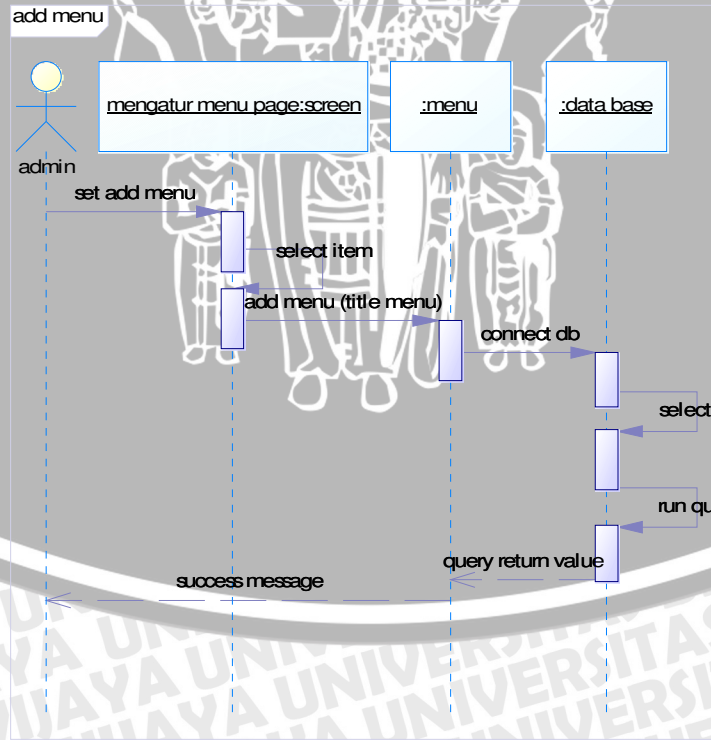
Gambar 4.26 merupakan diagram sekuensial untuk *use case mengatur menu*.



Gambar 4.26 *sequence diagram update data*
 Sumber: [Perancangan]

- **Diagram sekuensial *use case add menu***

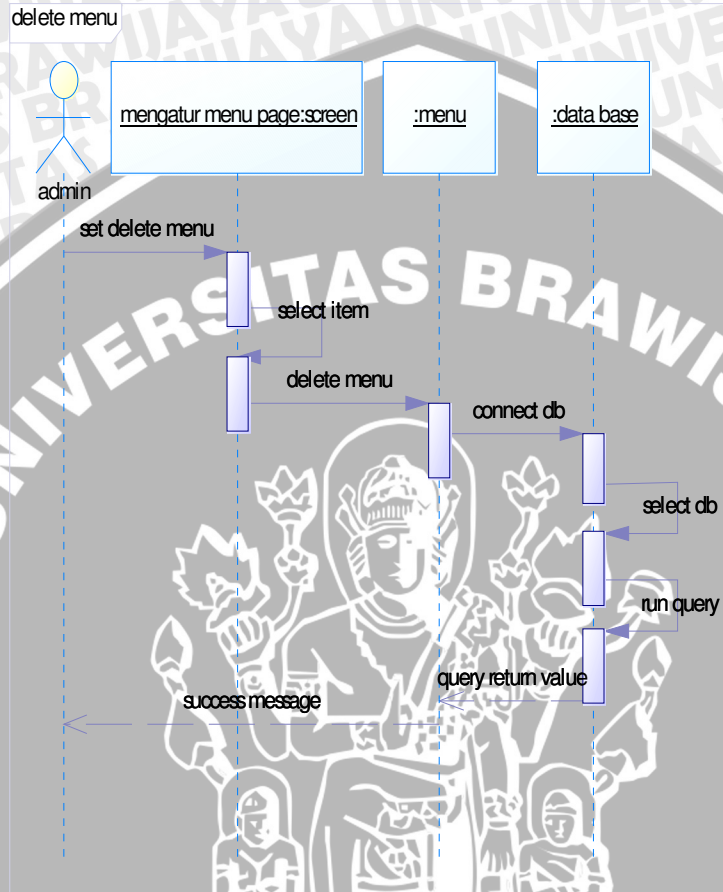
Gambar 4.27 merupakan diagram sekuensial untuk *use case add menu*.



Gambar 4.27 *sequence diagram update data*
 Sumber: [Perancangan]

- Diagram sekuensial *use case delete menu*

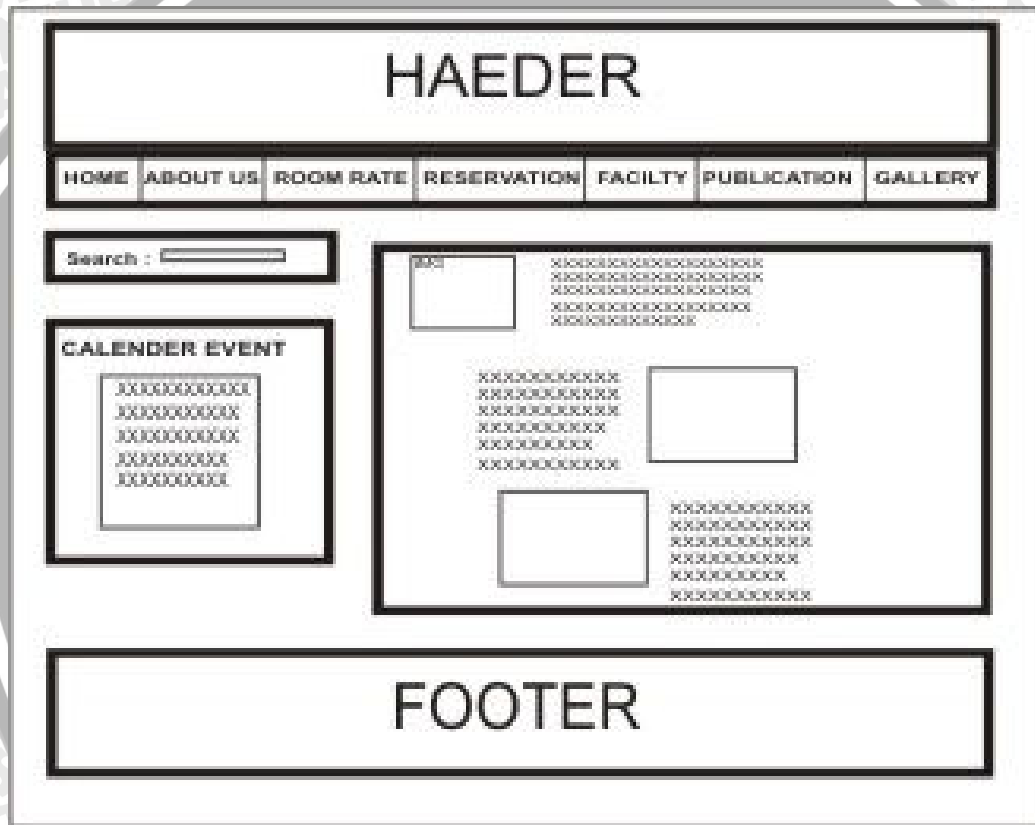
Gambar 4.28 merupakan diagram sekuensial untuk *use case delete menu*.



Gambar 4.28 *sequence diagram update data*
 Sumber: [Perancangan]

4.8.1 Desain utama

Menu ini merupakan tampilan menu utama secara umum. Semua *user* bisa masuk dalam menu ini untuk melihat fasilitas-fasilitas yang ditawarkan dan sekaligus melakukan reservasi. Adapun rancangan dari menu tersebut dapat dilihat dalam Gambar 4.29

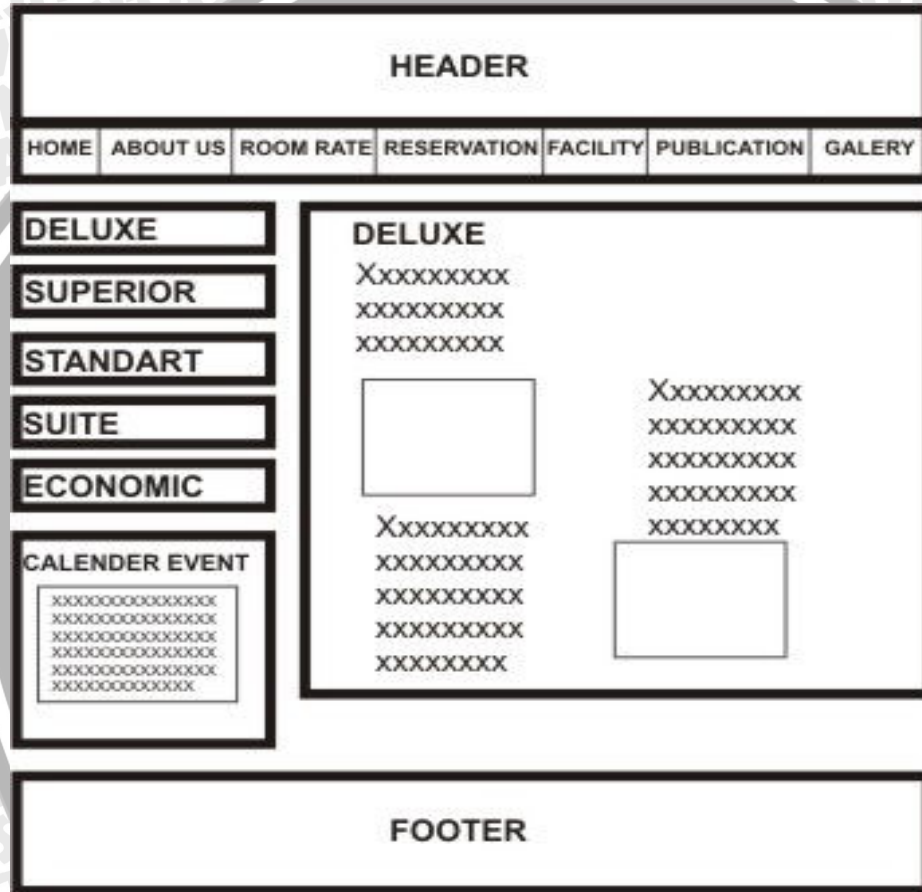


Gambar 4.29 Desain Utama

Sumber: [Perancang]

4.8.2`Desain Room Type

Pada menu ini menampilkan tipe kamar yang di tawarkan oleh management hotel kepada konsummen. Konsumen bisa melihat dan memilih kamar sesuai yang mereka inginkan. Rancangannya dapat di lihat pada gambar 4.30.

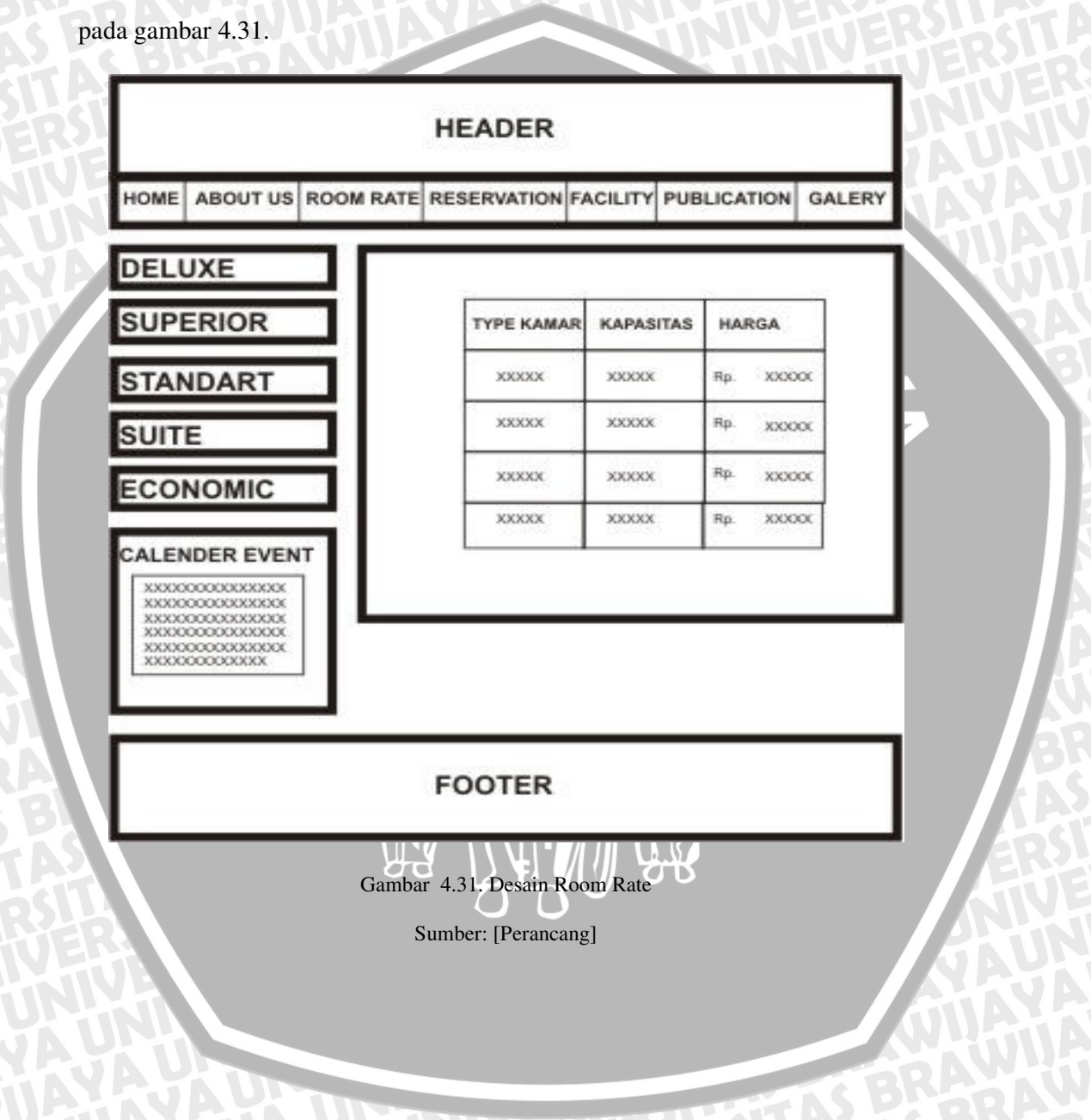


Gambar 4.30. Desain Room Type

Sumber: [Perancang]

4.8.3 Desain Room Rate

Pada menu ini menampilkan harga kamar sesuai tipenya. Konsumen bisa memilih harga kamar sesuai yang mereka inginkan. Rancangannya dapat di lihat pada gambar 4.31.



Gambar 4.31. Desain Room Rate

Sumber: [Perancang]

4.8.4 Desain Reservation

Pada menu ini menampilkan proses pengisian reservasi yang diisi oleh calon pelanggan sesuai tipe kamar yang mereka inginkan dan kapan calon pelanggan melakukan check in dan check out serta berapa jumlah kamar yang mereka pesan. Rancangannya dapat di lihat pada gambar 4.32.

The diagram illustrates the layout of a reservation page. It is structured as follows:

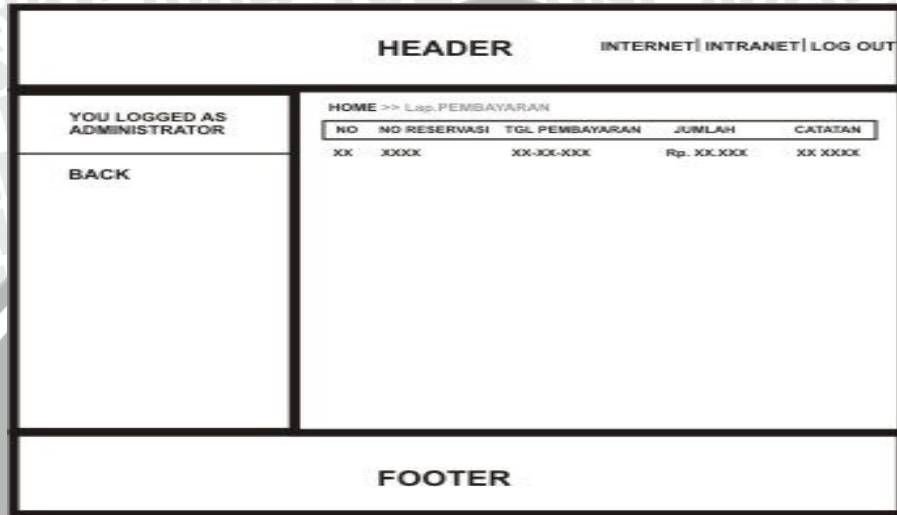
- HEADER:** Contains navigation links: HOME, ABOUT US, ROOM RATE, RESERVATION, FACILITY, PUBLICATION, GALERY.
- Left Sidebar:**
 - Room type selection buttons: DELUXE, SUPERIOR, STANDART, SUITE, ECONOMIC.
 - CALENDER EVENT section with placeholder text (XXXXXXXXXXXXXXXXXX).
- Main Content Area:**
 - Reservasi** section with the instruction "Please make reservation complete".
 - Form fields: Nama (text input), Email (text input), Tanggal check in (calendar icon), Tanggal check iout (calendar icon), Jumlah kamar (text input), Tipe kamar (text input).
 - MAKE RESERVATION** button.
- FOOTER:** A section at the bottom of the page.

Gambar 4.32 Desain Reservasi

Sumber:[Perancang]

4.8.5 Desain Lap.Pembyaran

Pada menu ini menampilkan laporan pembayaran sesudah melakukan reservasi. Rancangannya dapat di lihat pada gambar 4.33

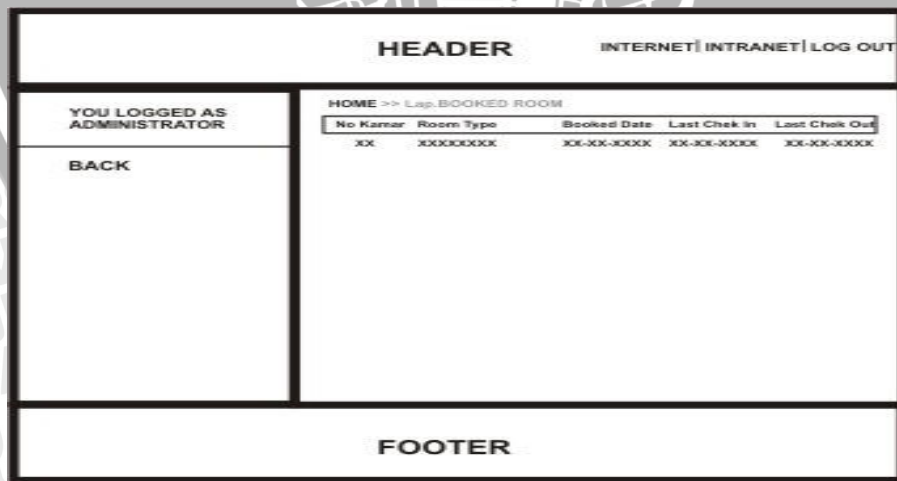


Gambar 4.33. Desain Lap.Pembayaran

Sumber: [Perancang]

4.8.6 Desain Lap.Booked Kamar

Pada menu ini menampilkan suatu kamar sudah di pesan. Rancangannya dapat di lihat pada gambar 4.26



Gambar 4.26. Desain Lap.Booked Kamar

Sumber: [Perancang]

BAB V IMPLEMENTASI

Implementasi merupakan penerapan berupa *coding* hasil dari proses analisis dan perancangan yang dibuat sebelumnya berdasarkan bahasa pemrograman tertentu dalam hal ini adalah PHP dan MySQL. Implementasi diawali dengan perancangan algoritma program dan dilanjutkan dengan penerapan algoritma program pada bahasa pemrograman berupa kode program sesungguhnya.

5.1. Spesifikasi Hardware dan Software

Untuk menjalankan Sistem Sistem Penjualan Online dibutuhkan Software untuk server, yaitu :

- Sistem Operasi Windows XP Service Pack 1
- Xampp 1.4 (Apache, PHP, MySQL)
- Browser (Internet Explorer, Mozilla, Opera)

Untuk pembuatan dan pengoperasian aplikasi Sistem Informasi Penjualan Online pada Tugas Akhir ini digunakan komputer dengan spesifikasi sebagai berikut :

Tabel 5.1 Spesifikasi Hardware

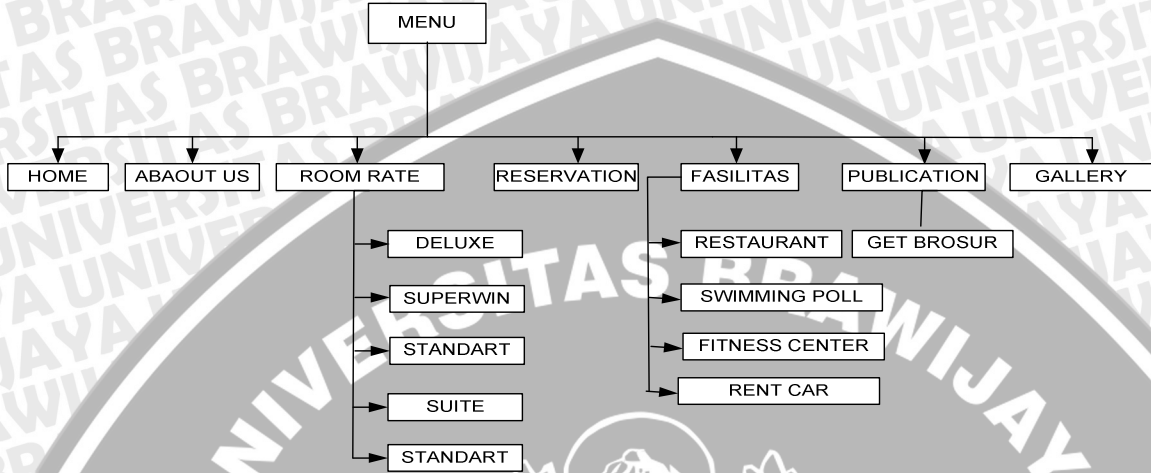
Nama	Tipe
Processor	min Intel Pentium III
Motherboard	compatible dengan processor
Memory	min 256 MB
Harddisk	40GB ATA
VGA	Onboard
LAN	Modem min 64Kbps
CDRW	CDRW 52x32x52
Monitor	CRT 14"
Keyboard	Standard Keyboard
Mouse	Optical
UPS	UPS 600VA
Casing	Casing ATX min 300watt

Sumber : Implementasi

5.2. Tampilan Program

5.2.1. Menu

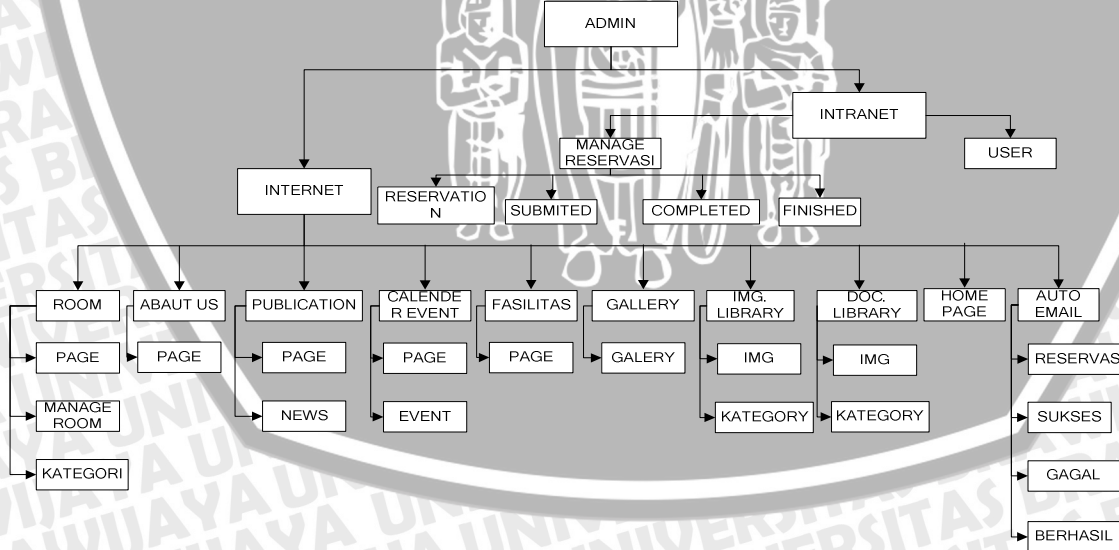
5.2.1.1. Menu Pelanggan



Gambar5.1. Gambar Menu Pelanggan

Sumber : Implementasi

5.2.1.2. Content Management System (CMS)

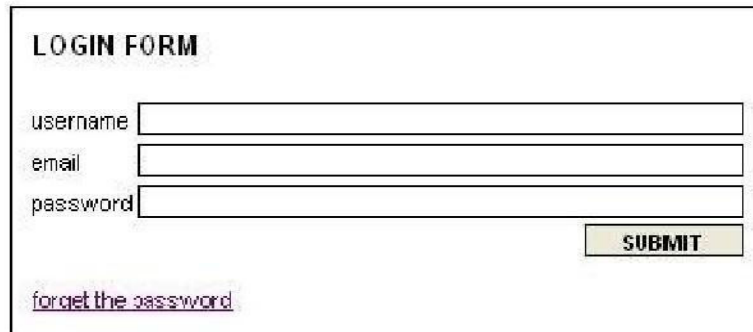


Gambar 5.2. Gambar Content Management System

Sumber : Implementasi

5.2.3. Tampilan dan Logika Program

5.2.3.1. Program Admin



Gambar 5.3. Tampilan Administrator

Sumber : Implementasi

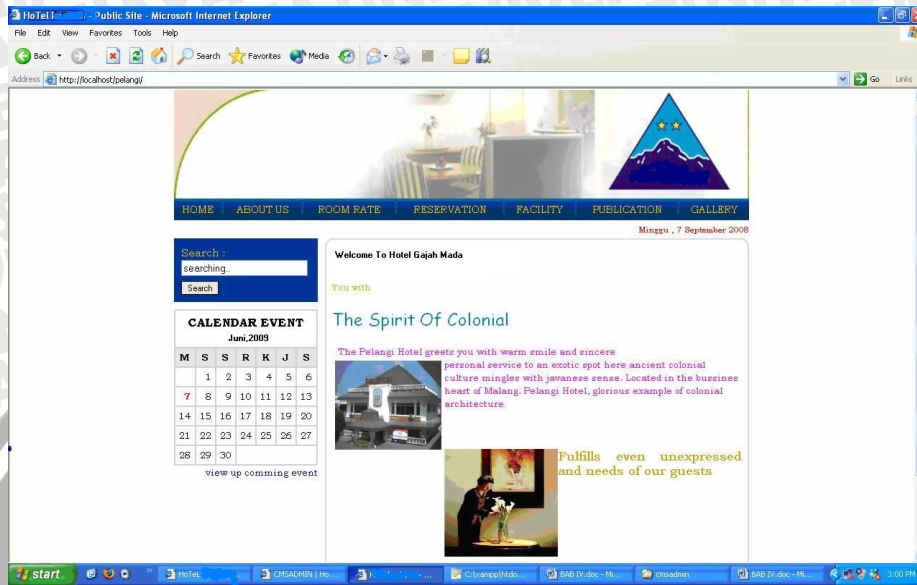
Semua user dan admin yang akan menggunakan sistem CMS ini harus melakukan login terlebih dahulu pada alamat <http://localhost/gajahmada/cmsadmin/login.php>. User harus memasukkan username dan password. Data username dan password akan diperiksa di table tbluser. Apabila data ada maka username akan dicatat pada Session.

Script login dimana mengambil field userID,username,type dari tabel user.

```
$sSQL = "SELECT userID, username, type FROM tbluser WHERE
username='".$$_in_un."' AND email='".$$_in_em."' AND
password='".$$_encrypt($_in_ps)."'";
$exeSQL = mysql_query($sSQL);
if($hsl=mysql_fetch_array($exeSQL)) {
    $_SESSION["userID"] = $hsl["userID"];
    $_SESSION["username"] = $hsl["username"];
}
```

Data yang diinputkan juga akan diperiksa tipe usernya, jika tipenya 3 maka dianggap sebagai *User* akan masuk pada halaman CMS Admin tetapi hanya dapat memenage data, dan jika tipenya 1 maka dianggap sebagai *Admin* akan masuk pada bagian Admin. Maksud User disini adalah pengguna CMS dari karyawan Hotel selain Admin.

5.2.3.2. Program Utama



Gambar 5.4. Tampilan Utama

Sumber : Implementasi

Customer bisa mengakses halaman homepage dengan mengetikkan URL <http://localhost/gajahmada/> pada address bar. Tampilan homepage bisa dimanage dari CMS. Content dari halaman homepage diambil dari tabel tblHomepage pada field hometitle dan homecontent. Script untuk menampilkan halaman homepage.

```

$ssql = " SELECT hometitle,homecontent FROM tblHomepage" ;
$exeSQL = mysql_query($ssql) ;
$hsl = mysql_fetch_array($exeSQL) ;
$title = $hsl[0];
$content = str_replace('../..', $sitePath , $hsl[1]);
    
```

5.2.3.3. Program Input Type Kamar Pada CMS Admin


Add Category


General Properties

Type Name * :

Rates * : per Nite

Capacity * : Org

Description * : 



Merupakan Kamar super mewah yang memiliki fasilitas yang lebih dari kamar yang ada di Hotel Pelangi, dengan harga yang terjangkau dan memiliki kapasitas 4 orang dalam satu kamarnya. Ini merupakan kamar yang diunggulkan di Hotel Pelangi serta dapat dijadikan pasngan yang lagi bulan madu.

Path:

Facility

Facility :

* require

Gambar 5.5. Program Input Kategori kamar

Sumber : Implementasi

Untuk menampilkan data Type Kamar pada public, Admin terlebih dahulu memasukkan data type kamar pada CMS. Dari halaman ini Admin memasukan nama type kamar, harga sewa, kapasitas, fasilitas dan deskripsi dari kamar tersebut. Selanjutnya data type kamar tersebut akan disimpan ke dalam tabel tblTypeKamar. Script untuk memasukan data kamar ke dalam field pada tabel type kamar adalah sebagai berikut:

```

$sTypeName = $_POST["txtTypeName"] ;
$sFacility = $_POST["txtFacility"] ;
$iRate = $_POST["txtRate"] ;
$iCapacity = $_POST["txtCapacity"] ;
$sDescription = $_POST["txtDescription"] ;
$SQL = " INSERT INTO tblTypeKamar VALUES ('', ''
.$sTypeName. ', ' . $iRate. ', ' . $iCapacity. ', ' '
.$sDescription. ', ' . $sFacility. ' )";

```


5.2.3.4. Program Entry Kamar

Gambar 5.6. Program Input Kamar

Sumber : Implementasi

Halaman entry kamar hanya bisa diakses oleh Admin. Halaman ini ditujukan untuk menginputkan data kamar yang ada. Dari data yang didapatkan dari Hotel Gajah Mada bahwa satu type kamar tidak hanya memiliki satu kamar saja. Pada halaman ini Admin hanya memasukan nomer kamar ke dalam textbox dan memilih type kamar dalam bentuk dropdown yang sudah di inputkan dari halaman entry type kamar. Script untuk menampilkan daftar type kamar ke dalam dropdown yang mengambil field noType dan Nama pada tabel type kamar adalah sebagai berikut :

```

$$SQL = "SELECT noType, Nama FROM tblTypeKamar";
$exeSQL = mysql_query($$SQL);
while($hsl=mysql_fetch_array($exeSQL)) { ?>
<option value="<?=$hsl["noType"]?>" <?if ($iSelTipe ==
$hsl["noType"]) echo "selected"?> ><?=$hsl["Nama"]?>
</option><?}?>

```

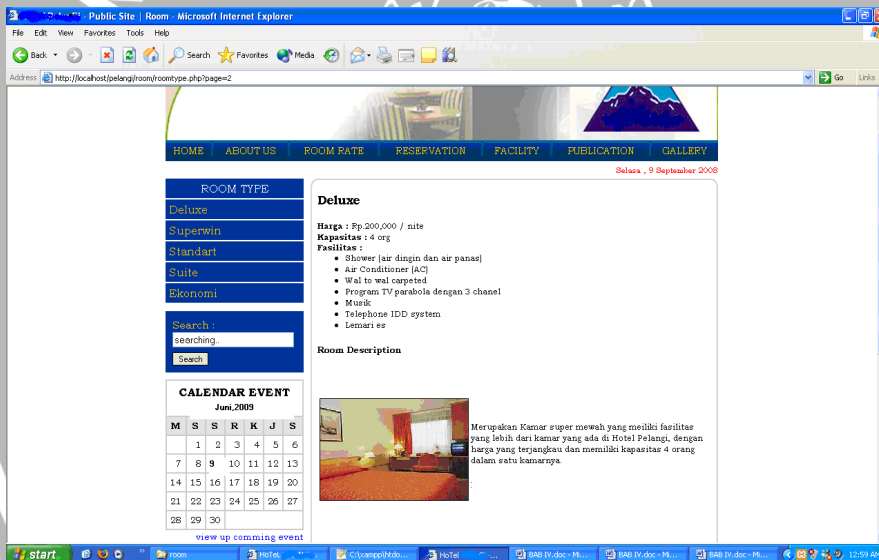
Saat melakukan proses penyimpanan data, sistem akan melakukan pengecekan apakah nomer kamar sudah ada di tabel tbl Room. Jika nomer kamar sudah ada, maka akan keluar keterangan bahwa nomer kamar sudah ada dan tidak akan dilakukan penyimpanan data. Script untuk menyimpan data kamar adalah sebagai berikut :

```

if($action == "save") {
    $sRoomNumber = $_POST["txtRoomNumber"];
    $iSelTipe = $_POST["sel_Tipe"];
    $bFounded = false;
    $_sSQL = "SELECT noKamar FROM tblRoom WHERE
    noKamar='". $sRoomNumber. "' ";
    $_ifFound = mysql_num_rows(mysql_query($_sSQL));
    if($_ifFound == 0) {
        $SQL = "INSERT INTO tblRoom (noKamar, Tipe, isAvailable) VALUES ('"
        . $sRoomNumber . "','" . $iSelTipe . "','1')";
        if(mysql_query($SQL)) {
            include($path . "/include/dbDisconnect.inc");
            header("Location: room_list.php");}
        } else {$bFounded = true;}
        } else {$iSelTipe = 0;}
    }

```

5.2.3.5. Menampilkan Room Rate Pada Public



Gambar 5.7. Program Menampilkan Kategori kamar

Sumber : Implementasi

Pada Tampilan menu Room Rate pada web utama jika field section sama dengan 3 pada tabel section maka menampilkan data kamar beserta type kamar yang oleh sistem secara otomatis menampilkan data yang sudah diinputkan melalui halaman CMS Admin dimana sistem mengambil data pada tabel type kamar (nama, harga, kapasitas,

description, dan fasilitas) sesuai dengan no type yang diminta atau direquest. Script dibawah ini untuk menampilkan menu pada halaman Room Rate.

```

if($sectionID == 3) { //display ROOM
$SQL = "SELECT noType,Nama FROM tblTypeKamar";
$exeSQL = mysql_query($SQL);
$rowSQL=mysql_num_rows($exeSQL);
if ($rowSQL > 0) {
?>
<div id="left-navigation">
<ul>
<li style="text-align:center;color: white">ROOM TYPE</li>
<?
while($hsl=mysql_fetch_array($exeSQL)) {
?>
<li><a href="roomtype.php?page=<?=$hsl["noType"]?>"
><?=$hsl["Nama"]?></a></li>
<?
}
?>
</ul>
</div>
<?
} }
?>

```

Pada halaman Room Rate ditampilkan juga Sub Menu tipe kamar sesuai dengan data yang sudah diinputkan dari CMS. Tidak semua halaman menu menampilkan Sub Menu tipe kamar, karena Sub Menu tipe kamar hanya ditampilkan pada halaman Room Rate. Maka untuk menampilkan Sub Menu tipe kamar dilakukan filter pada sectionID. Script untuk menampilkan Sub Menu pada Menu Room Rate dimana no type pada tabel type kamar sama dengan variabel yang dipasing (, \$pageID):

```

$SQL = "SELECT nama,harga,kapasitas,description,fasilitas FROM
tblTypeKamar WHERE noType = " . $pageID;
$exeSQL = mysql_query($SQL);
while($hsl=mysql_fetch_array($exeSQL)) {
$iNoID = $hsl["noType"];
$sTitle = $hsl["nama"];
$iHarga = $hsl["harga"];
$iKapasitas = $hsl["kapasitas"];
$sContent = $hsl["description"];
$sContent = str_replace ("../../././imgLib/", $sitePath .
"/imgLib/", $sContent);
$sFacility = $hsl["fasilitas"];

```

5.2.3.6. Program Reservasi

Reservasi

Please make a reservation, completed

Nama :

Email :

Tanggal CheckIn : ... x

Tanggal CheckOut : ... x

Jumlah Kamar :

Tipe Kamar :

Gambar 5.8. Program Form reservasi

Sumber : Implementasi

Tampilan halaman reservasi ini menampilkan formulir pemesanan kamar oleh customer. Pada proses reservasi ini pelanggan harus mengisi nama, email pada textbox yang disediakan, tanggal check in, check out dengan menekan tombol browse (...) yang disediakan serta memilih tipe kamar pada dropdown yang disediakan. Value dari dropdown tipe kamar didapatkan dari tabel `tblTypeKamar`.

Script untuk menampilkan Tipe kamar pada dropdown dengan mengambil field `noType` dan nama pada table `TypeKamar`:

```
<td>Tipe Kamar</td>
<td align="center">:</td>
<td>
<select class="textbox" name="selType" style="width:
140px;"/><option value="0">[pilih tipe]</option>
<?
$SQL = "SELECT noType,nama FROM tblTypeKamar";
$exeSQL = mysql_query($SQL);
while($hsl=mysql_fetch_array($exeSQL)) { ?>
    <option value="<?=$hsl["noType"]?>"><?=$hsl["nama"]?></option>
<? } ?>
```

Setelah semua form terisi maka Customer menekan tombol Make Reservation untuk memulai proses penyimpanan data ke dalam tabel `tblReservasi`. Saat proses penyimpanan sistem melakukan pengecekan pada `tblReservasi` apakah sudah ada data

reservasi. Jika tidak ada maka no reservasi akan bernilai "HP00001" . Jika sudah ada sistem mengambil no reservasi terakhir. Lalu sistem akan mengambil nilai integer dari no reservasi dan ditambahkan dengan numeric 1 sehingga menghasilkan no reservasi yang baru.

```

$$SQL = "SELECT no_reservasi FROM tblReservasi ORDER BY
no_reservasi DESC" ;
$exeSQL = mysql_query($$SQL) ;
$numRow = mysql_num_rows($exeSQL) ;
if ($numRow > 0 ) {$hsl = mysql_fetch_array($exeSQL) ;
    $noTrans = $hsl["no_reservasi"] ;
    $Temp = substr ($noTrans, 2, 5) ;
    $Temp = $Temp + 1 ;
    switch (strlen($Temp)) {
        case 1 : $noTrans = "HP0000" . $Temp; break;
        case 2 : $noTrans = "HP000" . $Temp; break;
        case 3 : $noTrans = "HP00" . $Temp; break;
        case 4 : $noTrans = "HP0" . $Temp; break;
        case 5 : $noTrans = "HP" . $Temp; break;
    }
} else {
    $noTrans = "HP00001" ; }

```

Selanjutnya inputan untuk disimpan ke tblReservasi diambil dari form dan khusus untuk jumlah transaksinya didapat dari hasil jumlah kamar yang diinputkan dikalikan dengan harga kamar. Untuk mendapatkan harga kamar dilakukan pengambilan field harga dari tblTypeKamar sesuai dengan type yang sudah dipilih dari dropdown form reservasi.

```

$$Nama = $_POST["txtNama"] ;
$$Email = $_POST["txtEmail"] ;
$dCheckIn = $_POST["txtTanggalCIn"] ;
$dCheckOut = $_POST["txtTanggalCOut"] ;
$iJmlh = $_POST["txtJumlah"] ;
$iTypeKmr = $_POST["selType"] ;
$$SQLKamar = "SELECT harga, Nama FROM tblTypeKamar WHERE noType
= " . $iTypeKmr ;
$exeSQLKamar = mysql_query($$SQLKamar) ;
$hsl = mysql_fetch_array($exeSQLKamar) ;
$harga = $hsl["harga"] ;
$$TipeKamar = $hsl["Nama"] ;
$jumlahTransaksi = $iJmlh * $harga ;
$$KeyApp = date("dmYHis") ;

```

Setelah data sudah terlengkapi maka akan dilakukan proses pengisian atau penyimpanan data ke table reservasi

```
$sSQL = " INSERT INTO tblReservasi VALUES (' . $noTrans. ', ' '
.$currentDate. ', ' ' . $sNama. ', ' ' . $sEmail. ', ' ' . $dCheckIn. ', '
' . $dCheckOut. ', ' ' . $iJmlh. ', ' ' . $iTypeKmr. ', ' ' . $jumlahTransaksi .
', ' ' . $sKeyApp. ', ' , 0) " ;
```

Setelah proses input data selesai maka sistem akan secara otomatis mengirimkan email konfirmasi reservasi. Isi dari email konfirmasi diambil dari tabel tblAutoEmail.

Yang mana script untuk melakukan pengiriman email adalah sebagai berikut :

```
$to = $sEmail ;
$header = "From: Hotel
GajahMada<reservasi@hotelgajahmada.com>\n";
$header .= "MIME-Version: 1.0\n";
$header .= "Content-Type: text/html; charset=iso 8859-1\n";
$header .= "Content-Transfer-Encoding: 8bit\n";
$header .= "X-Mailer: hotelpelangi.co.id (php; 1.00)\n";

$sSQLEmail = "SELECT subject, content FROM tblAutoEmail WHERE
indexname = 'konfirmasi_reservasi' " ;
$hsl = mysql_fetch_array(mysql_query($sSQLEmail)) ;
$sSubject = $hsl["subject"];
$sBody = $hsl["content"];

$sLinkURL = $sitePath . "/room/validasi_reservasi.php?key="
.$sKeyApp ;
$sLinkURL = "<a href=\"". $sLinkURL .
\"target=\"_blank\">\". $sLinkURL. "</a>";

$jumlahTransaksi = "Rp. " . number_format($jumlahTransaksi) ;

$sBody = str_replace('[%tanggal_check_in%]',
DateToString($dCheckIn), $sBody);
$sBody = str_replace('[%tanggal_check_out%]',
DateToString($dCheckOut), $sBody);
$sBody = str_replace('[%nama_pemesan%]', $sNama, $sBody);
$sBody = str_replace('[%link_validasi%]', $sLinkURL, $sBody);
$sBody = str_replace('[%tipe_kamar%]', $sTipeKamar, $sBody);
$sBody = str_replace('[%jumlah_kamar%]', $iJmlh, $sBody);
$sBody = str_replace('[%jumlah_pembayaran%]', $jumlahTransaksi,
$sBody);
$sBody = "<html><head><title>" . $sSubject .
"</title></head><body>" . $sBody . "<body></html>";
$skirim = @mail($to, $sSubject, $sBody, $header);
```

Sebelumnya jika sistem sukses melakukan pengiriman konfirmasi, maka sistem akan menampilkan halaman reservasi sukses, namun jika tidak maka sistem

menampilkan reservasi gagal. Script untuk menampilkan halaman reservasi sukses maupun gagal dengan mengambil dari table section detail dimana mengambil field special ID 2 dan 4, untuk special ID 5 digunakan apabila data reservasi belum lengkap:

```

$$SQL = " SELECT recID FROM tblSectionDetail WHERE sectionID = 3
AND specialID = 2 " ;
$hsl = (mysql_fetch_array(mysql_query($$SQL))) ;
$pageID = $hsl[0] ;
$bReservation = true ;
$ket = $_REQUEST["ket"] ;
if ($ket == "sukses") {
$$SQL = " SELECT recID FROM tblSectionDetail WHERE sectionID = 3
AND specialID = 4 " ;
$hsl = (mysql_fetch_array(mysql_query($$SQL))) ;
$pageID = $hsl[0] ;
$bReservation = false;
} else if ($ket == "gagal") {
$$SQL = " SELECT recID FROM tblSectionDetail WHERE sectionID = 3
AND specialID = 5 " ;
$hsl = (mysql_fetch_array(mysql_query($$SQL))) ;
$pageID = $hsl[0] ;
$bReservation=false;
}
    
```

Dibawah ini merupakan gambar tampilan reservasi gagal dan reservasi sukses yang digunakan untuk informasi konfirmasi reservasi.



Gambar 5.9. Tampilan Reservasi Gagal

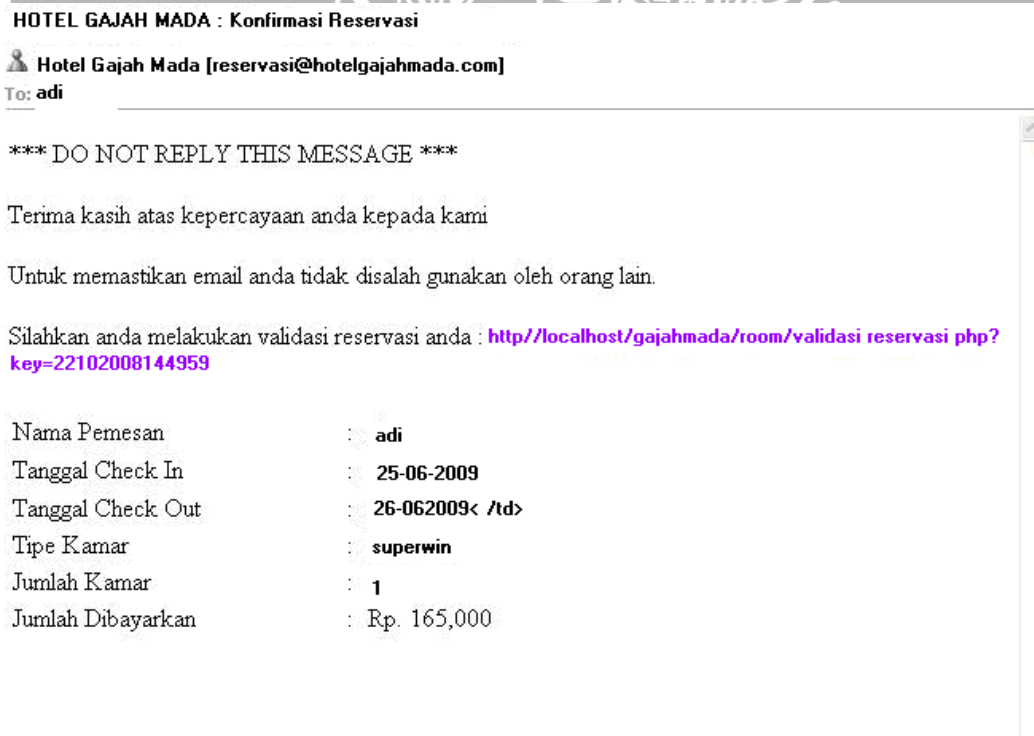
Sumber : Implementasi



Gambar 5.10 Tampilan Resevasi Sukses

Sumber : Implementasi

5.2.3.7. Konfirmasi Reservasi



Gambar 5.11. Email Konfirmasi Resevasi

Sumber : Implementasi

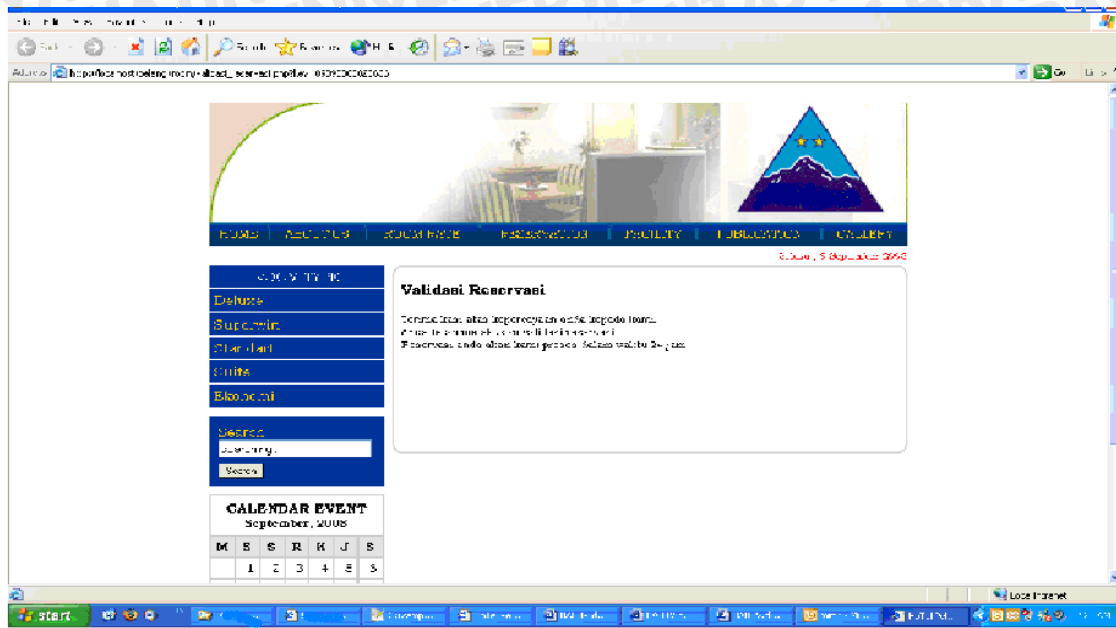
Customer akan mendapatkan email konfirmasi reservasi. Pada email sudah disediakan link untuk melakukan validasi reservasi. Customer tinggal melakukan klik pada link yang tersedia dan nantinya sistem akan melakukan proses validasi reservasi. Pada saat system melakukan melakukan proses validasi, system akan melakukan verifikasi terlebih dahulu, apakah data konfirmasi benar-benar valid atau tidak. Jika data valid maka system akan merubah status reservasi menjadi submitted (stat_reservasi = 2) dan system akan menampilkan halaman konfirmasi reservasi sukses dan jika data tidak valid maka Customer akan dibawa ke halaman homepage. Script untuk melakukan konfirmasi :

```

$$sSQL = " SELECT 1 FROM tblReservasi WHERE stat_reservasi = 0 AND
tgl_in > '" & $currentDate & "' AND appkey = '" . $$sKeyApp . "'"
;
$exeSQL = mysql_query($sSQL);
$numRow = mysql_num_rows($exeSQL);
if($numRow > 0) {
$$sSQL = " UPDATE tblReservasi SET stat_reservasi = 1 WHERE appkey
= '" . $$sKeyApp . "'" ;
if (mysql_query($sSQL)) {
$$sSQL = " SELECT recID FROM tblSectionDetail WHERE sectionID = 3
AND specialID = 3 " ;
$hsl = (mysql_fetch_array(mysql_query($sSQL))) ;
$pageID = $hsl[0] ;
} else {
include($sitePath . "/include/dbDisconnect.inc") ;
header("Location: ../index.php");
}} else {
include($sitePath . "/include/dbDisconnect.inc") ;
header("Location: ../index.php");
}

```

Gambar dibawah ini jika adalah tampilan validasi reservasi sukses.



Gambar 5.12 Validasi Reservasi

Sumber : Implementasi

5.2.3.8. Program Validasi Reservasi oleh Admin

INFORMASI RESERVASI : (SUBMITTED)		<input type="checkbox"/> verified ?
No Reservasi	: HP00004	
Tanggal Reservasi	: 19-10-2008	
Nama Pemesan	: test again	
Email Pemesan	: teas@localhost	
Tanggal Check In	: 23-10-2008	
Tanggal Check Out	: 28-10-2008	
Tipe Kamar	: Deluxe	
Jumlah Kamar	: 3	
Total Transaksi	: Rp. 200,000	

Gambar 5.13 Validasi Reservasi oleh Admin

Sumber : Implementasi

Program validasi Reservasi merupakan program yang dibuat untuk Admin ditujukan customer untuk mevalidasi reservasi saat transaksi dimana stat_reservasi = 1 yaitu masih submitted pada tabel reservasi dan belum diverifikasi. Script dibawah ini

untuk memproses data reservasi dimana mengambil field `stat_reservasi = 1` dan `no_reservasi` pada tabel reservasi.

```

$iReservasiID = $_REQUEST["id"] ;
$sql = " SELECT 1 FROM tblReservasi WHERE stat_reservasi = 1 AND
no_reservasi = '" . $iReservasiID . "' " ;
$exeSQL = mysql_query($sql);
$numRow = mysql_num_rows($exeSQL);
if($numRow <= 0) {
include($path . "/include/dbConnection.inc");
header("Location: index.php") ;}

```

Setelah dilakukan proses maka akan ditampilkan data reservasi, jika kamar tersedia data reservasi akan dilanjutkan. Script dibawah ini untuk menampilkan data reservasi yang berstatus `submitted` atau belum diverifikasi dengan mengambil tabel reservasi dengan tabel type kamar.

```

$sql = "SELECT no_reservasi, tblReservasi.nama AS namapemesan,
email, appkey, tipe_kamar, jml_kamar,
tgl_in, tgl_out, totalTransaksi, tblTypeKamar>Nama AS namakamar
FROM tblReservasi
INNER JOIN tblTypeKamar ON tblTypeKamar.noType =
tblReservasi.tipe_kamar
WHERE no_reservasi = '" . $iReservasiID . "' ;

```

Pada saat admin atau user melakukan cek *verified* pada checkbox yang ada pada halaman validasi reservasi sistem akan melakukan pengecekan pada tabel resevasi apakah ada reservasi pada type kamar dan tanggal yang sama. Setelah itu sistem akan membandingkan jumlah kamar yang dipeservasi dengan jumlah kamar yang tersedia dihotel yang akan menghasilkan jumlah kamar yang bisa dipesan.kamar yang kosong pada tanggal chek in sampai check out pada type kamar yang dipesan, Jika kamar tersedia lebih besar sama dengan jumlah yang dipeservasi maka sistem akan mengubah status reservasi menjadi `verified` (`stat_reservasi = 2`). Kemudian sistem akan mengupdate tabel kamar pada field booking date dengan no kamar sama dengan no kamar yang

direresvasi. Kemudian sistem akan menyimpan no reservasi dan no kamar pada tabel tblreservasiantoroom.

```

$ssSQL = "SELECT jml_kamar FROM tblReservasi WHERE tipe_kamar = "
        . $iTipeKamar . "
        AND (tgl_in >= ' " . $dTanggalIn . "')
        AND no_reservasi <> ' " . $sNoReservasi . "'
        AND no_reservasi NOT IN (SELECT no_reservasi FROM tblReservasi
        WHERE tgl_out <= ' " . $dTanggalIn . "')
        AND stat_reservasi > 1" ;

//check kamar dibooking
$exeSQL = mysql_query($ssSQL);
$hsl = mysql_fetch_array($exeSQL);
$kamar_booked = $hsl["jml_kamar"];
if (mysql_num_rows($exeSQL) == 0) {
    $kamar_booked = 0 ;}

//check jumlah kamar hotel
$ssSQL = "SELECT COUNT(NoKamar) FROM tblRoom WHERE Tipe = " .
        $iTipeKamar ;
$exeSQL = mysql_query($ssSQL);
$hsl = mysql_fetch_array($exeSQL);
$kamar = $hsl["kamar_avail"];

        //jumlah kamar yang tidak di booking
        $available = $kamar - $kamar_booked ;

//pembandingan jika jumlah kamar yang dipesan <= jumlah kamar
yang tersedia maka reservasi sukses
$bSuccess = false;
if ($iJumlah <= $available) {
    $bSuccess = true ;}

if($bSuccess) {$ssSQL = "UPDATE tblReservasi SET stat_reservasi =
2 WHERE no_reservasi = ' " . $iReservasiID . "' " ;
if(mysql_query($ssSQL)) {$ssSQL = "SELECT noKamar FROM tblRoom
WHERE Tipe = " . $iTipeKamar . " AND (BookedDate < ' " .
$dTanggalIn . "' OR BookedDate > ' " . $dTanggalOut . "')
ORDER BY noKamar ASC LIMIT 0, " . $iJumlah ;
$exeSQL = mysql_query($ssSQL);
while($hsl = mysql_fetch_array($exeSQL)) {$ssSQL = "UPDATE tblRoom
SET BookedDate = ' " . $dTanggalIn . "' , nextCheckOut = ' " .
$dTanggalOut . "' WHERE noKamar = ' " . $hsl["noKamar"] . "' " ;
mysql_query($ssSQL);
$ssSQL = "INSERT INTO tblReservasiToRoom VALUES (' " .
$sNoReservasi . " , ' " . $hsl["noKamar"] . " ' ) " ;
mysql_query($ssSQL);}}

```

BAB VI PENGUJIAN

Bab ini membahas proses pengujian terhadap Reservasi Hotel Gajah Mada Ponorogo. Proses pengujian dilakukan melalui dua tahapan (strategi) yaitu pengujian algoritma dan pengujian validasi. Pengujian algoritma menggunakan teknik pengujian *white box* (*white box Testing*). Pengujian validasi menggunakan teknik pengujian *black box* (*black box Testing*).

6.1 Pengujian Algoritma

Teknik pengujian yang dilakukan adalah teknik pengujian kotak putih (*white box testing*) dengan teknik pengujian jalur dasar (*basis path testing*). Pada pengujian jalur dasar akan dilakukan pembuatan *programme design language* (PDL), pembuatan grafik alir (*flow graph*), menentukan kompleksitas siklomatis, menentukan jalur independen dan melakukan kasus uji.

6.1.1 Pengujian Proses Login

1. Pembuatan PDL untuk desain *test case* dengan simpul – simpul yang diidentifikasi

```

PROCEDURE login

Prosedur pengecekan login serta untuk menampilkan menu yang dapat diakses user
TYPE user, password IS STRING

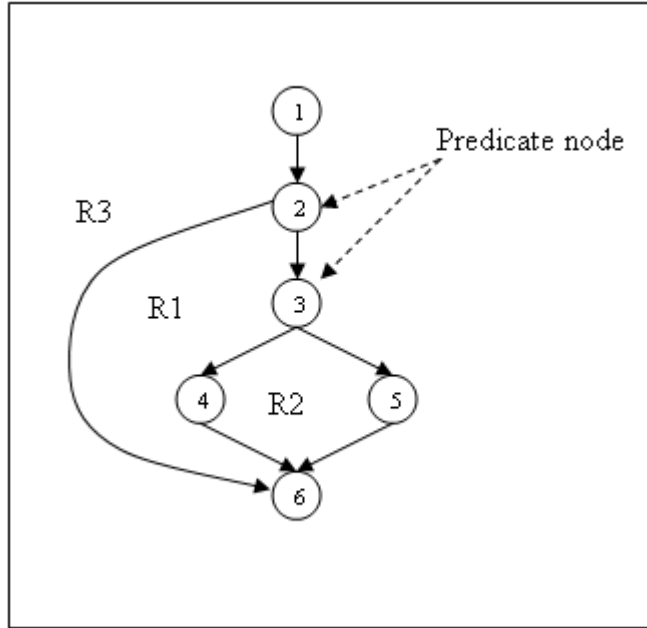
1 { INPUT user_value and password_value
   CONNECT TO database
   SELECT username,password FROM database WHERE username='$user' AND
   password='$password'
2 ← IF (username=user_value and
   3 ← password=password_value) THEN
   4 { SESSION REGISTER as user_value and password_value
     PRINT "login berhasil"
     VIEW menu
5 { ELSE
   PRINT "Login Tidak Berhasil"
6 ← ENDIF

ENDPROC
  
```

Gambar 6.2 PDL Proses Login

Sumber : [Pengujian]

2. Pembuatan grafik alir (*flow graph*) yang sesuai



Gambar 6.3 *Flow Graph* Proses login

Sumber : [Pengujian]

3. Penentuan kompleksitas siklomatis dari grafik alir resultan

$$V(G) = 3 \text{ region}$$

$$V(G) = 7 \text{ edge} - 6 \text{ node} + 2 = 3$$

$$V(G) = 2 \text{ predicate node} + 1 = 3$$

4. Penentuan jalur independen

Jalur 1 : 1 – 2 – 6

Jalur 2 : 1 – 2 – 3 – 5 – 6

Jalur 3 : 1 – 2 – 3 – 4 – 6

5. Melakukan kasus uji

Penentuan kasus uji untuk masing-masing jalur dan hasil eksekusi untuk masing-masing kasus uji dijelaskan pada Tabel 6.1.

Tabel 6.1 Kasus uji untuk pengujian proses login

Jalur	Kasus uji	Hasil yang diharapkan	Hasil yang didapatkan
1	username = null or username != username value	User gagal login	User gagal login
2	Password = null or password != password value	User gagal login dan keluar pesan error	User gagal login dan keluar pesan error
3	Username = username value and password = password value	User sukses login dan tampilkan menu	User sukses login dan tampilkan menu

Sumber : [Pengujian]

- Analisis : saat melakukan pengujian proses login username dan password, data yang dimasukkan tidak boleh kosong atau salah(tidak tersedia dalam database). Proses login akan berhasil dilakukan jika username dan password tidak kosong dan tersedia dalam database.

No	Nama kasus uji	Hasil yang didapat	Status
1	Proses login	User sukses login dan tampilkan menu.	ok

Sumber : [Pengujian]

6.1.2 Pengujian Proses Tambah Data

1. Pembuatan PDL untuk desain *test case* dengan simpul – simpul yang diidentifikasi

```

PROCEDURE tambah data

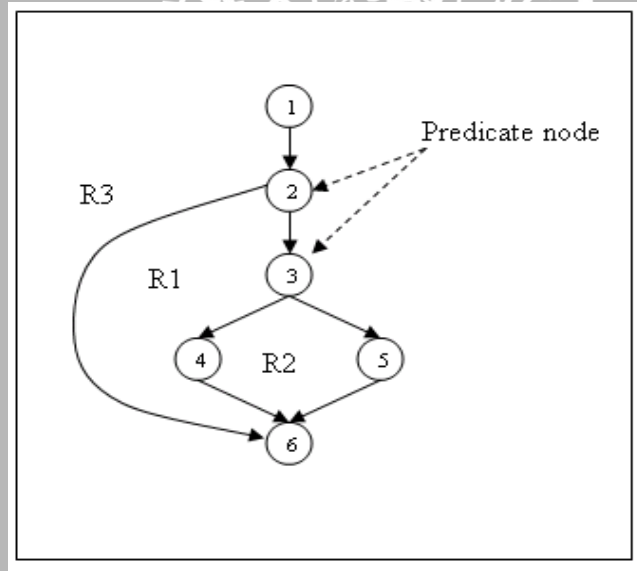
Prosedur untuk menambahkan data
TYPE id, nama_data, isi_data IS STRING
1 ← INPUT nama_data value and isi_data value
2 ← { CONNECT TO database
      SQL INSERT INTO database VALUES(id value, nama_data value and
      isi_data value)
3 ← IF (sukses) THEN
4 ← PRINT "Data Telah Ditambahkan"
5 ← { ELSE
      PRINT "Data Gagal Ditambahkan"
6 ← ENDIF

ENDPROC
    
```

Gambar 6.4 PDL Proses Tambah Data

Sumber : [Pengujian]

2. Pembuatan grafik alir yang sesuai



Gambar 6.5 Flow Graph Proses Tambah Data

Sumber : [Pengujian]

3. Penentuan kompleksitas siklomatis dari grafik alir resultan

$$V(G) = 3 \text{ region}$$

$$V(G) = 7 \text{ edge} - 6 \text{ node} + 2 = 3$$

$$V(G) = 2 \text{ predicate node} + 1 = 3$$

4. Penentuan jalur independen

Jalur 1 : 1 – 2 – 6

Jalur 2 : 1 – 2 – 3 – 4 – 6

Jalur 3 : 1 – 2 – 3 – 5 – 6

5. Melakukan kasus uji

Penentuan kasus uji untuk masing-masing jalur dan hasil eksekusi untuk masing-masing kasus uji dijelaskan pada Tabel 6.2.

Tabel 6.2 Kasus uji untuk pengujian proses tambah data

Jalur	Kasus uji	Hasil yang diharapkan	Hasil yang didapatkan
1	Data value = null	Data yang dimasukkan kosong	Data yang dimasukkan kosong
2	Id_data value != null and data value != null and query sql "gagal"	Data gagal ditambahkan	Data gagal ditambahkan
3	Id_data value != null and data value != null and query sql "sukses"	Data berhasil ditambahkan	Data berhasil ditambahkan

Sumber : [Pengujian]

- Analisis : hasil pengujian proses tambah data akan berhasil dilakukan apabila query berjalan serta id_data yang dimasukkan sesuai dengan database yang ada.

No	Nama kasus uji	Hasil yang didapat	Status
1	Proses tambah data	Data berhasil ditambahkan	ok

Sumber : [Pengujian]

6.1.3 Pengujian Proses Edit Data

1. Pembuatan PDL untuk desain *test case* dengan simpul – simpul yang diidentifikasi

```

PROCEDURE edit data

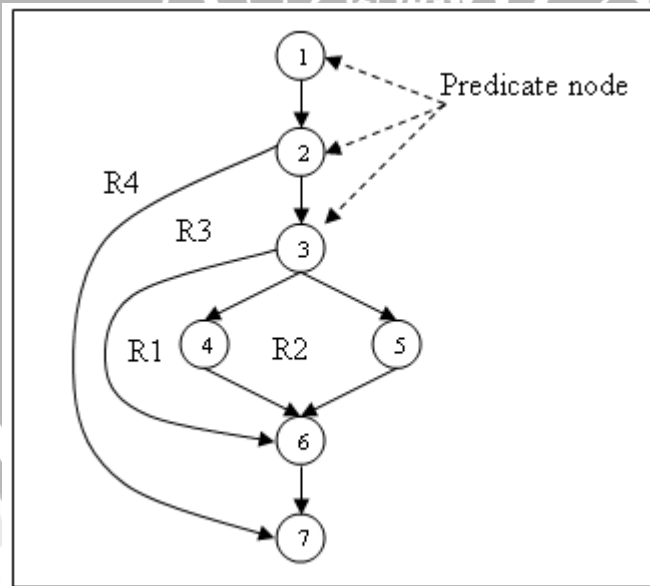
Prosedur untuk mengedit data
TYPE id, data IS STRING

1 { CONNECT TO database
   IF (GET(id))
   SQL SELECT * FROM datab           alue
2 ← IF (data update found) THEN
3 { SQL UPDATE FROM database WHERE id = id value
   IF (update success) THEN
4 ← PRINT "Data Telah Diubah";
5 { ELSE
   PRINT "Data Gagal Diubah";
6 ← ENDFIF
7 ← ENDFIF

ENDPROC
    
```

Gambar 6.6 PDL Proses Edit Data
 Sumber : [Pengujian]

2. Pembuatan grafik alir yang sesuai



Gambar 6.7 Flow Graph Proses Edit Data
 Sumber : [Pengujian]

3. Penentuan kompleksitas siklomatis dari grafik alir resultan

$$V(G) = 4 \text{ region}$$

$$V(G) = 9 \text{ edge} - 7 \text{ node} + 2 = 4$$

$$V(G) = 3 \text{ predicate node} + 1 = 4$$

4. Penentuan jalur independen

Jalur 1 : 1 – 2 – 7

Jalur 2 : 1 – 2 – 3 – 6 – 7

Jalur 3 : 1 – 2 – 3 – 4 – 6 – 7

Jalur 4 : 1 – 2 – 3 – 5 – 6 – 7

5. Melakukan kasus uji

Penentuan kasus uji untuk masing-masing jalur dan hasil eksekusi untuk masing-masing kasus uji dijelaskan pada Tabel 6.3.

Tabel 6.3 Kasus uji untuk pengujian proses edit data

Jalur	Kasus uji	Hasil yang diharapkan	Hasil yang didapatkan
1	SQL update "gagal"	Data gagal diupdate	Data gagal diupdate
2	SQL update "sukses"	Data berhasil diupdate	Data berhasil diupdate

Sumber : [Pengujian]

- Analisis : hasil pengujian proses edit data akan berhasil dilakukan data yang dimasukkan sesuai dan terhubung dengan database yang ada.

No	Nama kasus uji	Hasil yang didapat	Status
1	Proses login	User sukses login dan tampilkan menu.	ok

Sumber : [Pengujian]

6.1.4 Pengujian Proses Hapus Data

1. Pembuatan PDL untuk desain *test case* dengan simpul – simpul yang diidentifikasi

```

PROCEDURE hapus data

Prosedur untuk menghapus data
TYPE id IS STRING
CONNECT TO database

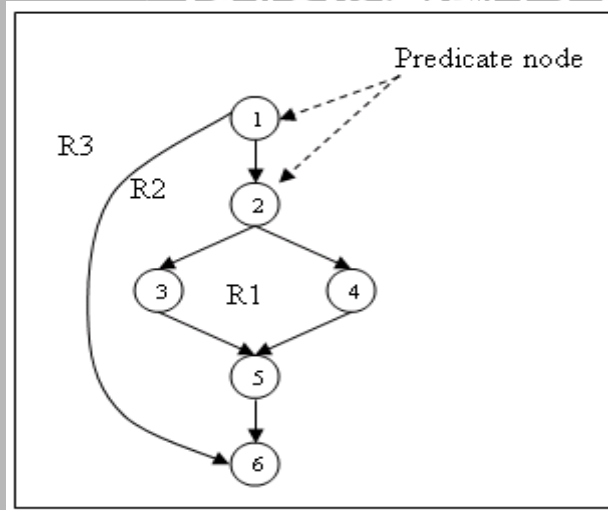
1  { IF (GET (id))
    {   SQL DELETE FROM datab;
2  ← IF (delete success) THEN
3  ← PRINT "Data Telah Dihapus";
4  { ELSE
    {   PRINT "Data Gagal Dihapus";
5  ← ENDIF
6  ← ENDIF

ENDPROC
    
```

Gambar 6.8 PDL Proses Hapus Data

Sumber : [Pengujian]

2. Pembuatan grafik alir yang sesuai



Gambar 6.9 Flow Graph Proses Hapus Data

Sumber : [Pengujian]

3. Penentuan kompleksitas siklomatis dari grafik alir resultan

$$V(G) = 3 \text{ region}$$

$$V(G) = 7 \text{ edge} - 6 \text{ node} + 2 = 3$$

$$V(G) = 2 \text{ predicate node} + 1 = 3$$

4. Penentuan jalur independen

Jalur 1 : 1 – 6

Jalur 2 : 1 – 2 – 4 – 5 – 6

Jalur 3 : 1 – 2 – 3 – 5 – 6

5. Melakukan kasus uji

Penentuan kasus uji untuk masing-masing jalur dan hasil eksekusi untuk masing-masing kasus uji dijelaskan pada Tabel 6.4.

Table 6.4 Kasus uji untuk pengujian proses hapus data

Jalur	Kasus uji	Hasil yang diharapkan	Hasil yang didapatkan
1	Id_data value = null	Obyek list dengan jumlah elemen 0.	Obyek list dengan jumlah elemen 0
2	SQL delete "gagal"	Data gagal dihapus	Data gagal dihapus
3	SQL delete "sukses"	Data berhasil dihapus	Data berhasil dihapus

Sumber : [Pengujian]

Analisis : hasil pengujian proses hapus data, akan berhasil dilakukan apabila id_data yang dimasukkan sesuai dengan database yang ada.

No	Nama kasus uji	Hasil yang didapat	Status
1	Proses hapus data	Data berhasil di hapus.	ok

Sumber : [Pengujian]

6.2 Pengujian Validasi

Pengujian validasi digunakan untuk mengetahui apakah sistem yang dibangun sudah benar sesuai dengan yang dibutuhkan. Item-item yang telah dirumuskan dalam daftar kebutuhan dan merupakan hasil analisis kebutuhan akan menjadi acuan untuk melakukan pengujian validasi. Pengujian validasi menggunakan metode pengujian *Black Box*, karena tidak memerlukan untuk berkonsentrasi terhadap alur jalannya algoritma

program dan lebih ditekankan untuk menemukan konformitas antara kinerja sistem dengan daftar kebutuhan.

6.2.1 Kasus Uji Validasi

Untuk mengetahui kesesuaian antara kebutuhan dengan kinerja sistem, pada setiap kebutuhan (*requirement*) dilakukan proses pengujian dengan kasus uji masing-masing.

6.2.1.1 Kasus Uji Login

Kasus uji login ditunjukkan dalam Tabel 6.6a dan Tabel 6.6b.

Tabel 6.6a Kasus uji login dimana *username* dan *password* ada dalam sistem

Nama kasus uji	Kasus uji login
Obyek uji	Kebutuhan login ke sistem
Tujuan pengujian	Menguji bahwa hanya pengguna yang telah terdaftar dapat masuk dan mengakses fasilitas tertentu. Pengguna sistem yang tidak login hanya akan menjadi <i>user</i> biasa.
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem telah berjalan. 2. Memasukkan <i>username</i> dan <i>password</i> pada form login (<i>username</i> = "admin", <i>password</i> = "admin"). 3. Menekan tombol login.
Hasil yang diharapkan	Pengguna berhasil login pada sistem.

Sumber : [Pengujian]

- Analisis : proses login berhasil, apabila *username* dan *password* terdaftar, jika tidak maka akan dianggap sebagai user biasa.

No	Nama kasus uji	Hasil yang didapat	Status
1	Login dimana <i>username</i> dan <i>password</i> ada dalam sistem	Pengguna berhasil login pada sistem.	Ok

Sumber : [Pengujian]

Tabel 6.6b Kasus uji login dimana *username* dan *password* tidak ada dalam sistem

Nama kasus uji	Kasus uji login dimana <i>username</i> dan <i>password</i> tidak ada dalam sistem
Obyek uji	Kebutuhan login ke sistem
Tujuan pengujian	Menguji bahwa sistem dapat menampilkan pesan bahwa login tidak berhasil.
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem telah berjalan. 2. Memasukkan <i>username</i> dan <i>password</i> pada form login (<i>username</i>="xxx", <i>password</i> = "xxx"). 3. Menekan tombol login.
Hasil yang diharapkan	Menampilkan pesan login tidak berhasil.

Sumber : [Pengujian]

- Analisis : pesan proses login tidak berhasil akan ditampilkan apabila *username* dan *password* tidak tersedia.

No	Nama kasus uji	Hasil yang didapat	Status
1	Login dimana <i>username</i> dan <i>password</i> tidak ada dalam sistem	Menampilkan pesan login tidak berhasil.	Ok

Sumber : [Pengujian]

6.2.2.2 Kasus Uji Logout

Kasus uji logout ditunjukkan dalam Tabel 6.7.

Tabel 6.7 Kasus uji logout

Nama kasus uji	Kasus uji logout
Obyek uji	Kebutuhan logout dari sistem
Tujuan pengujian	Menguji bahwa sistem dapat menyediakan fasilitas bagi <i>user</i> yang telah login untuk keluar dari sistem
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem telah berjalan dan <i>user</i> telah <i>login</i> 2. <i>User</i> menekan tombol logout
Hasil yang diharapkan	<i>User</i> berhasil keluar dari sistem

Sumber : [Pengujian]

- Analisis : setelah user berhasil masuk dalam sistem, user akan keluar dari sistem jika, user memilih menu logout.

No	Nama kasus uji	Hasil yang didapat	Status
1	Logout	User berhasil keluar dari sistem	Ok

Sumber : [Pengujian]

6.2.2.3 Kasus Uji Tambah User

Kasus uji tambah user yang hanya bisa dilakukan oleh admin ditunjukkan dalam Tabel

6.8a, Tabel 6.8b dan Tabel 6.8c.

Tabel 6.8a Kasus uji tambah user

Nama kasus uji	Kasus uji tambah user
Obyek uji	Kebutuhan menambah user
Tujuan pengujian	Menguji bahwa sistem dapat menyediakan fasilitas untuk menambah user. Data yang harus dicatat dari penambahan user adalah username, password.
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem telah berjalan dan user telah login sebagai admin. 2. Admin memilih menu "user menu". 3. Memilih <i>tab</i> tambah user. 4. Memasukkan data-data pada <i>form</i> (username = "pengajar1", password = "pengajar1"). 5. Menekan tombol proses.
Hasil yang diharapkan	User anggota berhasil ditambahkan

Sumber : [Pengujian]

- Analisis : data user baru akan berhasil dilakukan, bila user tersebut sudah mendapat konfirmasi dari admin.

No	Nama kasus uji	Hasil yang didapat	Status
1	Tambah user	User anggota berhasil ditambahkan	Ok

Sumber : [Pengujian]

Tabel 6.8b Kasus uji tambah user dimana masukan data kosong

Nama kasus uji	Kasus uji tambah user dimana masukan data kosong
Obyek uji	Kebutuhan menambah user
Tujuan pengujian	Menguji bahwa sistem dapat menampilkan pesan kesalahan bahwa <i>username</i> dan <i>password</i> tidak boleh kosong
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem telah berjalan dan user telah login sebagai admin. 2. Admin memilih menu “user menu”. 3. Memilih <i>tab</i> tambah user. 4. Memasukkan data kosong pada <i>form</i> tambah user. 5. Menekan tombol proses.
Hasil yang diharapkan	Sistem menampilkan pesan kesalahan bahwa tambah user gagal.

Sumber : [Pengujian]

- Analisis : apabila dalam penambahan user, data username dan password kosong, maka akan menampilkan pesan, ada kesalahan username dan password

No	Nama kasus uji	Hasil yang didapat	Status
1	Tambah user dimana masukan data kosong	Sistem menampilkan pesan kesalahan bahwa tambah user gagal	Ok

Sumber : [Pengujian]

Tabel 6.8c Kasus uji tambah user dimana *username* sudah ada

Nama kasus uji	Kasus uji tambah user dimana masukan <i>username</i> sudah ada dalam sistem
Obyek uji	Kebutuhan menambah user
Tujuan pengujian	Menguji bahwa sistem dapat menampilkan pesan kesalahan bahwa <i>username</i> sudah ada dalam sistem
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem telah berjalan dan user telah login sebagai admin. 2. Admin melihat menu “user menu”. <ol style="list-style-type: none"> 1. Memilih <i>tab</i> tambah user 2. Memasukkan data-data pada <i>form</i> (<i>username</i>=”pengajar”, <i>password</i>=”pengajar” dimana <i>username</i>=”pengajar” sudah ada dalam sistem). 3. Menekan tombol proses.
Hasil yang diharapkan	Sistem menampilkan pesan kesalahan bahwa <i>username</i> sudah ada dalam sistem

Sumber : [Pengujian]

- Analisis : apabila dalam penambahan user, *username* sudah ada sebelumnya dalam database, maka akan muncul pesan ada kesalahan *username*.

No	Nama kasus uji	Hasil yang didapat	Status
1	Tambah user dimana <i>username</i> sudah ada	Sistem menampilkan pesan kesalahan bahwa <i>username</i> sudah ada dalam sistem	Ok

Sumber : [Pengujian]

6.2.1.4 Kasus Uji Melihat Daftar User

Kasus uji melihat daftar user yang hanya bisa dilakukan oleh admin ditunjukkan dalam

Tabel 6.9.

Tabel 6.9a Kasus uji melihat daftar user

Nama kasus uji	Kasus uji melihat daftar user
Obyek uji	Kebutuhan menampilkan daftar user
Tujuan pengujian	Menguji bahwa sistem dapat menampilkan semua user yang tercatat di dalam sistem.
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem telah berjalan dan <i>user</i> telah login sebagai Admin. 2. Memilih menu “user menu”. 3. admin memilih menu edit dan hapus menu.
Hasil yang diharapkan	Semua daftar user dapat ditampilkan.

Sumber : [Pengujian]

- analisis : admin akan bisa melihat daftar user, apabila sudah login sebagai admin dan memilih menu, user menu.

No	Nama kasus uji	Hasil yang didapat	Status
1	Melihat daftar user	Semua daftar user dapat ditampilkan.	Ok

Sumber : [Pengujian]

6.2.1.5 Kasus Uji Mengedit Data User

Kasus uji mengedit data user yang hanya bisa dilakukan oleh admin ditunjukkan dalam Tabel 6.10a, Tabel 6.10b dan Tabel 6.10c.

Tabel 6.10a Kasus uji mengedit data user

Nama kasus uji	Kasus uji mengedit data user
Obyek uji	Kebutuhan mengedit data user
Tujuan pengujian	Menguji bahwa sistem dapat mengedit data user.
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem telah berjalan dan <i>user</i> telah login 2. Memilih menu edit. 3. Memasukkan data pada <i>form</i> (<i>username</i>="pengajar2 edit", <i>password</i>=" pengajar2 edit") 4. Menekan tombol edit
Hasil yang diharapkan	Data user dapat diedit dan disimpan.

Sumber : [Pengujian]

- Analisis : user akan berhasil melakukan proses edit data user, apabila sudah login terlebih dahulu dan memilih menu edit user.

No	Nama kasus uji	Hasil yang didapat	Status
1	Mengedit data user	Data user dapat diedit dan disimpan.	Ok

Sumber : [Pengujian]

Tabel 6.10b Kasus uji mengedit data user dimana *username* tidak boleh kosong

Nama kasus uji	Kasus uji mengedit data dimana <i>username</i> tidak boleh kosong
Obyek uji	Kebutuhan mengedit data user
Tujuan pengujian	Menguji bahwa sistem dapat menampilkan pesan kesalahan bahwa data <i>username</i> tidak boleh kosong.
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem telah berjalan dan <i>user</i> telah login sebagai Admin. 2. Memilih menu edit dan hapus user. 3. Memasukkan data pada form (<i>username=""</i>) 4. Menekan tombol edit
Hasil yang diharapkan	Menampilkan pesan kesalahan gagal edit user

Sumber : [Pengujian]

- Analisis : apabila dalam melakukan proses edit user, data yang dimasukkan kosong, maka akan ada pesan, edit user gagal.

No	Nama kasus uji	Hasil yang didapat	Status
1	Mengedit data user dimana <i>username</i> tidak boleh kosong	Menampilkan pesan kesalahan gagal edit user.	Ok

Sumber : [Pengujian]

Tabel 6.10c Kasus uji mengedit data user dimana *username* sudah ada

Nama kasus uji	Kasus uji mengedit data user dimana <i>username</i> sudah ada
Obyek uji	Kebutuhan mengedit data user
Tujuan pengujian	Menguji bahwa system dapat menampilkan pesan kesalahan bahwa <i>username</i> sudah ada.
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem telah berjalan dan <i>user</i> telah login.

	<ol style="list-style-type: none"> 2. Memilih menu edit dan hapus user. 3. Memasukkan data pada <i>form</i> (<i>username</i>="pengajar2") dimana telah ada <i>username</i> pengajar 4. Menekan tombol edit
Hasil yang diharapkan	Menampilkan pesan kesalahan bahwa <i>username</i> sudah ada dalam sistem

Sumber : [Pengujian]

- Analisis : dalam melakukan proses edit user, *username* tidak boleh sama.

No	Nama kasus uji	Hasil yang didapat	Status
1	Mengedit data user dimana <i>username</i> sudah ada	Menampilkan pesan kesalahan bahwa <i>username</i> sudah ada dalam sistem	Ok

Sumber : [Pengujian]

6.2.2.6 Kasus Uji Menghapus User

Kasus uji menghapus user yang hanya bisa dilakukan oleh admin ditunjukkan dalam Tabel 6.11.

Tabel 6.11 Kasus uji menghapus data user

Nama kasus uji	Kasus uji menghapus data user
Obyek uji	Kebutuhan menghapus data user
Tujuan pengujian	Menguji bahwa sistem dapat menghapus data user
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem telah berjalan dan <i>user</i> telah login sebagai Admin. 2. Memilih menu edit dan hapus user. 3. Memilih user yang akan dihapus dan menekan tombol "hapus".
Hasil yang diharapkan	User bisa dihapus dari sistem

Sumber : [Pengujian]

- Analisis : proses menghapus user akan berhasil dilakukan jika user melakukan login sebagai admin dan memilih menu hapus user.

No	Nama kasus uji	Hasil yang didapat	Status
1	Menghapus data user	User bisa dihapus dari sistem	Ok

Sumber : [Pengujian]

6.2.2.7 Kasus Uji Modifikasi Data User

Kasus uji modifikasi data user ditunjukkan dalam Tabel 6.12a, Tabel 6.12b.

Tabel 6.12a Kasus uji modifikasi data user

Nama kasus uji	Kasus uji modifikasi data user
Obyek uji	Kebutuhan modifikasi data user
Tujuan pengujian	Menguji bahwa sistem dapat memodifikasi data user(nama dan password).
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem telah berjalan dan <i>user</i> telah login sebagai Admin atau user anggota. 2. Memilih menu modifikasi account. 3. Memasukkan dan merubah data pada <i>form</i>. 4. Menekan tombol proses.
Hasil yang diharapkan	Data user berhasil dimodifikasi dan disimpan.

Sumber : [Pengujian]

- Analisis : proses modifikasi user akan berhasil dilakukan apabila sudah login terlebih dahulu sebagai user atau admin.

No	Nama kasus uji	Hasil yang didapat	Status
1	Modifikasi data user	Data user berhasil dimodifikasi dan disimpan	Ok

Sumber : [Pengujian]

Tabel 6.12b Kasus uji modifikasi data user dimana password aktif tidak cocok.

Nama kasus uji	Kasus uji modifikasi data dimana password aktif salah
Obyek uji	Kebutuhan modifikasi data user

Tujuan pengujian	Menguji bahwa sistem dapat menampilkan pesan kesalahan jika password aktif salah.
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem telah berjalan dan <i>user</i> telah login sebagai admin atau user anggota. 2. Memilih menu modifikasi account. 3. Memasukkan data pada <i>form</i> (password aktif salah) 4. Menekan tombol proses.
Hasil yang diharapkan	Menampilkan pesan kesalahan password aktif tidak cocok.

Sumber : [Pengujian]

- Analisis : user tidak dapat melakukan modifikasi jika ada kesalahan username dan password.

No	Nama kasus uji	Hasil yang didapat	Status
1	Modifikasi data user dimana password aktif tidak cocok.	Menampilkan pesan kesalahan password aktif tidak cocok.	Ok

Sumber : [Pengujian]

6.2.2.8 Kasus Uji Menambah Artikel

Kasus uji menambah artikel ditunjukkan dalam Tabel 6.18a dan Tabel 6.18b.

Tabel 6.18a Kasus uji menambah artikel

Nama kasus uji	Kasus uji menambah artikel
Obyek uji	Kebutuhan menambah artikel
Tujuan pengujian	Menguji bahwa sistem dapat menambah artikel. Data yang harus dicatat dari penambahan artikel adalah judul, isi artikel.
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem telah berjalan dan <i>user</i> telah login sebagai admin atau user anggota. 2. Memilih <i>tab</i> artikel. 3. Memilih tambah artikel 4. Memasukkan data-data pada <i>form</i> tambah artikel 5. Menekan tombol simpan
Hasil yang diharapkan	Artikel baru berhasil dibuat

Sumber : [Pengujian]

- Analisis : proses penambahan artikel akan berhasil dilakukan apabila sudah terlebih dahulu login.

No	Nama kasus uji	Hasil yang didapat	Status
1	Menambah artikel	Artikel baru berhasil dibuat	Ok

Sumber : [Pengujian]

Tabel 6.18b Kasus uji menambah Artikel dengan masukan judul kosong

Nama kasus uji	Kasus uji tambah artikel dimana masukan judul kosong
Obyek uji	Kebutuhan menambah artikel
Tujuan pengujian	Menguji bahwa sistem dapat menampilkan pesan kesalahan bahwa judul tidak boleh kosong
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem telah berjalan dan <i>user</i> telah login sebagai admin atau user anggota. 2. Memilih <i>tab</i> artikel. 3. Memilih <i>tab</i> tambah artikel. 4. Memasukkan data pada <i>form</i> (judul = “ ”) 5. Menekan tombol simpan.
Hasil yang diharapkan	Sistem menampilkan pesan kesalahan bahwa artikel gagal ditambahkan.

Sumber : [Pengujian]

- Analisis : data yang dimasukkan tidak boleh kosong, apabila data yg dimasukkan kosong maka akan muncul pesan, artikel gagal ditambahkan.

No	Nama kasus uji	Hasil yang didapat	Status
1	Menambah Artikel dengan masukan judul kosong	Sistem menampilkan pesan kesalahan bahwa artikel gagal ditambahkan	Ok

Sumber : [Pengujian]

6.2.2.9 Kasus Uji Mengedit Artikel

Kasus uji mengedit artikel ditunjukkan dalam Tabel 6.19.

Tabel 6.19 Kasus uji mengedit artikel

Nama kasus uji	Kasus uji mengedit artikel
Obyek uji	Kebutuhan mengedit artikel
Tujuan pengujian	Menguji bahwa sistem dapat mengedit artikel.
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem telah berjalan dan <i>user</i> telah login sebagai Admin atau user anggota. 2. Memilih menu edit pada kolom artikel. 3. Memasukkan data pada <i>form</i> (judul="artikel2edit", isi="isi artikel2edit") 4. Menekan tombol simpan
Hasil yang diharapkan	Data artiel berhasil diedit dan disimpan.

Sumber : [Pengujian]

- Analisis : untuk melakukan proses edit, user terlebih dahulu login dan memilig menu edit artikel.

No	Nama kasus uji	Hasil yang didapat	Status
1	Mengedit artikel	Data artiel berhasil diedit dan disimpan.	Ok

Sumber : [Pengujian]

6.2.2.10 Kasus Uji Hapus Artikel

Kasus uji menghapus artikel ditunjukkan dalam Tabel 6.20.

Tabel 6.20 Kasus uji menghapus artikel

Nama kasus uji	Kasus uji menghapus artikel
Obyek uji	Kebutuhan menghapus artikel
Tujuan pengujian	Menguji bahwa sistem dapat menghapus artikel
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem telah berjalan dan <i>user</i> telah login sebagai Admin. 2. Memilih menu hapus pada kolom artikel.
Hasil yang diharapkan	Data artikel berhasil dihapus

Sumber : [Pengujian]

- Analisis : untuk melakukan proses menghapus artikel, user terlebih dahulu login dan memilih menu hapus artikel.

No	Nama kasus uji	Hasil yang didapat	Status
1	Menghapus artikel	Data artikel berhasil dihapus	Ok

Sumber : [Pengujian]

6.2.2.11 Kasus Uji Membuat Topik Forum

Kasus uji membuat topik forum ditunjukkan dalam Tabel 6.21a dan Tabel 6.21b.

Tabel 6.21a Kasus uji membuat topik forum

Nama kasus uji	Kasus uji membuat topik forum
Obyek uji	Kebutuhan membuat topik baru forum
Tujuan pengujian	Menguji bahwa sistem dapat membuat topik forum. Data yang harus dicatat dari pembuatab topik forum adalah nama, subjek dan isi forum.
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem telah berjalan dan <i>user</i> telah login sebagai admin atau user anggota. 2. Memilih <i>tab</i> topik baru. 3. Memasukkan data-data pada <i>form</i> topik baru 4. Menekan tombol simpan
Hasil yang diharapkan	Topik forum baru berhasil ditambahkan

Sumber : [Pengujian]

- Analisis : untuk membuat topik baru dalam forum, user terlebih dahulu login, kemudian memilih menu forum.

No	Nama kasus uji	Hasil yang didapat	Status
1	Membuat topik forum	Topik forum baru berhasil ditambahkan	Ok

Sumber : [Pengujian]

Tabel 6.21b Kasus uji membuat topik forum dengan masukan judul kosong

Nama kasus uji	Kasus uji membuat topik forum dimana masukan judul kosong
Obyek uji	Kebutuhan membuat topik forum

Tujuan pengujian	Menguji bahwa sistem dapat menampilkan pesan kesalahan bahwa judul tidak boleh kosong
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem telah berjalan dan <i>user</i> telah login sebagai admin atau user anggota. 2. Memilih <i>tab</i> topik baru. 3. Memasukkan data pada <i>form</i> (judul = “ “) 4. Menekan tombol simpan.
Hasil yang diharapkan	Sistem menampilkan pesan kesalahan bahwa topik baru gagal ditambahkan.

Sumber : [Pengujian]

- Analisis : untuk membuat topik baru dalam forum, user terlebih dahulu login, kemudian memilih menu forum, judul forum tidak boleh kosong.

No	Nama kasus uji	Hasil yang didapat	Status
1	Membuat topik forum dengan masukan judul kosong	Sistem menampilkan pesan kesalahan bahwa topik baru gagal ditambahkan.	Ok

Sumber : [Pengujian]

6.2.2.12 Kasus Uji Membalas Topik Forum

Kasus uji membalas topik forum ditunjukkan dalam Tabel 6.22.

Tabel 6.22 Kasus uji membalas topik forum

Nama kasus uji	Kasus uji membalas topik forum
Obyek uji	Kebutuhan membalas topik forum
Tujuan pengujian	Menguji bahwa sistem dapat membalas topik dala forum.
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem telah berjalan dan <i>user</i> telah login sebagai Admin atau user anggota. 2. Memilih salah satu topik forum. 3. Memilih tab replay. 4. Memasukkan data pada <i>form</i> replay. 5. Menekan tombol simpan.
Hasil yang diharapkan	Data replay topik forum dapat tersimpan dan ditampilkan.

Sumber : [Pengujian]

- Analisis : untuk me-reply, user terlebih dahulu login, kemudian memilih menu forum.

No	Nama kasus uji	Hasil yang didapat	Status
1	Membalas topik forum	Data replay topik forum dapat tersimpan dan ditampilkan.	Ok

Sumber : [Pengujian]

6.2.2.13 Kasus Uji Hapus Topik Forum

Kasus uji menghapus topik forum ditunjukkan dalam Tabel 6.23.

Tabel 6.23 Kasus uji menghapus topik forum

Nama kasus uji	Kasus uji menghapus topik forum
Obyek uji	Kebutuhan menghapus topik forum
Tujuan pengujian	Menguji bahwa sistem dapat menghapus topik forum
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem telah berjalan dan <i>user</i> telah login sebagai Admin atau user anggota 2. Memilih menu hapus pada topik forum.
Hasil yang diharapkan	Data topik forum berhasil dihapus

Sumber : [Pengujian]

- analisis : untuk menghapus topik forum, user harus terlebih dahulu login.

No	Nama kasus uji	Hasil yang didapat	Status
1	Menghapus topik forum	Data topik forum berhasil dihapus.	Ok

Sumber : [Pengujian]

6.2.3 Hasil Pengujian Validasi

Dari kasus uji yang telah dilaksanakan sesuai dengan prosedur pengujian pada sub pokok bahasan 6.2.1, didapatkan hasil seperti ditunjukkan pada Tabel 6.34.

Tabel 6.34 Hasil Pengujian Validasi

No	Nama kasus uji	Hasil yang didapat	Status Validitas
1	Kasus uji <i>login</i>	Pengguna berhasil login pada sistem.	Valid
2	Kasus uji login dimana <i>username</i> dan <i>password</i> tidak ada dalam sistem	Menampilkan pesan login tidak berhasil.	Valid
3	Kasus uji <i>logout</i>	Pengguna berhasil keluar dari sistem.	Valid
4	Kasus uji tambah user	User anggota berhasil ditambahkan.	Valid
5	Kasus uji tambah user dimana masukan data kosong	Sistem menampilkan pesan kesalahan bahwa tambah user gagal.	Valid
6	Kasus uji tambah user dimana <i>username</i> sudah ada	Menampilkan pesan kesalahan bahwa <i>username</i> sudah ada dalam sistem	Valid
7	Kasus uji melihat daftar user	Semua daftar user dapat ditampilkan.	Valid
8	Kasus uji mengedit data user	Data user dapat diedit dan disimpan.	Valid
9	Kasus uji mengedit data user dimana <i>username</i> tidak boleh kosong	Menampilkan pesan kesalahan gagal edit user	Valid
10	Kasus uji mengedit data	Menampilkan pesan kesalahan bahwa <i>username</i> sudah ada dalam sistem	Valid

	user dimana <i>username</i> sudah ada		
11	Kasus uji menghapus user	User bisa dihapus dari sistem.	Valid
12	Kasus uji modifikasi data user	Data user berhasil dimodifikasi dan disimpan.	Valid
13	Kasus uji modifikasi data user dimana password aktif tidak cocok.	Menampilkan pesan kesalahan password aktif tidak cocok.	Valid
14	Kasus uji menambah artikel	Artikel baru berhasil dibuat	Valid
15	Kasus uji menambah Artikel dengan masukan judul kosong	Sistem menampilkan pesan kesalahan bahwa artikel gagal ditambahkan.	Valid
16	Kasus uji mengedit artikel	Data artikel berhasil diedit dan disimpan.	Valid
17	Kasus uji menghapus artikel	Data artikel berhasil dihapus	Valid
18	Kasus uji membuat topik forum	Topik forum baru berhasil ditambahkan	Valid
19	Kasus uji membuat topik forum dengan masukan judul kosong	Sistem menampilkan pesan kesalahan bahwa topik baru gagal ditambahkan.	Valid
20	Kasus uji membalas topik forum	Data replay topik forum dapat tersimpan dan ditampilkan.	Valid
21	Kasus uji menghapus topik forum	Data topik forum berhasil dihapus	Valid

Sumber : [Pengujian]

Dari 21 kasus uji yang ditunjukkan dalam Tabel 6.34 dapat disimpulkan bahwa sistem yang dibangun tidak terdapat kesalahan dan terjamin validitasnya.

Pengujian Kecepatan Query

a. Tujuan

Pengujian dilakukan untuk mengetahui waktu yang dibutuhkan untuk melakukan sebuah *query* pada *database* sistem Reservasi. Pengujian kecepatan *query* ini menggunakan contoh *query* lihat data soal berdasarkan *block_id* soal.

Pengujian dilakukan untuk mendapatkan perbandingan waktu *query* yang dilakukan terhadap jumlah data yang berbeda pada *database* Reservasi.

b. Prosedur Pengujian

PC Client:

- Mengakses `query_tes.php`. Halaman ini dibuat khusus untuk melakukan pengujian waktu akses *query* dengan menggunakan fungsi `microtime` pada `php`.
- Tabel `data_soal` diberi masukan data sebanyak 1, 5, 10, 20, dan 50 data *entry* dalam *database*.
- Melakukan perhitungan untuk mendapatkan nilai waktu akses *query* yang dilakukan terhadap jumlah data yang berbeda dalam *database*.

c. Hasil Pengujian dan Analisis

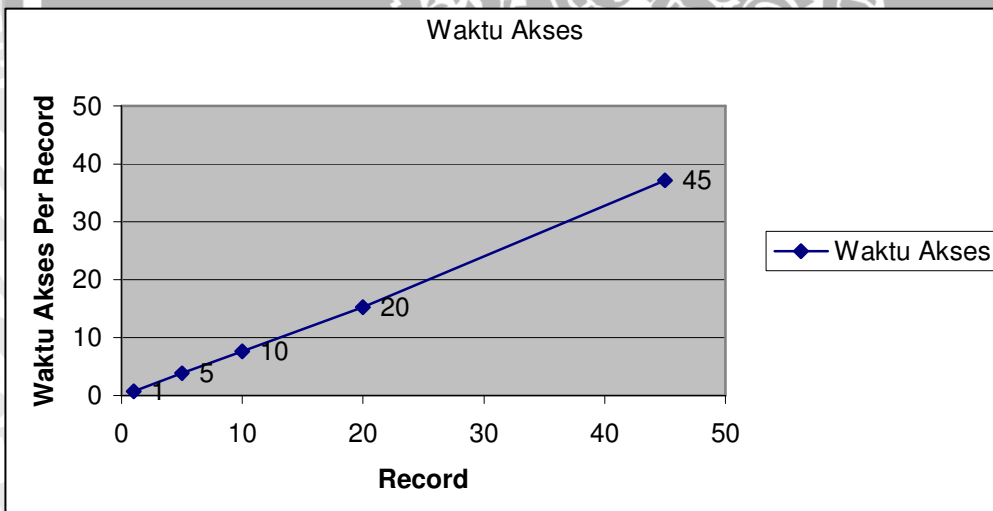
- Rata-rata pengujian waktu akses *query* lihat data soal dengan jumlah data Tabel `data_soal` sebanyak 11 data *entry* ditunjukkan dalam Tabel 6.1.

Tabel 6.1 Tabel Pengujian Waktu Akses *Query*

Pengujian	Jumlah <i>Record</i>	Waktu akses (detik)
1	1	0,74
2	5	3,85
3	10	7,68
4	20	15,58
5	45	37,16

Sumber : *Pengujian Dan Analisis*

- Grafik perbandingan waktu akses *query* pada lihat data soal dengan jumlah *block_id* berbeda diperlihatkan di dalam Gambar 6.37.



Gambar 6.37 Grafik Perbandingan Waktu Akses *Query*

Sumber : *Pengujian Dan Analisis*

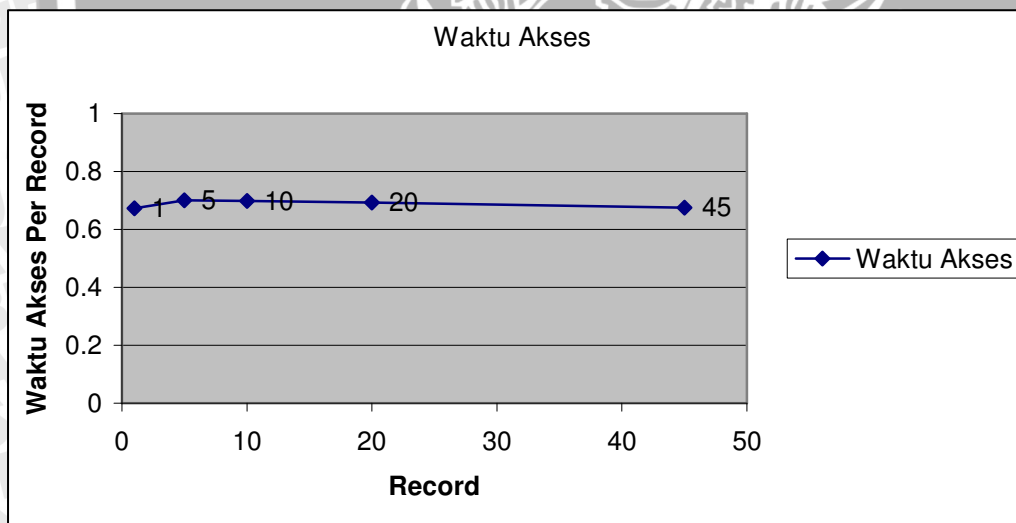
- Dari pengujian waktu akses *query* dapat diketahui waktu akses per *record* untuk *query* lihat data soal. Waktu akses per *record* merupakan hasil bagi antara waktu akses dengan total *record*. Perbandingan waktu akses per *record* dari *query* lihat data soal ditunjukkan dalam Tabel 6.2.

Tabel 6.2 Tabel Perbandingan Waktu Akses Per *Record* Dari *Query* Data soal

Pengujian	Total <i>Record</i>	Waktu akses (detik)	Waktu akses per <i>record</i> (detik)
1	1	0,74	0,744
2	5	3,85	0.771
3	10	7,68	0.778
4	20	15,58	0.762
5	45	37,16	0.825
Rata-rata waktu akses query per <i>record</i> (detik)			0.7794

Sumber : *Pengujian Dan Analisis*

- Grafik perbandingan waktu akses per *record* untuk *query* lihat data soal dengan jumlah data yang berbeda diperlihatkan di dalam Gambar 6.38

**Gambar 6.38** Grafik Waktu Akses Per *Record* Untuk *Query* Lihat Data soal

Sumber : *Pengujian Dan Analisis*

Berdasarkan hasil pengujian pada Tabel 6.2 waktu akses *query* untuk *query* lihat data soal dalam tabel yang berbeda yaitu 1, 5, 10, 20 dan 45 data *entry*, dapat diketahui bahwa rata-rata waktu akses per *record* sebesar 0.7794detik.

Berdasarkan hasil pengujian pada Gambar 6.38 dapat diketahui bahwa waktu akses rata-rata *query* lihat data soal semakin banyak data maka waktu akses relatif sama.



BAB VII PENUTUP

Kesimpulan

Dengan melihat hasil implementasi website pada Hotel Gajah Mada, maka dapat diambil beberapa kesimpulan, yaitu :

- Konfirmasi dan validasi reservasi dengan melalui via email lebih memberikan privasi pada pihak hotel maupun customer dalam melakukan proses reservasi.
- Customer dapat langsung melihat informasi mengenai fasilitas yang diberikan Hotel Gajah mada.
- Reservasi sekarang sudah dilakukan tanpa customer harus datang langsung ke hotel.
- Dengan adanya website ini Hotel Gajah mada dapat memberikan informasi fasilitas fasilitas yang diberikan serta dapat dijadikan sarana promosi.

Saran

Saran yang dapat diberikan oleh penulis untuk memperluas desain website www.gajahmada.com ini agar lebih sempurna antara lain :

- Perlu perbaikan pada menu Search engine.
- Penggunaan AJAX guna mempercepat proses loading halaman web.
- Penambahan fasilitas penghapusan data yang tidak valid secara berjangka.