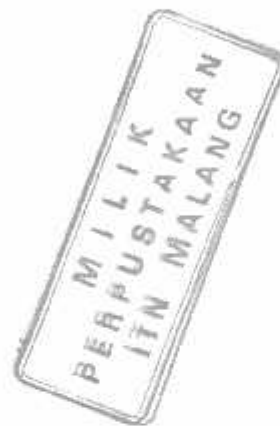


SKRIPSI

PERANCANGAN DAN PEMBUATAN SISTEM INFORMASI FRONT OFFICE PADA HOTEL VICTORY DI BATU-MALANG



Disusun Oleh :

ANDRIAS KRISTANTO WIDIGDO

05.12.526

**JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK KOMPUTER DAN INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2010**

LEMBAR PERSETUJUAN

PERANCANGAN DAN PEMBUATAN SISTEM INFORMASI FRONT OFFICE PADA HOTEL VICTORY DI BATU-MALANG

SKRIPSI

*Disusun dan Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh
Gelar Sarjana Teknik Komputer Dan Informatika Strata Satu (S-1)*

Disusun Oleh :

ANDRIAS KRISTANTO WIDIGDO
NIM : 05.12.526

Diperiksa dan Disetujui

Dosen Pembimbing I

Dosen Pembimbing II

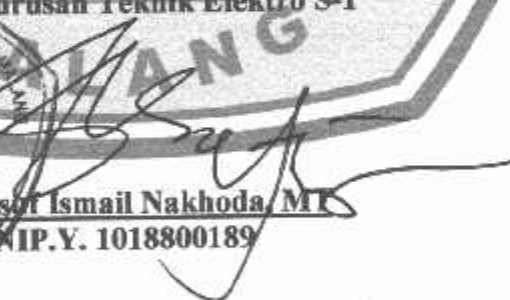

Ir. Yusuf Ismail Nakhoda, MT
NIP.Y. 1018800189


Febriana Santi W, S.Kom, M.Kom
NIP. 1018800189

Mengetahui

Ketua Jurusan Teknik Elektro S-1




Ir. Yusuf Ismail Nakhoda, MT
NIP.Y. 1018800189

JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK KOMPUTER DAN INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2010

**PERANCANGAN DAN PEMBUATAN SISTEM INFORMASI
ADMINISTRASI PERHOTELAN PADA HOTEL VICTORY
DI BATU-MALANG BERBASIS CLIENT-SERVER**

Andrias Kristanto Widigdo

**Jurusan Teknik Elektro S-1, Konsentrasi T.Komputer dan Informatika
Fakultas Teknologi Industri, Institut Teknologi Nasional Malang
Jln. Raya Karanglo Km 2 Malang
christanth_newbie@yahoo.com**

**Dosen Pembimbing : I. Ir. Yusuf Ismail Nakhoda, MT.
II. Febriana Santi W, S.Kom, M.Kom**

Abstraksi

Dalam era modern saat ini, begitu banyak sektor kehidupan yang tidak terlepas dari peran serta dan penggunaan teknologi informasi khususnya pada lingkup pekerjaan atau industri. Peran serta sistem informasi merupakan salah satu faktor keberhasilan dalam persaingan bisnis di berbagai bidang pekerjaan, yang mana dapat membantu proses komunikasi yang lebih efektif antar karyawan dalam sebuah bidang pekerjaan. Penerapan sistem informasi diharapkan dapat menyederhanakan sejumlah proses internal suatu lingkup pekerjaan atau industri.

Untuk mengatasi persaingan bisnis dalam usaha jasa penginapan yang semakin ketat saat ini maka salah satu langkah untuk meningkatkan kualitas dan mempertahankan keunggulan adalah dengan menyajikan sistem informasi secara cepat, tepat dan akurat yang bisa membantu pekerjaan lebih efektif dan efisien serta mampu memberikan pelayanan yang terbaik buat para customer.

Dengan penggunaan sistem informasi ini diharapkan dapat menunjang kinerja pada Hotel Victory dimana masih menggunakan sistem manual sehingga dengan adanya sistem informasi segala proses transaksi yang terjadi serta penyampaian informasi antar bagian pada Hotel Victory bisa berjalan lebih cepat dan efektif serta mudah digunakan oleh para karyawan yang bekerja di Hotel Victory.

Kata Kunci: Sistem Informasi, Hotel Victory

KATA PENGANTAR

Puji syukur kehadiran Tuhan Yang Maha Esa yang telah memberikan rahmat dan hidayah-Nya, sehingga dapat diselesaikan skripsi yang berjudul **“PERANCANGAN DAN PEMBUATAN SISTEM INFORMASI ADMINISTRASI PERHOTELAN PADA HOTEL VICTORY DI BATU-MALANG BERBASIS CLIENT-SERVER”** ini dengan lancar. Skripsi ini merupakan persyaratan kelulusan Studi pada Jurusan Teknik Elektro S-1 Konsentrasi Teknik Komputer dan Informatika ITN Malang dan untuk mencapai gelar Sarjana Teknik.

Keberhasilan penyelesaian laporan skripsi ini tidak lepas dari dukungan dan bantuan berbagai pihak. Untuk itu penyusun menyampaikan terima kasih kepada :

1. Prof. Dr. Ir. Abraham Lomi, MSEE, selaku Rektor Institut Teknologi Nasional Malang,
2. Bapak Ir. Sidik Noertjahjono, MT, selaku Dekan Fakultas Teknologi Industri,
3. Bapak Ir. F. Yudi Limpraptono, MT selaku Ketua Jurusan Teknik Elektro S-1,
4. Bapak Ir. Yusuf Ismail Nakhoda, MT selaku Sekretaris Jurusan Teknik Elektro S-1 serta Dosen Pembimbing I,
5. Ibu Febriana Santi W, S.Kom, M.Kom selaku Dosen Pembimbing II.

6. Almarhumah Ibu Ir. TH. Mimien Mustikawati, MT selaku Dosen Pembimbing I.
7. Ayah dan Ibu serta saudara-saudara kami yang telah memberikan do'a restu, dorongan, semangat, dan biaya.
8. Rekan-rekan instruktur di Laboratorium Pemrograman Komputer dan Multimedia ITN Malang.
9. Semua yang telah membantu dalam penyelesaian penyusunan skripsi ini.

Penulis telah berusaha semaksimal mungkin dan menyadari sepenuhnya akan keterbatasan pengetahuan dalam menyelesaikan laporan ini. Untuk itu penyusun mengharapkan saran dan kritik yang membangun dari pembaca demi kesempurnaan laporan ini.

Harapan penyusun semoga laporan skripsi ini memberikan manfaat bagi perkembangan ilmu pengetahuan dan pembaca.

Malang, Agustus 2010

Penyusun

DAFTAR ISI

LEMBAR PERSETUJUAN	i
ABSTRAKSI	ii
KATA PENGANTAR	iii
DAFTAR ISI	v
DAFTAR GAMBAR	ix
DAFTAR TABEL	xi
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Perumusan Masalah	2
1.3. Tujuan	3
1.4. Batasan Masalah	3
1.5. Metode Penelitian.....	3
1.5.1. Metode Pengumpulan Data	3
1.5.2. Metode Perancangan Sistem	4
1.6. Sistematika Penulisan.....	6
BAB II DASAR TEORI	7
2.1. Pengertian Dasar Sistem	7
2.1.1. Metode Pengembangan Sistem	9
2.2. Pegertian Dasar Sistem	10
2.3. Sistem Informasi	12

2.4. Hotel	15
2.4.1. Definisi Hotel	15
2.4.2. Sistem Informasi Administrasi Perhotelan	17
2.5. Pengertian Database	17
2.6. Perangkat Analisis dan Perancangan	19
2.6.1. Diagram Alir Data (DFD).....	19
2.6.2. Entity Relationship Diagram	23
2.7. Aplikasi (<i>Aplication</i>)	25
2.7.1. Borlanf Delphi 7	25
2.7.2. SQL Server	26
 BAB III ANALISA DAN PERANCANGAN SISTEM	 28
3.1. Sistem Pada Hotel Victory Batu Malang	28
3.1.1. Spesifikasi Sistem Saat Ini	29
3.1.2. Kekurangan Sistem Saat Ini	30
3.1.3. Pengguna Sistem	30
3.1.4. Spesifikasi Sistem	31
3.2. Perancangan Sistem	32
3.2.1. Data Flow Diagram	32
3.2.2. Alur Diagram DFD	34
3.2.3. Diagram Kontexs (<i>Context Diagram</i>)	34
3.2.4. Data Flow Diagram (DFD) Level 1	35
3.2.5. Data Flow Diagram (DFD) Level 2	36

3.2.5.1 DFD Level 2 Proses 1 (Proses Akses)	37
3.2.5.2 DFD Level 2 Proses 2 (Proses Keuangan)	37
3.2.6 Desain Basis Data	38
3.2.6.1. Relasi Antar Tabel	38
3.2.6.2. Struktur Tabel-tabel yang digunakan	40
3.2.7 Desain Antarmuka Aplikasi	46
BAB IV IMPLEMENTASI DAN PENGUJIAN SISTEM	47
4.1. Form Login	47
4.2. Form Utama	48
4.3. Main Menu	49
4.3.1. Form Registration	49
4.3.2. Tab Registrasi	50
4.3.3. Tab Reservasi	50
4.3.4. Tab Ceck Out	51
4.3.5. Form Restaurant	52
4.4. Menu Transaksi	52
4.4.1. Form Biaya Fasilitas	53
4.4.2. Form Biaya Telephone	53
4.4.3. Form Tagihan Tamu	54
4.5. Menu Pelaporan	54
4.5.1. Lap Data Tamu	55
4.5.2. Lap Reservasi	55

4.5.3. Lap Registrasi	56
4.5.4. Lap Restaurant	56
4.5.5. Lap Rugi Laba	57
4.6. Menu Akses	57
4.6.1. Form Entry Data Kamar	58
4.6.2. Form Entry Data Karyawan	58
BAB V PENUTUP	60
5.1. Kesimpulan	60
5.2. Saran	60
DAFTAR PUSTAKA	62
LAMPIRAN	63

DAFTAR GAMBAR

Gambar 2.1. Model Umum Sistem	7
Gambar 2.2. Siklus Informasi	11
Gambar 3.1. Desain Sistem	32
Gambar 3.2. Simbol Utama DFD	34
Gambar 3.3. Diagram Konteks	35
Gambar 3.4. DFD Level 1	36
Gambar 3.5. DFD Level 2 Proses 1 (Proses Akses)	37
Gambar 3.6. DFD Level 2 Proses 2 (Proses Keuangan)	37
Gambar 3.7. <i>Conceptual Data Model (CDM)</i>	39
Gambar 3.8. <i>Physical Data Model (PDM)</i>	40
Gambar 4.1. Form Login	47
Gambar 4.2. Form Menu Utama	48
Gambar 4.3. Form Registration	49
Gambar 4.4. Tab Registrasi	50
Gambar 4.5. Tab Reservasi	51
Gambar 4.6. Tab <i>Ceck Out</i>	51
Gambar 4.7. Form Restaurant	52
Gambar 4.8. Form Biaya Fasilitas	53
Gambar 4.9. Form Biaya Telephone	53
Gambar 4.10. Form Tagihan Tamu	54
Gambar 4.11. Form Lap Data Tamu	55

Gambar 4.12. Form Lap Reservasi	55
Gambar 4.13. Form Lap Registrasi	56
Gambar 4.14. Form Lap Restaurant	56
Gambar 4.15. Form Lap Rugi Laba	57
Gambar 4.16. Form Entry Data Kamar	58
Gambar 4.17. Form Entry Data Karyawan	59

DAFTAR TABEL

Tabel 2.1 Tabel Simbol DFD	20
Tabel 3.1 Tabel Reservasi	41
Tabel 3.2 Tabel Registrasi	41
Tabel 3.3 Tabel User	42
Tabel 3.4 Tabel Pembayaran atau Kasir	42
Tabel 3.5 Tabel Detil Bayar	43
Tabel 3.6 Tabel Harga Fasilitas	43
Tabel 3.7 Tabel Harga Kamar	43
Tabel 3.8 Tabel Data Karyawan	44
Tabel 3.9 Tabel Kode Harga Makan/Minum	44
Tabel 3.10 Tabel Data Restaurant	45
Tabel 3.11 Tabel Tipe Kamar	45
Tabel 3.12 Tabel Transaksi Fasilitas	46
Tabel 3.13 Tabel Transaksi Telephone	46

BAB I

PENDAHULUAN

1.1 Latar Belakang

Dalam era modern seperti saat ini, begitu banyak sektor kehidupan yang tidak terlepas dari peran serta dan penggunaan teknologi komputer dan internet, khususnya pada bidang-bidang dan lingkup pekerjaan atau industri. Semakin hari, kemajuan teknologi komputer, baik dibidang piranti lunak maupun perangkat keras berkembang dengan sangat pesat, disisi lain juga berkembang kearah yang sangat mudah dari segi pengaplikasian dan murah dalam hal biaya. Perkembangan komputer tersebut telah banyak dimanfaatkan di dalam sektor perdagangan, perindustrian dan jasa. Saat ini sudah banyak perusahaan-perusahaan yang sudah terkomputerisasi. Salah satunya adalah di dalam hal administrasi keuangan.

Pemanfaatan perkembangan teknologi komputer khususnya dalam bidang jasa penginapan dalam hal ini pada sebuah hotel belum sepenuhnya bisa dimanfaatkan secara maksimal, hal mengelola data pelayanan tamu dan transaksi lainnya ternyata belum digunakan secara maksimal pada semua hotel. Salah satunya adalah pada Hotel Victory Batu Malang saat ini. Pada hotel tersebut masih menggunakan sistem komputerisasi yang sangat sederhana yakni hanya pada bagian *accounting*-nya saja sehingga untuk transaksi lainnya masih menggunakan cara manual. Dalam pelaksanaannya masih banyak kendala yang dihadapi, antara lain adalah dibutuhkan waktu yang lama dalam memasukkan data-data mengenai transaksi yang dilakukan oleh hotel tersebut. Tentunya apabila kendala-kendala tersebut tidak segera diperbaiki maka Hotel Victory Batu Malang

tidak akan mampu mengikuti perkembangan dan kebutuhan teknologi di masa akan datang.

Untuk mengatasi masalah dan kendala ini perlu dikembangkan Sistem Informasi Front Office yang mampu menyajikan informasi secara cepat, tepat, dan akurat yang bisa membantu pekerjaan menjadi lebih efektif dan efisien, dan mampu memberikan pelayanan yang terbaik buat para *customer*.

Dengan penggunaan teknologi informasi diharapkan penyampaian informasi antar bagian dalam sebuah hotel bisa berjalan lebih cepat dan efektif. Selain itu juga akan didukung oleh sistem administrasi yang akan menangani transaksi yang terjadi sehari-hari dalam sebuah hotel. Dengan adanya Sistem Informasi Front Office ini diharapkan dapat digunakan sebagai sarana untuk mempermudah pengolahan informasi data *customer* beserta administrasinya yang ada di Hotel Victory Batu Malang.

1.2 Perumusan Masalah

Berdasarkan latar belakang di atas maka timbullah suatu permasalahan bagaimana merancang suatu aplikasi sistem informasi yang dapat meningkatkan layanan suatu Hotel terhadap *costumer*, di mana rumusan masalah tersebut meliputi bagaimana membuat sistem informasi yang sesuai pada sebuah hotel guna mempermudah penyampaian informasi antar bagian, bagaimana membuat sistem informasi administrasi yang dapat mempermudah proses transaksi yang dilakukan dalam sebuah hotel. Sehingga dapat dihasilkan data yang terstruktur dan informatif.

1.3 Tujuan

Tujuan dari skripsi ini adalah menciptakan suatu sistem informasi yang berguna untuk memudahkan penyampaian informasi antar bagian dalam sebuah hotel dan juga dilengkapi dengan sistem administrasi guna mempermudah proses transaksi yang dilakukan.

1.4 Batasan Masalah

Agar permasalahan mengarah sesuai dengan tujuan maka pembahasan dibatasi pada hal-hal sebagai berikut :

1. Proses pendataan daftar tamu.
2. Pengguna aplikasi ini hanya kalangan tertentu yang memiliki hak akses terhadap program.
3. Tidak membahas masalah jaringan yang digunakan.
4. Aplikasi Sistem Informasi Front Office ini hanya digunakan di Hotel Victory Batu Malang.
5. System aplikasi ini dibuat menggunakan program Delphi 7 dan sistem basis data menggunakan SQL Server 2000.

1.5 Metode Penelitian

1.5.1. Metode Pengumpulan Data

Data merupakan sumber atau bahan mentah yang sangat berharga bagi proses menghasilkan informasi. Oleh sebab itu dalam pengambilan data perlu dilakukan penanganan secara cermat dan hati-hati, sehingga data yang diperoleh dapat bermanfaat dan berkualitas.

Dalam pengumpulan data penyusun menggunakan metode sebagai berikut :

1. Studi Lapangan

Dengan metode ini data-data diperoleh langsung dari sumber yang bersangkutan, yang dilakukan dengan cara :

a. *Survey*

Teknik pengumpulan data dengan cara terjun secara langsung dan mencatat secara sistematis terhadap obyek masalah.

b. *Wawancara / Interview*

Teknik pengumpulan data dengan jalan mengadakan komunikasi atau tanya jawab secara langsung dengan Bapak Drs. Heru Kuswanto selaku Manager Hotel Victory Batu-Malang tentang sistem yang diterapkan.

2. Studi Pustaka / Literatur

Pengumpulan data ini dilakukan dengan cara mencari bahan-bahan kepustakaan sebagai landasan teori yang ada hubungannya dengan permasalahan yang dijadikan obyek penelitian.

1.5.2. Metode Perancangan Sistem

Metode yang digunakan dalam perancangan Sistem Informasi Administrasi ini adalah :

- *Rekayasa Sistem*

Tahap ini ditekankan pada pengumpulan kebutuhan pengguna tingkatan sistem dengan mendefinisikan konsep sistem beserta

interface yang menghubungkannya dengan lingkungan. Hasil dari tahap ini adalah spesifikasi sistem.

- Analisis

Tahap ini melakukan pengumpulan kebutuhan elemen-elemen ditingkat perangkat lunak. Dengan analisa harus dapat ditentukan kebutuhan data, fungsi, proses atau prosedur yang diperlukan beserta unjuk kerjanya, dan *interfaces*. Hasil akhir tahap ini adalah spesifikasi kebutuhan perangkat lunak.

- Design

Spesifikasi perangkat lunak yang dihasilkan dari tahap analisa ditransformasikan kedalam bentuk arsitektur perangkat lunak yang memiliki karakteristik mudah dimengerti dan tidak sulit untuk diimplementasikan. Tahap ini terdiri dari dua tahap, yaitu *preliminary design* dan *detailed design*. Subtahap pertama menghasilkan rancangan yang bersifat global, sedangkan subtahap yang kedua menghasilkan rancangan detil sehingga semua kelas, tipe data, fungsi dan prosedurnya terdefinisi.

- Implementasi Sistem

Setelah mendapatkan gambaran yang jelas tentang rancang bangun sistem, kemudian dilakukan implementasi rancangan sistem ke dalam kode-kode dalam bahasa pemrograman. Pada tahap ini dilakukan pembuatan komponen-komponen sistem yang meliputi implementasi modul-modul program, antarmuka dan basis data.

- Pengujian Sistem

Tujuan dilakukan tahapan pengujian ini adalah untuk mendapatkan perangkat lunak yang benar-benar valid dan sesuai dengan kebutuhan yang sudah dideskripsikan.

1.6 Sistematika Penulisan

Sistematika penulisan yang digunakan dalam penyusunan skripsi ini adalah sebagai berikut:

BAB I : PENDAHULUAN

Bab ini berisi tentang latar belakang, tujuan, permasalahan, batasan masalah, dan sistematika pembahasan dari skripsi ini.

BAB II : DASAR TEORI

Bab ini berisi tentang landasan teori yang dapat dipakai sebagai penunjang dalam pembuatan aplikasi.

BAB III : PERANCANGAN DAN PEMBUATAN SISTEM

Bab ini berisi tentang analisis dan desain dengan membuat *data flow diagram (DFD)* dan *conceptual data model (CDM)* pada aplikasi yang dibuat.

BAB IV : HASIL DAN ANALISA

Bab ini berisi pembahasan hasil pengujian dan analisa mengenai cara kerja dari sistem.

BAB V : PENUTUP

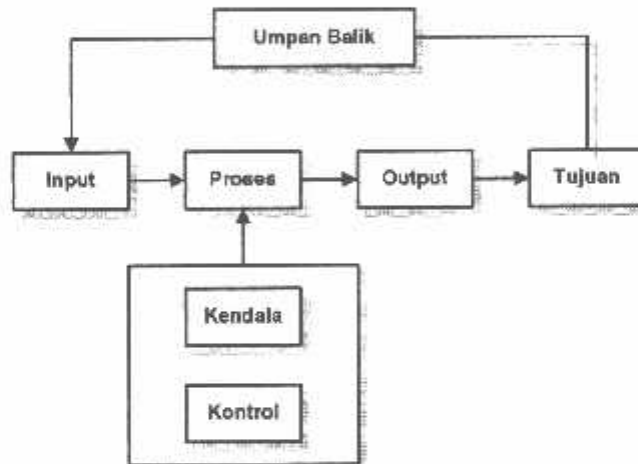
Bab ini berisi kesimpulan dan saran yang didapatkan dari pembuatan aplikasi.

BAB II

DASAR TEORI

2.1. Pengertian Dasar Sistem

Sistem adalah suatu kesatuan utuh yang terdiri dari beberapa bagian yang saling berhubungan dan berinteraksi untuk mencapai tujuan tertentu.[2] Di bawah ini merupakan gambar model umum dari sebuah sistem, di mana terdapat lima buah komponen utama dalam sistem yang membuat sebuah sistem dapat bekerja dengan baik.



Gambar 2.1 Model Umum Sistem ^[2]

- *Komponen Input*

Komponen *Input* merupakan bagian dari sistem yang bertugas untuk menerima data masukan. Data masukan ini digunakan sebagai komponen penggerak atau pemberi tenaga dimana sistem itu dioperasikan. Komponen penggerak ini terbagi menjadi dua, yaitu *Maintenance Input* merupakan energi yang dimasukkan supaya sistem dapat beroperasi dan *Signal Input* energi yang diproses untuk didapatkan keluaran.

- **Komponen Proses**

Komponen Proses merupakan komponen dalam sistem yang melakukan pengolahan *input* untuk mendapatkan hasil atau tujuan yang diinginkan. Di dalam suatu proses, terjadi berbagai kegiatan seperti klasifikasi, peringkasan, pencarian data, organisasi data dan sebagainya. Begitu kompleksnya sebuah proses, maka pada tahap ini diperlukan terjadinya suatu integrasi yang baik antar subsistem secara vertikal maupun secara horisontal agar proses interaksi untuk mencapai tujuan dapat berjalan lancar.

- **Komponen *Output***

Komponen *Output* merupakan komponen hasil pengoperasian dari suatu sistem.

- **Komponen Tujuan**

Terdapatnya suatu tujuan yang jelas akan memberikan arah yang jelas pula pada proses sistem. Komponen tujuan merupakan sasaran yang ingin dicapai oleh berjalannya sebuah sistem. Tujuan ini bisa berupa tujuan usaha, kebutuhan sistem, pemecahan suatu masalah, dan sebagainya.

- **Komponen Kendala**

Komponen Kendala merupakan komponen yang berisikan aturan atau batas-batas yang berlaku atas tujuan tersebut. Pendefinisian kendala yang jelas, akan membuat tujuan menjadi lebih bermanfaat. Dengan adanya kendala atau batas-batas yang jelas, maka akan mampu mengidentifikasi apa yang harus diantisipasi dalam mencapai tujuan sistem.

- **Komponen Kontrol**

Komponen Kontrol merupakan komponen pengawas dari pelaksanaan proses pencapaian tujuan. Kontrol disini dapat berupa kontrol pemasukan *input*, kontrol pengeluaran data, kontrol pengoperasian dan lain-lain.

- **Komponen Umpan Balik**

Komponen Umpan Balik merupakan komponen yang memberikan respon atas berjalannya suatu sistem. Komponen ini dapat berupa kegiatan seperti perbaikan atau pemeliharaan sistem.

2.1.1. Metode Pengembangan Sistem

Metode yang digunakan dalam pengembangan Sistem Informasi Administrasi ini adalah :

- **Rekayasa Sistem**

Tahap ini ditekankan pada pengumpulan kebutuhan pengguna tingkatan sistem dengan mendefinisikan konsep sistem beserta *interface* yang menghubungkannya dengan lingkungan. Hasil dari tahap ini adalah spesifikasi sistem.

- **Analisis**

Tahap ini melakukan pengumpulan kebutuhan elemen-elemen ditingkat perangkat lunak. Dengan analisa harus dapat ditentukan kebutuhan data, fungsi, proses atau prosedur yang diperlukan beserta unjuk kerjanya, dan *interfaces*. Hasil akhir tahap ini adalah spesifikasi kebutuhan perangkat lunak.

- *Design*

Spesifikasi perangkat lunak yang dihasilkan dari tahap analisa ditransformasikan kedalam bentuk arsitektur perangkat lunak yang memiliki karakteristik mudah dimengerti dan tidak sulit untuk diimplementasikan. Tahap ini terdiri dari dua tahap, yaitu *preliminary design* dan *detailed design*. Subtahap pertama menghasilkan rancangan yang bersifat global, sedangkan subtahap yang kedua menghasilkan rancangan detil sehingga semua kelas, tipe data, fungsi dan prosedurnya terdefinisi.

- Implementasi Sistem

Setelah mendapatkan gambaran yang jelas tentang rancang bangun sistem, kemudian dilakukan implementasi rancangan sistem ke dalam kode-kode dalam bahasa pemrograman. Pada tahap ini dilakukan pembuatan komponen-komponen sistem yang meliputi implementasi modul-modul program, antarmuka dan basis data.

- Pengujian Sistem

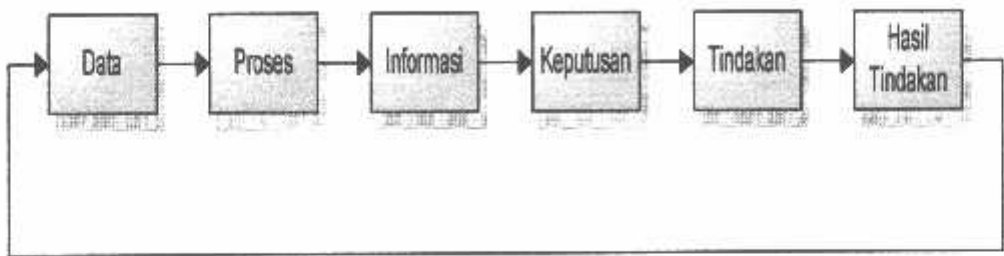
Tujuan dilakukan tahapan pengujian ini adalah untuk mendapatkan perangkat lunak yang benar-benar valid dan sesuai dengan kebutuhan yang sudah dideskripsikan.

2.2. Pengertian Dasar Informasi

Informasi merupakan hasil dari pengolahan data menjadi bentuk yang lebih berguna bagi yang menerimanya yang menggambarkan suatu kejadian-

kejadian nyata dan dapat digunakan sebagai alat bantu untuk pengambilan suatu keputusan.

Pengolahan data menjadi suatu informasi dapat digambarkan sebagai sebuah siklus yang berkesinambungan seperti berikut :



Gambar 2.2 Siklus Informasi ^[2]

Secara sederhana dapat dikatakan bahwa data diolah menjadi suatu informasi. Dan pada tahapan selanjutnya, sebuah informasi akan menjadi data untuk terciptanya informasi yang lain. Pada gambar 2.2 dapat dilihat bahwa pada awalnya data dimasukan kedalam model yang umumnya memiliki urutan proses tertentu dan pasti, setelah diproses akan dihasilkan informasi tertentu yang bermanfaat bagi penerima (*level management*) sebagai dasar dalam membuat suatu keputusan atau melakukan tindakan tertentu. Dari keputusan atau tindakan tersebut akan menghasilkan atau diperoleh kejadian-kejadian tertentu yang akan digunakan kembali sebagai data yang nantinya akan dimasukan kedalam model (*proses*), begitu seterusnya sehingga tercipta sebuah siklus yang berkesinambungan.

Kualitas Informasi (*quality of information*) sangat dipengaruhi atau ditentukan oleh 3 hal pokok, yaitu sebagai berikut :

- Relevansi (*Relevancy*)

Informasi akan relevan jika memberikan manfaat bagi pemakainya. Relevansi informasi untuk tiap-tiap orang satu dengan yang lainnya berbeda. Misalnya informasi mengenai hasil penjualan barang mingguan kurang relevan jika ditujukan pada manajer teknik, tetapi akan sangat relevan bila disampaikan pada manajer pemasaran.

- Akurasi (*Accuracy*)

Sebuah informasi dapat dikatakan akurat jika informasi tersebut bebas dari kesalahan-kesalahan dan harus jelas mencerminkan maksudnya. Ketidakakuratan sebuah informasi dapat terjadi karena sumber informasi (data) mengalami gangguan atau kesengajaan sehingga merusak atau merubah data-data asli tersebut.

- Tepat Waktu (*timeliness*)

Informasi yang dihasilkan dari suatu proses pengolahan data, datangnya tidak boleh terlambat. Informasi yang terlambat tidak akan mempunyai nilai yang baik, sehingga kalau digunakan sebagai dasar dalam pengambilan keputusan dapat menimbulkan kesalahan dalam tindakan yang akan diambil.

2.3. Sistem Informasi

Sistem Informasi didefinisikan oleh *Robert A. Leitch / Roscoe Davis* yang dikutip oleh jogianto (2001) sebagai berikut :

“Sistem Informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi,

bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diberikan”.

Sistem Informasi dikembangkan untuk tujuan yang berbeda-beda, tergantung pada kebutuhan bisnis. Jenis-jenis Sistem Informasi dapat dibagikan menjadi beberapa bagian yaitu :

a. *Transaction Processing System (TPS).*

Transaction Processing System (TPS) Adalah Sistem Informasi yang terkomputerisasi yang dikembangkan untuk memproses data dalam jumlah besar untuk transaksi bisnis rutin seperti daftar gaji dan *inventaris*. TPS berfungsi pada level organisasi yang memungkinkan organisasi bisa berinteraksi dengan lingkungan eksternal. Data yang dihasilkan oleh TPS dapat dilihat atau digunakan oleh Manajer.

b. *Office Automation system (OAS) dan Knowledge Work System (KWS).*

Office Automation System (OAS) dan *Knowledge Work System (KWS)* bekerja pada level *knowledge*. OAS mendukung pekerja data, yang biasanya tidak menciptakan pengetahuan baru melainkan hanya menganalisis informasi sedemikian rupa untuk mentransformasikan data atau memanipulasikannya dengan cara tertentu sebelum menyebarkan secara keseluruhan dengan organisasi dan kadang-kadang diluar organisasi. Aspek-aspek OAS seperti *word processing, spreadsheets, electronic scheduling*, dan komunikasi melalui *voice mail, email* dan *vidio conferencing*.

KWS mendukung para pekerja profesional seperti ilmuwan, Insinyur dan Doktor dengan membentuk menciptakan pengetahuan baru dan memungkinkan mereka mengkontribusikannya ke organisasi atau masyarakat.

c. Sistem Informasi Manajemen (SIM)

Sistem Informasi Manajemen (SIM) tidak menggantikan TPS, tetapi mendukung spektrum tugas-tugas organisasional yang lebih luas dari TPS, termasuk analisis keputusan dan pembuatan keputusan. SIM menghasilkan informasi yang digunakan untuk membuat keputusan, dan juga dapat membantu menyatukan beberapa fungsi informasi bisnis yang sudah terkomputerisasi (basis data).

d. *Decision Support System* (DSS)

Decision Support System (DSS) hampir sama dengan SIM karena menggunakan basis data sebagai sumber data. DSS bermula dari SIM karena menekankan pada fungsi mendukung pembuat keputusan diseluruh tahap-tahapnya, meskipun keputusan aktual tetap wewenang *eksklusif* pembuat keputusan.

e. Sistem Ahli (ES) dan Kecerdasan Buatan (AI)

Kecerdasan Buatan (AI) dimaksudkan untuk mengembangkan mesin-mesin yang berfungsi secara cerdas. Dua cara untuk melakukan riset AI adalah memahami bahasa alamiah dan menganalisis kemampuannya untuk berfikir melalui problema sampai kesimpulan logikanya. Sistem ahli menggunakan pendekatan-pendekatan pemikiran AI untuk menyelesaikan masalah serta memberikanya lewat pengguna bisnis. Sistem ahli juga disebut *knowledge-based system* secara efektif menangkap dan menggunakan pengetahuan seorang ahli untuk menyelesaikan masalah yang dialami dalam suatu organisasi. Berbeda dengan DSS, DSS meninggalkan keputusan terakhir bagi pembuat keputusan sedangkan sistem ahli menyeleksi solusi terbaik terhadap

suatu masalah khusus. Komponen dasar sistem ahli adalah *knowledge-base* yakni suatu mesin inferensi yang menghubungkan pengguna dengan sistem melalui pengolahan pertanyaan lewat bahasa terstruktur dan antarmuka pengguna.

f. *Group Decision Support System (GDSS)* dan *Computer-Support Collaborative Work System (CSCW)*.

Bila kelompok, perlu bekerja bersama-sama untuk membuat keputusan semi terstruktur, maka *group Decision Support System* membuat suatu solusi. *Group Decision Support System (GDSS)* dimaksudkan untuk membawa kelompok bersama-sama menyelesaikan masalah dengan memberi bantuan dalam bentuk pendapat, kuisioner, konsultasi dan skenario. Kadang-kadang GDSS disebut juga dengan *Computer-Support Collaborative Work System (CSCW)* yang mencakup pendukung perangkat lunak yang disebut dengan “*groupware*” untuk kolaborasi tim melalui komputer yang terhubung dengan jaringan.

g. *Executive Support System (ESS)*.

Executive Support System (ESS) tergantung pada informasi yang dihasilkan TPS dan SIM serta ESS membantu eksekutif mengatur interaksinya dengan lingkungan eksternal dengan menyediakan grafik-grafik dan pendukung komunikasi di tempat-tempat yang bisa diakses seperti kantor.

2.4. Hotel

2.4.1. Definisi Hotel

Hotel adalah suatu bentuk bangunan, lambang, perusahaan atau badan usaha akomodasi yang menyediakan pelayanan jasa penginapan, penyedia makanan dan minuman serta fasilitas jasa lainnya dimana semua pelayanan itu diperuntukkan bagi masyarakat umum, baik mereka yang bermalam di hotel tersebut ataupun mereka yang hanya menggunakan fasilitas tertentu yang dimiliki hotel itu. Pengertian hotel ini dapat disimpulkan dari beberapa definisi hotel seperti tersebut di bawah ini :

- a. Salah satu jenis akomodasi yang mempergunakan sebagian atau keseluruhan bagian untuk jasa pelayanan penginapan, penyedia makanan dan minuman serta jasa lainnya bagi masyarakat umum yang dikelola secara komersil (Keputusan Menteri Parpostel no Km 94/HK103/MPPT 1987)
- b. Bangunan yang dikelola secara komersil dengan memberikan fasilitas penginapan untuk masyarakat umum dengan fasilitas sebagai berikut :
 - 1) Jasa penginapan
 - 2) Pelayanan makanan dan minuman
 - 3) Pelayanan barang bawaan
 - 4) Pencucian pakaian
 - 5) Penggunaan fasilitas perabot dan hiasan-hiasan yang ada di dalamnya.(Endar Sri,1996:8)
- c. Sarana tempat tinggal umum untuk wisatawan dengan memberikan pelayanan jasa kamar, penyedia makanan dan

minuman serta akomodasi dengan syarat pembayaran (Lawson, 1976:27)

2.4.2. Sistem Informasi Front Office

Sistem Informasi Front Office pada Hotel Victory Batu Malang memiliki fungsi untuk mengelola data pelayanan tamu dan transaksi yang dilakukan oleh tamu serta penyampaian informasi antar bagian dalam hotel tersebut. Data yang diolah adalah data tamu yang datang menginap serta transaksi layanan yang dilakukan tamu selama menginap, kemudian dibuatkan rekening tamu yang berisi total pembayaran tamu selama menginap, merupakan besarnya pemasukan yang didapat oleh hotel tersebut. Data lain yang diolah adalah data tentang stock barang yang tersedia untuk kegiatan operasional hotel, sehingga bisa diketahui informasi tentang ketersediaan barang untuk operasional hotel.

2.5. Pengertian *Database*

Basis data (*Database*) adalah Sekumpulan program-program aplikasi umum yang bersifat sekumpulan atau kelompok yang mengeksekusi dan memproses data secara umum (hapus, cari, *update*, dll). Basis data dimaksudkan untuk mengatasi problem pada sistem yang memakai pendekatan berbasis berkas. Sebuah konsep *Database* memiliki beberapa hal sebagai berikut :

- *Field*
 - a. *Field* merupakan implementasi dari suatu atribut data.

- b. *Field* juga merupakan unit terkecil dari data yang berarti (*meaningful data*) yang disimpan dalam suatu *file* atau basis data.
- *Record*
 - a. *Field-field* tersebut diorganisasikan dalam *record-record*
 - b. *Record* merupakan koleksi dari *field-field* yang disusun dalam format yang telah ditentukan.
 - c. Selama desain sistem, *record* akan diklasifikasikan sebagai *fixed-length record* atau *variable-length record*.
 - *Fixed-length record*: tiap instance *record* punya *field*, jumlah *field*, dan ukuran logik yang sama
 - *Variable-length record* : mengijinkan *record-record* yang berbeda dalam file yang sama memiliki panjang yang berbeda.
- *File dan Tabel*
 - a. *Record-record* yang serupa diorganisasikan dalam grup-grup yang disebut *file*. Jadi *file* merupakan kumpulan semua kejadian dari struktur *record* yang diberikan.
 - b. Tabel merupakan ekivalen basis data relasional dari sebuah file.

Untuk mengelola basis data diperlukan perangkat lunak yang disebut *Database Management System* (DBMS). *DataBase Management System* (DBMS) adalah sistem yang secara khusus dibuat untuk memudahkan pemakai dalam mengelola basis data. Sistem ini dibuat untuk mengatasi kelemahan sistem pemrosesan yang berbasis berkas. Pada pendekatan yang berbasis berkas,

umumnya perancangan sistem didasarkan pada kebutuhan individual pemakai, bukan berdasarkan kebutuhan sejumlah pemakai.

2.6. Perangkat Analisis dan Perancangan

2.6.1. Diagram Aliran Data (DFD)

DFD adalah suatu model logika atau proses yang dibuat untuk menggambarkan dari mana asal data dan yang keluar dari sistem. Dimana data disimpan, proses apa yang menghasilkan data tersebut dan interaksi antara data yang tersimpan dan proses yang dikenakan pada data tersebut.

DFD sering digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan di kembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut mengalir atau dimana data tersebut akan disimpan.

DFD merupakan alat yang digunakan pada *metodologi* pengembangan sistem yang terstruktur. Kelebihan utama pendekatan aliran data, yaitu :

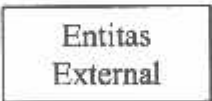
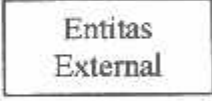
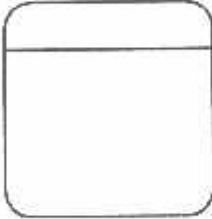
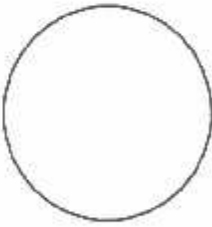
- a. Kebebasan dari menjalankan implementasi teknis sistem.
- b. Pemahaman lebih jauh mengenai keterkaitan satu sama lain dalam sistem dan subsistem.
- c. Mengkomunikasikan pengetahuan sistem yang ada dengan pengguna melalui diagram alir data.
- d. Menganalisis sistem yang diajukan untuk menentukan apakah data-data dan proses yang diperlukan telah ditetapkan.

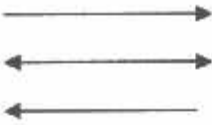
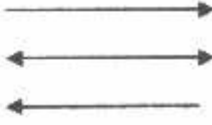
DFD terdiri dari *context diagram* dan diagram rinci (DFD *Levelled*). *Context* diagram berfungsi memetakan model lingkungan (menggambarkan

hubungan antara entitas luar, masukan dan keluaran sistem), yang direpresentasikan dengan lingkaran tunggal yang mewakili keseluruhan sistem. DFD *Levelled* menggambarkan sistem sebagai jaringan kerja antara fungsi yang berhubungan satu sama lain dengan aliran dan penyimpanan data, model ini hanya memodelkan sistem dari sudut pandang fungsi.

Dalam DFD *Levelled* akan terjadi penurunan *level* dimana dalam penurunan *level* yang lebih rendah harus mampu mempresentasikan proses tersebut ke dalam spesifikasi proses yang jelas. Jadi dalam DFD *Levelled* bisa dimulai dari DFD level 0 kemudian turun ke DFD level 1 dan seterusnya. Berikut merupakan tabel simbol DFD :

Tabel 2.1
Tabel Simbol DFD ⁽¹⁾

Gane/Sarson	Yourdon/De Marco	Keterangan
		Entitas eksternal, dapat berupa orang/ unit terkait yang berinteraksi dengan sistem tetapi diluar sistem
		Orang,unit yang mempergunakan atau melakukan transformasi data komponen fisik tidak diidentifikasi.

		Aliran data dengan arah khusus dari sumber ke tujuan
		Penyimpanan data atau tempat data direfer oleh proses.

Dalam menggambar DFD, ada beberapa peraturan yang harus diperhatikan sehingga dalam penggambarannya tidak terjadi kesalahan, aturan tersebut yaitu :

- Antara entitas tidak diijinkan terjadi hubungan atau relasi.
- Tidak boleh ada aliran data antara entitas eksternal dengan data *store*.
- Untuk alasan kerapian (menghindari aliran data yang bersilang), entitas eksternal atau data *store* boleh digambar beberapa kali dengan tanda khusus, misalnya diberi nomor.
- Satu aliran data boleh mengalirkan beberapa paket data.
- Bentuk anak panah aliran data boleh bervariasi.
- Semua objek harus mempunyai nama.
- Aliran data selalu mewakili atau diakhiri dengan proses.
- Semua aliran data harus mempunyai tanda arah.
- Jumlah proses tidak lebih dari sembilan proses dalam sistem, jika melebihi maka sebaiknya dikelompokkan beberapa proses yang bekerja bersama-sama di dalam suatu subsistem.

Ada beberapa petunjuk yang dapat digunakan dalam pembuatan DFD, yaitu sebagai berikut :

a. Penamaan yang jelas.

- Setiap entitas diberi nama yang sesuai dengan suatu kata benda.
- Nama aliran data dalam kata benda karena menunjukkan seseorang, tempat atau sesuatu.
- Proses pemberian nama menggunakan format kata kerja, kata sifat, kata bentuk untuk proses-proses yang rinci.
- Penyimpanan data diberi nama dengan suatu kata benda.

b. Memberi nomor pada proses.

- Nomor yang di berikan pada proses tidak harus menjadi nomor urut.
- Penomoran dimaksudkan sebagai identifikasi proses dan memudahkan penurunan (*level* yang rendah) ke proses berikutnya.
- Untuk proses primitif selain diberi nomor juga diberi tanda khusus (biasanya tanda*) untuk menyatakan bahwa proses tersebut tidak dirinci kembali.

c. Penggambaran kembali.

- Ukuran dan bentuk lingkaran tetap sama.
- Panah yang melengkung dan lurus tidak jadi masalah.

d. Hindari proses yang mempunyai masukan tetapi tidak mempunyai keluaran begitu juga sebaliknya hindari proses yang mempunyai keluaran tetapi tidak mempunyai masukan.

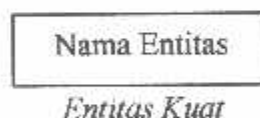
2.6.2. Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) adalah pemodelan data utama yang membantu mengorganisasikan data dalam suatu proyek ke dalam entitas-entitas dan menentukan hubungan antar entitas.[5] Notasi yang digunakan dalam ERD antara lain :

a. Entity.

Entitas adalah sesuatu atau objek di dunia nyata yang dapat dibedakan dari sesuatu objek yang lainnya. Sebagai contoh, setiap mahasiswa dalam suatu universitas adalah suatu entitas. Setiap fakultas dalam suatu universitas adalah juga entitas. Dapat dikatakan bahwa entitas bisa bersifat konseptual/abstrak atau nyata hadir di dunia nyata. Simbol dari entity dibagi menjadi 3 yaitu :

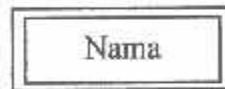
- a. Entitas kuat adalah entitas yang mandiri, yang keberadaannya tidak tergantung pada keberadaan entitas yang lainnya. Instansiasi entitas kuat selalu memiliki karakteristik yang untuk disebut identifier (sebuah atribut tunggal atau gabungan atribut-atribut yang secara unik dapat digunakan untuk membedakannya dari entitas kuat yang lain).



- b. *Entitas Asosiatif* adalah entitas yang keberadaannya berada antara entitas kuat dan entitas lemah.



- c. Entitas lemah atau *weak entity* adalah entitas yang keberadaanya sangat bergantung pada keberadaan entitas yang lainnya. Entitas lemah tidak memiliki arti apa-apa dan tidak dikehendaki kehadirannya dalam diagram ER tanpa kehadiran entitas di mana mereka bergantung.



Entitas Lemah Atau Week Entity

b. Relationship.

Hubungan antara relasi adalah hubungan antara suatu himpunan entitas dengan himpunan entitas yang lainnya. Misalnya, entitas mahasiswa memiliki hubungan tertentu dengan entitas matakuliah (Mahasiswa mengambil mata kuliah). Pada penggambaran model E-R, relasi adalah perekat yang menghubungkan suatu entitas dengan entitas yang lainnya.



Relasi Relationship

c. Atribut.

Atribut adalah properti *deskriptif* yang dimiliki oleh setiap anggota dari himpunan entitas. Sebagai contoh entitas mahasiswa, atribut-atribut yang dimiliki adalah nim, nama mahasiswa, alamat dan lain-lain.



d. Garis.

Garis sebagai penghubung antara relasi dengan entitas, relasi dan entitas dengan atribut.



2.7. Aplikasi (*Application*)

2.7.1. Borland Delphi 7.0

Delphi adalah suatu bahasa pemrograman (*development language*) yang biasa digunakan untuk merancang suatu aplikasi program. Kegunaan umum dari Delphi 7 ini adalah untuk membuat aplikasi *windows*, untuk merancang aplikasi program berbasis grafis, untuk membuat program berbasis jaringan (*client/server*), untuk merancang program .Net (berbasis internet).

Berikut Keunggulan Delphi :

1. *IDE (Integrated Development Environment)* atau lingkungan pengembangan aplikasi sendiri adalah satu dari beberapa keunggulan delphi,

didalamnya terdapat menu – menu yang memudahkan kita untuk membuat suatu proyek program.

2. Proses Kompilasi cepat, pada saat aplikasi yang kita buat dijalankan pada Delphi, maka secara otomatis akan dibaca sebagai sebuah program, tanpa dijalankan terpisah.
3. Mudah digunakan, *source code* delphi yang merupakan turunan dari pascal, sehingga tidak diperlukan suatu penyesuain lagi.
4. Bersifat *multi purphase*, artinya bahasa pemograman Delphi dapat digunakan untuk mengembangkan berbagai keperluan pengembangan aplikasi.

2.7.2. SQL Server

SQL (*Structured Query Language*) adalah sebuah bahasa yang dipergunakan untuk mengakses data dalam basis data relasional. Bahasa ini secara *de facto* merupakan bahasa standar yang digunakan dalam manajemen basis data relasional. Saat ini hampir semua server basis data yang ada mendukung bahasa ini untuk melakukan manajemen datanya.

Secara umum, SQL terdiri dari dua bahasa, yaitu *Data Definition Language* (DDL) dan *Data Manipulation Language* (DML). Implementasi DDL dan DML berbeda untuk tiap sistem manajemen basis data (SMBD), namun secara umum implementasi tiap bahasa ini memiliki bentuk standar yang ditetapkan ANSI. Artikel ini akan menggunakan bentuk paling umum yang dapat digunakan pada kebanyakan SMBD.

- ***Data Definition Language (DDL)***

Data Definition Language (DDL) digunakan untuk mendefinisikan, mengubah, serta menghapus basis data dan objek-objek yang diperlukan dalam basis data, misalnya tabel, *view*, *user*, dan sebagainya. Secara umum, DDL yang digunakan adalah CREATE untuk membuat objek baru, USE untuk menggunakan objek, ALTER untuk mengubah objek yang sudah ada, dan DROP untuk menghapus objek. DDL biasanya digunakan oleh *administrator* basis data dalam pembuatan sebuah aplikasi basis data.

- ***Data Manipulation Language (DML)***

Data Manipulation Language (DML) digunakan untuk memanipulasi data yang ada dalam suatu tabel. Perintah yang umum dilakukan adalah:

- *SELECT* untuk menampilkan data
- *INSERT* untuk menambahkan data baru
- *UPDATE* untuk mengubah data yang sudah ada
- *DELETE* untuk menghapus data

BAB III

ANALISIS DAN PERANCANGAN SISTEM

Pada bab ini dijelaskan mengenai analisis dan perancangan sistem aplikasi. Analisis ditujukan untuk memberikan gambaran secara umum terhadap aplikasi. Hal ini berguna untuk menunjang perancangan aplikasi yang akan dikembangkan sehingga kebutuhan akan aplikasi tersebut dapat diketahui sebelumnya. Kemudian hasil analisis akan menjadi dasar untuk melakukan perancangan atau desain aplikasi sesuai kebutuhan sistem.

Dalam merancang aplikasi pada skripsi ini terlebih dahulu dilakukan pembuatan desain proses, desain data, serta desain antar muka aplikasi. Desain proses berguna untuk mengintegrasikan semua proses yang terjadi dalam aplikasi yang akan dibuat. Desain data berguna untuk mengetahui data apa saja yang dibutuhkan dalam proses yang akan dikerjakan. Sedangkan perancangan antarmuka berfungsi sebagai antar muka interaksi antara pengguna dengan sistem aplikasi yang dibuat, sehingga pengguna dapat mengoperasikan aplikasi yang dibuat.

3.1 Sistem Pada Hotel Victory Batu Malang

Sistem yang ada saat ini sudah berbasis komputer, tetapi masih belum menerapkan sistem basis data yang terstruktur. Aplikasi yang digunakan terbatas hanya pada aplikasi *Microsoft Office Excel* yang disediakan oleh Hotel Victory Batu. Dalam pelaksanaannya masih banyak kendala yang dihadapi, antara lain adalah dibutuhkan waktu yang lama dalam memasukkan data-data mengenai transaksi yang dilakukan oleh hotel tersebut. Tentunya apabila kendala-kendala

tersebut tidak segera diperbaiki maka dikhawatirkan Hotel Victory Batu Malang tidak mampu untuk melakukan pengolahan data lebih lanjut.

Data yang dihasilkan akan dicetak dalam bentuk laporan atau disimpan ke dalam media penyimpanan seperti *Disket*, *CD*, dan *Flashdisk*, kemudian laporan tersebut akan dicetak secara berkala untuk mengetahui laporan masing-masing divisi. Kemudian ditempat yang lain laporan-laporan tersebut diperiksa oleh pimpinan untuk mengetahui mana yang harus dibetulkan atau dikoreksi. Sehingga akan membutuhkan waktu yang lama dan kurang efisien.

3.1.1. Spesifikasi Sistem Saat Ini

Spesifikasi sistem yang berjalan saat ini antara lain :

- Sistem yang digunakan adalah aplikasi *Microsoft Office Excel* untuk mengelola data dan informasi di Hotel Victory Batu Malang.
- Sistem ini hanya menggunakan satu buah komputer untuk *Entri Data* yang menggunakan sistem operasi *Microsoft Windows XP* yang tidak dapat terhubung dengan jaringan.
- Karena hanya menggunakan satu buah komputer dan tidak terhubung dengan jaringan, maka sistem ini tidak memiliki komputer server.
- Informasi yang dihasilkan dicetak dalam bentuk laporan atau disimpan ke media penyimpanan sebelum dikirim oleh petugas kepada pimpinan.
- Pengoperasian sistem tidak terlalu rumit sehingga cukup mudah dimengerti dan digunakan oleh petugas atau operator.

3.1.2. Kekurangan Sistem Saat Ini

Kekurangan yang dimiliki sistem saat ini antara lain :

- Apabila harus didistribusikan ke setiap divisi yang ada di Hotel Victory Batu Malang akan kesulitan karena sistem yang digunakan bukan aplikasi *client-server*.
- Setiap pergantian periode pelaporan, aplikasi tersebut harus disimpan dengan nama yang berbeda sesuai periode pelaporan. Dengan cara seperti ini, ada banyak tumpukan file aplikasi di setiap komputer sehingga cukup sulit dalam pencarian data karena terdapat tumpukan file di setiap komputer.

3.1.3. Pengguna Sistem

Pengguna Sistem Informasi Front Office Hotel Victory Batu Malang antara lain :

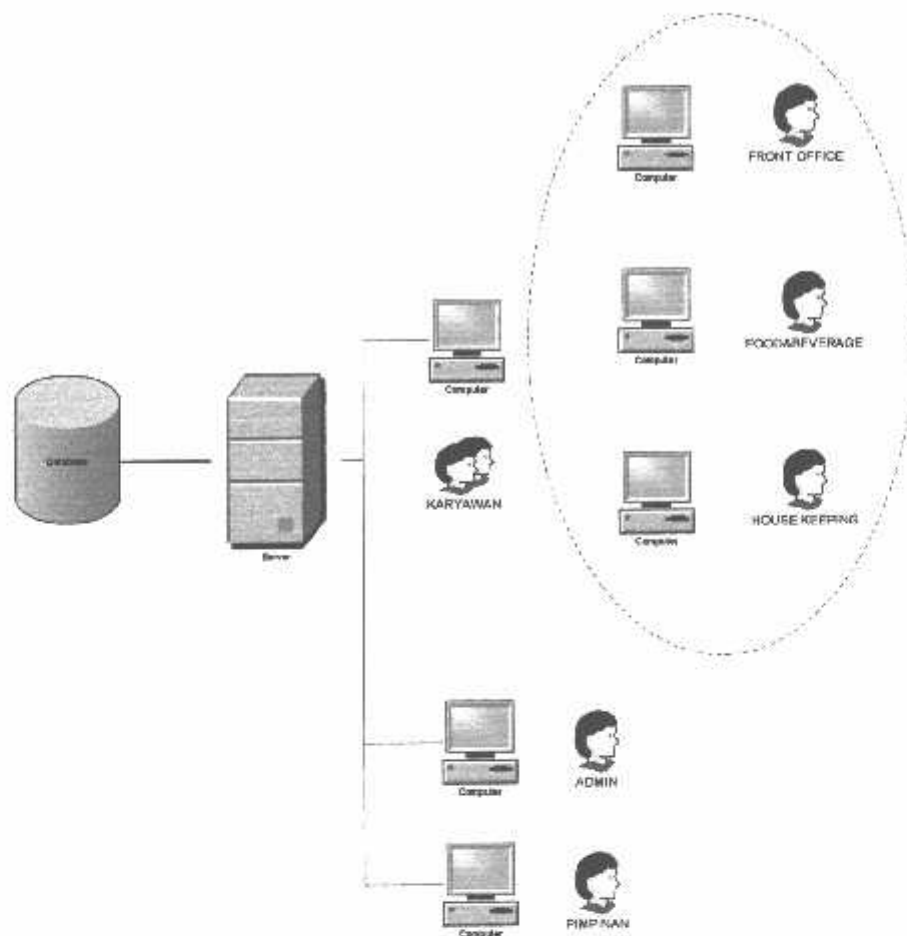
- Administrator, dalam hal ini adalah administrator Hotel Victory Batu Malang, yaitu pengguna yang memiliki hak akses penuh terhadap aplikasi Sistem Informasi Front Office pada Hotel Victory Batu Malang.
- Pimpinan, yaitu direktur serta dewan komisaris yang memiliki hak akses untuk menggunakan aplikasi Sistem Informasi Front Office pada Hotel Victory Batu Malang.
- Operator, yaitu pengguna yang akan memasukkan data-data ke aplikasi. Operator disini dibedakan hak aksesnya sesuai dengan wewenang dan tugasnya masing-masing.

Setiap pengguna tersebut di atas dibatasi oleh hak akses aplikasi kecuali administrator yang memiliki hak akses penuh termasuk menentukan hak akses pengguna maupun menambah pengguna sistem.

3.1.4. Spesifikasi Sistem

Sesuai dengan tujuan Perancangan dan Pembuatan Sistem Informasi Front Office pada Hotel Victory Batu, maka spesifikasi sistem adalah sebagai berikut :

1. Sistem informasi dengan model *client - server*, dimana aplikasi *client* dapat mengakses basis data.
2. Sistem basisdata *server* menggunakan *SQL Server 2000* dengan menggunakan satu buah *server* dengan sistem operasi *Windows XP*.
3. Sebuah modul pelaporan yang dapat memberikan data-data dan rekapan yang akurat yang sangat berguna untuk mengetahui pemasukan tiap bulan hotel.
4. Sistem informasi ini, diharapkan dapat membuat kerja para karyawan di Hotel Victory Batu menjadi lebih efektif dan efisien.
5. Sistem informasi ini, terutama bagian aplikasi *client* diharapkan dapat dengan mudah dipakai oleh karyawan, administrator, dan pimpinan yang mengoperasikan sistem ini.



Gambar 3.1 Desain Sistem

3.2 Perancangan Sistem

3.2.1 Data Flow Diagram (DFD)

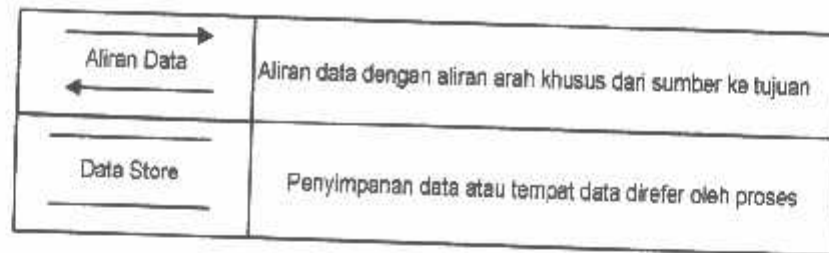
Data Flow Diagram (DFD) adalah suatu diagram yang menggunakan notasi-notasi untuk arus dari data sistem, yang penggunaannya sangat membantu untuk memahami sistem secara logika, terstruktur dan jelas.^[7] Atau DFD bisa juga dikatakan sebagai suatu model logika data atau proses yang dibuat untuk menggambarkan dari mana asal data dan kemana tujuan data yang keluar dari sistem, dimana data disimpan, proses apa yang menghasilkan data tersebut dan interaksi antara data yang tersimpan dan proses yang dikenakan pada data

tersebut. DFD ini sering disebut juga dengan nama *Bubble chart*, *Bubble diagram*, model proses, diagram alur kerja, atau model fungsi. DFD ini adalah salah satu alat pembuatan model yang sering digunakan, khususnya bila fungsi-fungsi sistem merupakan bagian yang lebih penting dan kompleks dari pada data yang dimanipulasi oleh sistem. Dengan kata lain, DFD adalah alat pembuatan model yang memberikan penekanan hanya pada fungsi sistem. DFD ini merupakan alat perancangan sistem yang berorientasi pada alur data dengan konsep dekomposisi dapat digunakan untuk penggambaran analisa maupun rancangan sistem yang mudah dikomunikasikan oleh profesional sistem kepada pemakai maupun pembuat program.

DFD terdiri dari *context diagram* dan diagram rinci (*DFD Levelled*). Context diagram berfungsi memetakan model lingkungan (menggambarkan hubungan antara entitas luar, masukan dan keluaran sistem), yang direpresentasikan dengan lingkaran tunggal yang mewakili keseluruhan sistem. *DFD levelled* menggambarkan sistem sebagai jaringan kerja antara fungsi yang berhubungan satu sama lain dengan aliran dan penyimpanan data, model ini hanya memodelkan sistem dari sudut pandang fungsi. ^[7]

Berikut ini merupakan simbol-simbol yang biasa digunakan di DFD :

Symbol	Keterangan
	Entitas Eksternal, dapat berupa orang / unit terkait yang berinteraksi dengan sistem tapi di luar sistem
	Orang, unit yang mempergunakan atau melakukan transformasi data. komponen fisik tidak diidentifikasi



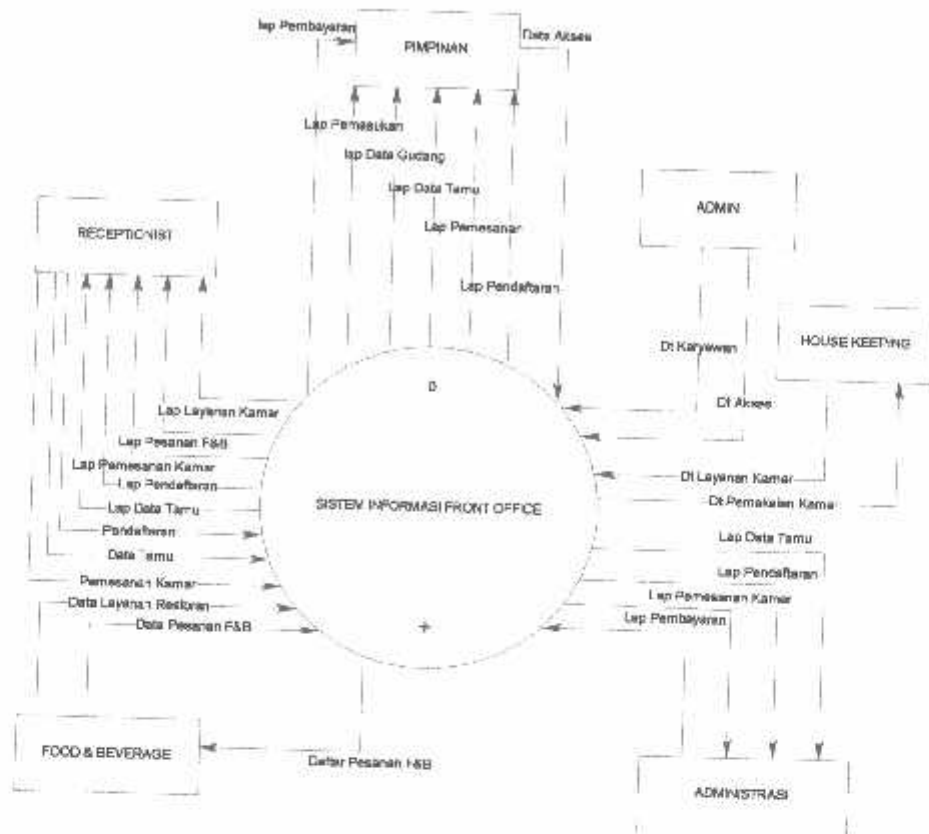
Gambar 3.2 Simbol Utama DFD

3.2.2. Alur Diagram DFD

Alur Diagram DFD merupakan langkah awal atau perancangan awal untuk membuat Data Flow Diagram (DFD), dimana dalam alur ini di gambarkan proses-proses dari DFD, dari proses-proses ini dapat diketahui proses mana yang dapat dijabarkan atau diturunkan.

3.2.3. Diagram Konteks (*Context Diagram*)

Dalam merancang suatu aplikasi, diperlukan suatu desain sistem yang dibuat dengan pemodelan proses. Model proses dari Sistem Informasi Administrasi Perhotelan ini dibuat dalam bentuk diagram yaitu Diagram Konteks (*Context Diagram*). Diagram Konteks menunjukkan bagaimana alur kerja sistem informasi secara nyata dan menjelaskan hubungan sistem dengan lingkungan atau kesatuan luar. Dalam hal ini, Diagram Konteks melibatkan kesatuan luar, yaitu Administrator, Pimpinan, Receptionist, Restourant, House Keeping dan Kasir. Dimana satu sama lain saling terkait dalam Sistem Informasi Administrasi Perhotelan.



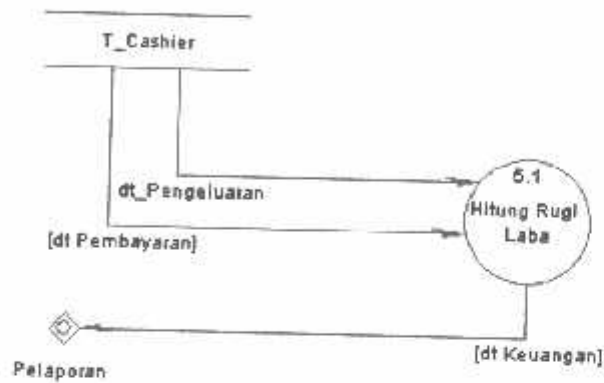
Gambar 3.3 Diagram Konteks Sistem Informasi Front Office pada Hotel Victory
Batu Malang

3.2.4. Data Flow Diagram (DFD) Level 1

Untuk memperjelas proses pada Diagram Konteks, maka perlu dilakukan sebuah pengembangan ke level berikutnya. Dimana level tersebut merupakan Data Flow Diagram (DFD) Level 1. Data Flow Diagram (DFD) Level 1 ini merupakan penjabaran dari proses Sistem Informasi Administrasi Perhotelan pada Hotel Victory Batu Malang yang terdapat pada pada diagram konteks (*Context Diagram*) dimana yang memuat proses-proses yang ada dalam sistem secara garis besar dan keseluruhan. DFD level 1 juga mencantumkan kesatuan luar yang berhubungan dengan system. Proses ini dapat dilihat pada gambar 3.4

3.2.5.1. DFD Level 2 Proses 1 (Proses Akses)

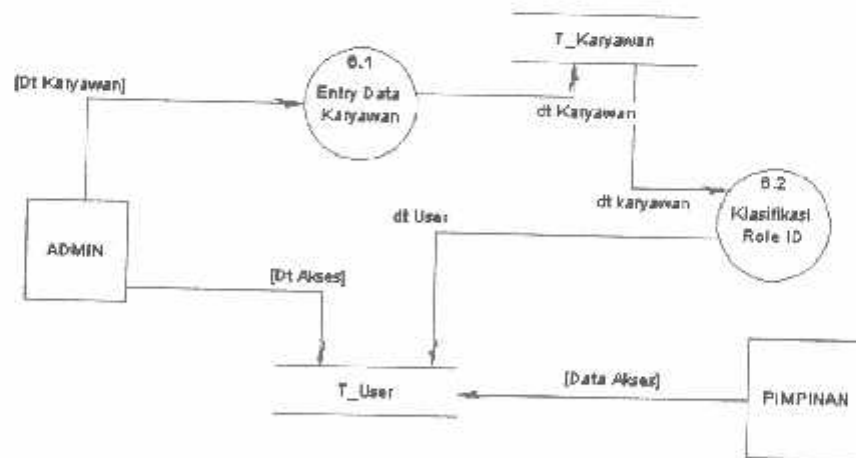
Proses yang pertama adalah Proses untuk mengentry data karyawan, dimana data tersebut meliputi nama, alamat, tempat/tanggal lahir serta data-data pribadi tiap karyawan.



Gambar 3.5 DFD Level 2 Proses 1 (Proses Akses)

3.2.5.2. DFD Level 2 Proses 2 (Proses Keuangan)

Proses yang kedua adalah proses untuk melakukan perhitungan rugi laba melalui selisih antara pemasukan serta pengeluaran hotel secara periodik.



Gambar 3.6 DFD Level 2 Proses 2 (Proses Keuangan)

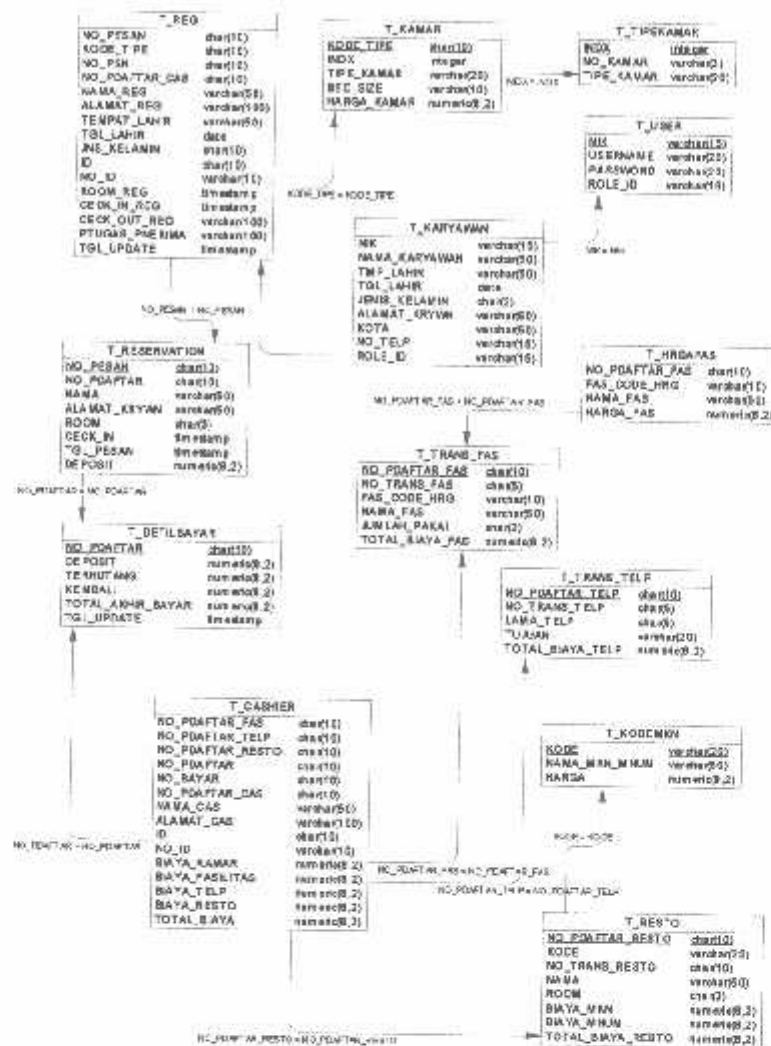
3.2.6. Desain Basisdata

Disini terdapat basisdata yang digunakan pada aplikasi Sistem Informasi Front Office pada Hotel Victory Batu Malang ini, yaitu basisdata untuk Data Tamu, Restoran, House Keeping dan Akuntansi yang menjadi satu bagian dari basisdata Sistem Informasi Front Office pada Hotel Victory Batu Malang itu sendiri.

Basisdata untuk Data Tamu berfungsi untuk menyimpan data-data seperti Daftar Reservasi dan Registrasi Tamu secara periodik. Basisdata untuk Restoran berfungsi untuk menyimpan data-data tentang transaksi yang dilakukan oleh restoran secara periodik. Basisdata untuk House Keeping berfungsi untuk menyimpan data-data seperti transaksi telepon serta transaksi penggunaan fasilitas secara periodik. Sedangkan basisdata untuk akuntansi yaitu sistem kontroling data keuangan yang akan dikelola oleh sistem.

3.2.6.1. Relasi Antar Tabel

Relasi antar tabel pada basisdata Sistem Informasi Front Office pada Hotel Victory Batu Malang digambarkan dalam bentuk konsep atau *Conceptual Data Model (CDM)* dan dalam bentuk fisik atau *Physical Data Model (PDM)*, seperti yang ditunjukkan pada gambar berikut :



Gambar 3.8 Physical Data Model (PDM)

Hotel Victory Batu Malang.

3.2.6.2. Struktur Tabel – Tabel Yang Digunakan

Berdasarkan dari diagram di atas, maka tabel-tabel yang digunakan antara lain :

1. T_Reservation

Table ini berfungsi untuk menyimpan data pesanan kamar tamu sesuai dengan tanggal mereka memesannya.

Tabel 3.1 Tabel Reservasi

	Column Name	Data Type	Length	Allow Nulls
🔑	no_pesanan	char	10	
	nama	varchar	100	✓
	alamat	varchar	100	✓
	room	char	3	✓
	cek_in	datetime	8	✓
	tgl_pesanan	datetime	8	✓
	deposit	money	8	✓
	tgl_update	datetime	8	✓
▶				

2. T_Reg

Tabel ini berfungsi untuk menyimpan data tamu-tamu yang telah melakukan proses registrasi.

Tabel 3.2 Tabel Registrasi

	Column Name	Data Type	Length	Allow Nulls
	no_pesanan	char	10	
🔑	no_Pdaftar	char	10	
	nama	varchar	100	✓
	alamat	varchar	100	✓
	tempat_lahir	varchar	50	✓
	tgl_lahir	datetime	8	✓
	jns_kelamin	char	10	✓
	ID	char	10	✓
	no_ID	varchar	50	✓
	room	char	3	✓
	cek_in	datetime	8	✓
	cek_out	datetime	8	✓
	ptugas_penerima	varchar	100	✓
	tgl_Update	datetime	8	✓
▶				

3. T_user

Tabel ini berfungsi untuk menyimpan daftar username serta password karyawan selaku pengguna sistem ini.

Tabel 3.3 Tabel User

	Column Name	Data Type	Length	Allow Nulls
	nik	varchar	15	
	username	varchar	20	✓
	password	varchar	20	✓
	role_ID	varchar	15	✓

4. T_cashier

Tabel ini berfungsi untuk melakukan perhitungan semua biaya persewaan kamar serta fasilitas-fasilitas lain yang telah digunakan oleh tamu.

Tabel 3.4 Tabel Pembayaran atau kasir

	Column Name	Data Type	Length	Allow Nulls
	no_bayar	char	10	✓
	no_pdaftar	char	10	
	nama	varchar	100	✓
	alamat	varchar	100	✓
	ID	char	10	✓
	no_ID	varchar	50	✓
	biaya_kamar	money	8	✓
	biaya_fasilitas	money	8	✓
	biaya_telp	money	8	✓
	biaya_resto	money	8	✓
	total_biaya	money	8	✓

5. T_detilBayar

Tabel ini berfungsi untuk melakukan perhitungan secara detail biaya sewa kamar dalam Hotel Victory Batu.

Tabel 3.5 Tabel Detil Bayar

	Column Name	Data Type	Length	Allow Nulls
🔑	no_Pdaftar	char	10	
	deposit	money	8	✓
	terhutang	money	8	✓
	kembali	money	8	✓
	total_akhir_bayar	money	8	✓
	tgl_update	datetime	8	✓
▶				

6. T_hrgaFas

Tabel ini berfungsi untuk menyimpan data biaya-biaya fasilitas yang disediakan oleh Hotel Victory Batu.

Tabel 3.6 Tabel Harga Fasilitas

	Column Name	Data Type	Length	Allow Nulls
🔑	kode_fas	varchar	10	
	nama_fas	varchar	50	
	harga_fas	money	8	✓
▶				

7. T_kamar

Tabel ini berfungsi untuk menyimpan data biaya-biaya kamar yang disediakan oleh Hotel Victory Batu.

Tabel 3.7 Tabel Harga Kamar

	Column Name	Data Type	Length	Allow Nulls
🔑	kode_tipe	char	10	
	tipe_kamar	varchar	20	✓
	bed_size	varchar	10	✓
	harga_kamar	money	8	✓
▶				

8. T_karyawan

Tabel ini berfungsi untuk menyimpan detail data karyawan Hotel Victory Batu Malang seperti nama, alamat, tempat/tgl lahir dan lain-lain.

Tabel 3.8 Tabel Data Karyawan

	Column Name	Data Type	Length	Allow Nulls
PK	nik	varchar	20	
	nama_karyawan	varchar	50	✓
	tmp_lahir	varchar	50	✓
	tgl_lahir	datetime	8	✓
	jenis_kelamin	char	2	✓
	alamat	varchar	50	✓
	kota	varchar	50	✓
	no_telp	varchar	15	✓
	role_ID	varchar	20	✓
▶				

9. T_kodeMkn

Tabel ini berisi daftar harga makanan dan minuman yang tersedia di restoran Hotel Victory Batu.

Tabel 3.9 Tabel Kode harga Makan/Minum

	Column Name	Data Type	Length	Allow Nulls
PK	kode	varchar	10	
	[nama_mkn/mnum]	varchar	50	✓
	harga	money	8	✓
▶				

10. T_resto

Tabel ini berisi total biaya makanan atau minuman yang harus dibayar oleh pengunjung restoran maupun tamu Hotel Victory Batu.

Tabel 3.10 Tabel Data Restoran

	Column Name	Data Type	Length	Allow Nulls
	no_trans_resto	char	10	✓
	no_pdaftar	char	10	
	nama	varchar	100	✓
	room	char	3	✓
	biaya_makan	money	8	✓
	biaya_minum	money	8	✓
	total_biaya_resto	money	8	✓

11. T_tipekamar

Tabel ini berfungsi untuk menyimpan daftar nomor kamar serta tipe masing-masing kamar dari tiap nomor tersebut.

Tabel 3.11 Tabel Tipe Kamar

	Column Name	Data Type	Length	Allow Nulls
	no_kamar	varchar	3	✓
	tipe_kamar	varchar	20	✓
	indX	int	4	

12. T_trans_fas

Tabel ini berfungsi untuk melakukan perhitungan biaya fasilitas apa saja yang telah digunakan oleh tamu Hotel Victory Batu.

Tabel 3.12 Tabel Transaksi Fasilitas

	Column Name	Data Type	Length	Allow Nulls
	no_trans_fas	char	5	✓
	no_Pdaftar	char	10	
	kode_fas	char	5	✓
	nama_fas	varchar	30	✓
	jumlah_pakai	char	2	✓
	total_biaya_fas	money	8	✓

13. T_trans_telp

Tabel ini berfungsi untuk melakukan perhitungan biaya telepon yang telah digunakan oleh tamu Hotel Victory Batu.

Tabel 3.13 Tabel Transaksi Telepon

	Column Name	Data Type	Length	Allow Nulls
	no_trans_telp	char	5	✓
	no_Pdaftar	char	10	
	lama_telp	char	5	✓
	tujuan	varchar	20	✓
	total_biaya_telp	money	8	✓

3.2.7. Desain Antarmuka Aplikasi

Sesuai dengan spesifikasi sistem di atas, sistem informasi ini diharapkan dapat dengan mudah dipakai oleh banyak orang atau operator yang mengoperasikan sistem ini. Untuk itu harus dibuat desain antarmuka yang mudah dipahami dan tidak terlalu rumit. Dimana untuk lebih jelas dari desain antarmuka ini dapat dilihat pada bab Empat yaitu implementasi sistem atau pengujian dari program yang dibuat.

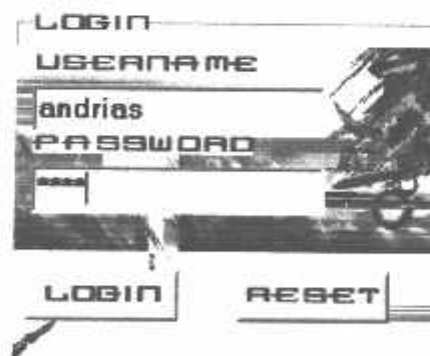
BAB IV

IMPLEMENTASI DAN PENGUJIAN SISTEM

Setelah mendesain perancangan sistem, pada bab ini akan dijelaskan tentang pengujian sistem atau pengujian dari program yang dibuat. Pengujian program ini ditujukan untuk memberikan gambaran secara keseluruhan dari program yang telah dibuat. Hal ini berguna untuk memberikan suatu bukti bahwa program yang telah dibuat sudah siap untuk di implementasikan. Dimana untuk pembuktiannya dapat dilihat pada gambar-gambar atau capture dari program yang telah dibuat.

4.1 Form Login

Untuk dapat mengakses menu-menu yang ada di program ini Pengguna atau Administrator harus melakukan Login terlebih dahulu dengan memasukan Username dan Password. Tujuannya untuk memferifikasikan data login dan menampilkan form halaman utama sesuai dengan hak akses yang dimiliki oleh pemakai aplikasi.



Gambar 4.1 Form Login

4.2 Form Utama

Desain Menu Utama adalah menu untuk mengakses seluruh transaksi-transaksi yang dilakukan baik oleh divisi Front Office, divisi Food and Beverage maupun divisi House Keeping. Dari menu login pada gambar sebelumnya adalah kunci dari otoritas yang akan dipakai atau yang tersedia pada menu utama. Sehingga setiap bagian atau divisi memiliki level menu yang berbeda-beda sesuai dengan hak akses yang dimiliki. Sedangkan untuk menu utama ini selaku administrator dapat menampilkan seluruh fasilitas yang tersedia pada Sistem Informasi Front Office pada Hotel Victory ini. Namun kewajiban administrator pada Hotel Victory, tetap akan diserahkan kepada pimpinan Hotel Victory.

Hotel Menu Transaksi Laporan Guest Guest

HOTEL VICTORY
Jl. Raya Junggo 107 Kota Batu - Malang
Telp. (0341) 893011, 892988 Fax. (0341) 893012

August 2010

Su	Mo	Tu	We	Th	Fr	Sa
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

Today: 8/21/2010

LOGIN

USERNAME

PASSWORD

LOGIN RESET

Status

Gambar 4.2 Form Menu Utama

4.3 Main Menu

Pada Main Menu terdapat beberapa menu pilihan yang ditampilkan diantaranya :

1. Registration (Tab Registrasi, Tab Reservasi, Tab Ceckout)
2. Restaurant.

4.3.1 Form Registration

Form Registration merupakan form yang berisikan Data Registrasi Tamu, Data Reservasi Kamar dan Data Tamu Ceck Out. Dalam form ini terdapat tampilan Status Kamar yang bisa dijadikan acuan tentang data kamar mana saja yang sudah terisi ataupun belum serta juga menunjukkan data kamar yang telah dipesan.

[illegible]

Gambar 4.3 Form Registration

4.3.2 Tab Registrasi

Tab Registrasi berisikan data tamu yang telah melakukan proses registrasi atau proses daftar dimana data-data tersebut terdiri dari nama, alamat, tempat/tanggal lahir, jenis kelamin, no ID, data tanggal ceckin dan ceckout serta no kamar yang disewa.



HOTEL VICTORY

Jl. Raya Junjuga 107 Kota Batu - Malang
Telp. (0341) 893011, 892988 Fax. (0341) 893012

Status Kamar

Juni 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31

Juli 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31

Agustus 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31

September 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31

Oktober 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31

November 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31

Desember 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31

Tagihan Kamar

Deposit: 100000
Kembali: 0

Tertunggak: 250000
Total Bayar: 350000

Tagihan Tamu

Deposit: 100,000.00
Tertunggak: 350,000.00
Total Bayar: 350,000.00

Tab. Registrasi | **Tab. Reservasi** | **Tab. CekOut**

Registrasi

No Pendaftaran: 2010.0608.1132 No Pendaftaran: 2010.0806.113415

Nama: sedria

Alamat: Malang

Tempat Lahir: Surabaya Tanggal Lahir: 7/ 8/1987

Jenis Kelamin: Laki-laki / Perempuan

ID: KTP No ID: 23345677

Cek In: 8/ 8/2010 Cek Out: 14/ 8/2010

Ruang: 232 Tipe Kamar: STANDARD

Pembayaran: Tunai Harga / Kamar: 350000

TAMBAH SIMPAN HAPUS

Data Tamu

nama	jenis kelamin	alamat	tempat lahir	tanggal lahir	id
sedria	Laki-laki	Malang	Surabaya	7/ 8/1987	23345677

Data Reservasi

no reservasi	nama	alamat	tempat lahir	tanggal lahir	id
2010.0508.2330	sedria	Malang	Surabaya	7/ 8/1987	23345677
2010.0508.2337	sedria	Malang	Surabaya	7/ 8/1987	23345677
2010.0508.2340	sedria	Malang	Surabaya	7/ 8/1987	23345677

Gambar 4.4 Tab Registrasi

4.3.3 Tab Reservasi

Tab Reservasi berisikan data tamu yang melakukan proses pesan kamar dimana data-data yang harus diisi terdiri dari nama, alamat, tempat/tanggal lahir, tanggal ceekin serta kamar yang dipesan.

Gambar 4.5 Tab Reservasi

Tab Ceck Out berisikan data tamu yang melakukan proses ceck out atau telah meninggalkan hotel. Data yang di simpan dalam Tab Ceck Out ini berisi tentang data identitas tamu yang telah menginap di dalam hotel.

Gambar 4.6 Tab Ceck Out

4.3.5 Form Restaurant

Form Restaurant merupakan form yang berisikan data transaksi restoran oleh tamu hotel yang menginap maupun pengunjung restoran yang hanya sekedar ingin menikmati menu yang disediakan oleh restoran hotel tanpa harus menginap.

HOTEL VICTORY
Jl. Raya Jember 907 Kota Batu - Malang
Telp. (0341) 693011, 692988 Fax. (0341) 693012

Restaurant
No Transaksi: 2010.2108.070843 No Pendaftaran: _____
Nama: _____
Room: _____

Pesanan menu
Kode Menu: 0003 Ayam Goreng tepung
Banyak: 1 Harga: 0000
Jumlah: 8,000.00 Total Biaya: 18000

Data Restaurant

no trans resto	no Pdftar	nama	room	no_menu	ht
2010.2108.070225				Capucino	
2010.2108.070501				Fuyunghai	
2010.2108.070501				Kari Ayam	
2010.2108.070501				Mujah Goreng	
2010.2108.070843				Sayur Asam	
2010.2108.070843				Ayam Goreng tepung	

Buttons: TAMBAH, SIMPAN, UBAH, CETAK, KELUAR, MENU

Gambar 4.7 Form Restaurant

4.4 Menu Transaksi

Pada Menu Transaksi terdapat berbagai menu pilihan yang ditampilkan. Dimana Menu Transaksi digunakan untuk melakukan beberapa macam transaksi. Menu-menu yang ditampilkan diantaranya :

- 1. Form Biaya Fasilitas
- 2. Form Biaya Telephone
- 3. Form Tagihan Tamu

4.4.1 Form Biaya Fasilitas

Form Biaya Fasilitas merupakan form yang berisikan Biaya-biaya fasilitas yang harus dibayar oleh tamu selama menginap tentunya setelah menggunakan fasilitas-fasilitas tambahan yang disediakan oleh Hotel Victory Batu.

HOTEL VICTORY
Jl. Raya Junggo 107 Kota Batu - Malang
Telp. (0341) 893011, 893005 Fax. (0341) 893012

Fasilitas

No Transaksi	2010.0608.1136
No Pendaftaran	2010.0806.112415
Kode Fasilitas	0002
Nama Fasilitas	1 kamar
Harga	20000
Jumlah Pemakaian	2
Total Biaya	40000

HITUNG

Data Fasilitas

no trans fas	no Pendaftaran	kode
2010.0608.0809	2010.0804.1002	
2010.0608.1136	2010.0806.1002	

TAMBAH **SIMPAN** **KELUAR**

Gambar 4.8 Form Biaya Fasilitas

4.4.2 Form Biaya Telephone

Form Biaya Telephone merupakan form yang berisikan data-data penggunaan fasilitas telephone yang dilakukan oleh tamu selama menginap.

HOTEL VICTORY
Jl. Raya Junggo 107 Kota Batu - Malang
Telp. (0341) 893011, 893005 Fax. (0341) 893012

Data Telephone

No Transaksi	2010.0408.2006
No Pendaftaran	2010.0804.200247
Lama Menelpon	4 / menit
Tujuan	Jakarta
Biaya Telepon	8000

HITUNG

Pemakaian Telephone

no trans telp	no Pendaftaran	lama telp	tujuan	total biaya tel
2010.0408.2006	2010.0804.200247	4	Jakarta	8
2010.0508.1136	2010.0805.2	2	malang	1

TAMBAH **SIMPAN** **KELUAR**

Gambar 4.9 Form Biaya Telephone

4.4.3 Form Tagihan Tamu

Form Tagihan Tamu merupakan form yang berisikan tentang semua tagihan yang harus dibayar oleh tamu hotel. Yang termasuk didalam tagihan tamu adalah seperti yang tampil pada gambar di bawah ini

HOTEL VICTORY
Jl. Raya Jengg 107 Kota Batu - Malang
Telp. (0341) 693011, 693985 Fax. (0341) 693012

Pembayaran

No Pembayoran: 4356
No Pendaftaran: 4356
Nama: Fais Fauzi
Alamat: Malang
ID: KTP No ID: KTP 9876.01
Sistem Bayar: ☒ Tunai ☐ Kartu
Biaya Kamar: 1000000
Biaya Fasilitas: 0
Biaya Telepon: 2500
Biaya Restaurant: 0
Total Biaya: 1002500 **HITUNG**

Pembayaran

no. bayar	no. Pdaftar	nama	alamat
0022	0022	Saiphul Anwar	Malang
3252	0061	sugianto	Jl. Besar Ijen
45/466	11235	wawan	ngoro jembangan
4356	4356	Fais Fauzi	Malang

UBAH SIMPAN CETAK KELUAR

Gambar 4.10 Form Tagihan Tamu

4.5 Menu Pelaporan

Pada Menu Pelaporan terdapat berbagai menu pilihan yang ditampilkan. Dimana menu Pelaporan dapat menampilkan laporan-laporan yang terjadi secara keseluruhan. Laporan-laporan yang ditampilkan diantaranya adalah :

1. Lap Data Tamu
2. Lap Reservasi
3. Lap Registrasi
4. Lap Restaurant
5. Lap Rugi Laba

4.5.1 Lap Data Tamu

Pada Lap Data Tamu merupakan laporan yang berisikan data semua tamu yang sedang menginap maupun yang sudah melakukan cekout.

Room	Nama	Alamat	L/P	Tmp/Tgl Lahir	Check In	Check Out	noID
344	andi hossaini	malang	P	malang 8/7/1970	8/4/2010 9:44	8/6/2010 9:44	KT640835
123	anu	malang	L	tutungagung 7/9/1985	8/18/2010 9:34	8/19/2010 9:34	KTJ58510819
114	agung jurnedi	surabaya	P	surabaya 8/7/2010	8/4/2010 10:48	8/4/2010 10:48	KT43876

Gambar 4.11 Form Lap Data Tamu

4.5.2 Lap Reservasi

Pada Lap Reservasi merupakan laporan yang berisikan data semua tamu yang melakukan pemesanan/reservasi kamar sebelum melakukan checkin.

Room	Nama	Alamat	Tanggal Reson	Tanggal Check In	Deposit
238	nina	malang	5/23/2010	8/6/2010 11:38:10 PM	0
347	andi	kupang	5/23/2010	8/6/2010 11:38:10 PM	0
236	eka	pacuruban	5/23/2010	8/6/2010 11:38:10 PM	0
242	jaki	bhalal	8/17/2010 1:30:15 PM	8/18/2010 1:30:15 PM	0
123	raditya	kawang malang	8/18/2010 10:41:18 AM	8/18/2010 10:41:18 AM	50000

Gambar 4.12 Form Lap Reservasi

4.5.3 Lap Registrasi

Pada Lap Registrasi merupakan laporan yang berisikan data semua tamu yang sudah melakukan proses registrasi dengan atau tanpa melakukan proses reservasi terlebih dahulu.



HOTEL VICTORY
Jl. Raya Jember 107 Kota Batu - Malang
Telp. (0341) 894015, 894016
Fax. (0341) 894018

Data Registrasi Tamu

Room	Nama	Alamat	L/P	Tipe/Tgl Lahir	Cek In	Cek Out	No ID
101	suano dudi	jakarta pusat	L	lampung 7/8/1980	8/13/2010	8/15/2010	8KTP 124 1765 1575 477
123	andi sja	surabaya	L	malang 7/8/1970	8/4/2010	8/5/2010	10KTP 455788
114	agung dumiadi	surabaya	P	surabaya 7/8/2010	8/4/2010	8/4/2010	10KTP 43578
109	hen wiyono	bilar	L	bilar 7/8/1970	8/3/2010	8/6/2010	10KTP 345057987
362	ristanto	banyuwangi	L	jombang 7/8/1987	8/4/2010	8/8/2010	7KTP 5437 3426 2864
300	joko puapito	sidoarjo	L	malang 7/8/1978	8/4/2010	8/5/2010	9KTP 2545 755 875
120	nucepta	jember	L	7/8/1979	8/4/2010	8/6/2010	8KTP 8786 78568
227	yudimira	kediri	L	kediri 7/8/1980	8/6/2010	8/7/2010	11KTP 86434567
102	kepel	penggala 39	P	waxo 7/8/2010	8/8/2010	8/7/2010	12KTP 5437
232	andras	malang	L	jombang 7/8/1987	8/8/2010	8/6/2010	11KTP 28345677
120	saya	malang	L	malang 7/8/1980	8/17/2010	8/18/2010	8KTP 5786878887
2	sidi	d	L	d 7/8/2010	8/16/2010	8/16/2010	5KTP 1 4
243	suwanto	malang	L	malang 7/8/1987	8/19/2010	8/21/2010	1KTP 8646343

Page 1 of 1

Gambar 4.13 Form Lap Registrasi

4.5.4 Lap Restaurant

Pada Lap Restaurant merupakan laporan yang berisikan data transaksi penjualan yang dilakukan oleh bagian restaurant/food and beverage.



HOTEL VICTORY
Jl. Raya Jember 107 Kota Batu - Malang
Telp. (0341) 894015, 894016
Fax. (0341) 894018

Nota Restoran

no nota : 2010.0408.1748

8/21/2010 8.50.21 AM

Nama Menu	Banyak	Harga Menu	Total
Kakap Goreng Tepung	2	25000	50000
Kakap Goreng Tepung	4	25000	100000
Kakap Goreng Tepung	2	25000	50000
Fuyunghai	2	14000	28000
Kare Ayam	1	7500	7500
Sup Bata-bata Tahu	2	12000	24000
Lala Goreng	1	7000	7000
Ayam Goreng Jawa	2	9000	18000
Ayam Goreng Tepung	3	8000	24000
Tahu Tempa Goreng	2	4000	8000
Lala Goreng	2	7000	14000
Lala Goreng	4	7000	28000
Lala Goreng	2	7000	14000
Bata Daging	2	10000	20000
Bata Ayam	2	10000	0

Page 1 of 2

Gambar 4.14 Form Lap Restaurant

4.5.5 Lap Rugi Laba

Pada Lap Rugi Laba merupakan laporan yang berisikan data semua pemasukan dan pengeluaran yang terjadi pada Hotel Victory secara berkala sehingga bisa didapatkan hasil Rugi/Laba yang terjadi.



HOTEL VICTORY

Jl. Raya Jungga 107 Kota Batu - Malang

Telp. (0341) 893811, 893888 Fax. (0341) 893812

Rugi / Laba

No Laporan

Dari tanggal ▼

Pemasukan :

Kamar	750000
Restaurant	199000
Fasilitas	40000
Telephone	4500

Sampai tanggal ▼

Pengeluaran :

Listrik	200000
Air	300000
Lain-lain	50000

Total Pemasukan

Hasil Rugi / Laba

Total Pengeluaran

HITUNG

TAMBAH

CETAK

SIMPAN

HAPUS

Laporan Rugi/Laba

no_lapR/L	msk_kmr	msk_resto	msk_fas	msk_telp
2010.0408.1942	12000	5000	10000	
2010.0808.1502	0	0	0	
▶ 5378486	750000	199000	40000	

Gambar 4.15 Form Lap Rugi Laba

4.6 Menu Akses

Pada Menu Akses terdapat beberapa pilihan hak akses, diantaranya hak akses untuk mengentry data kamar dan data karyawan.

4.6.1 Form Entry Data Kamar

Form Entry Data Kamar merupakan form yang digunakan untuk menginputkan data tipe-tipe kamar yang ada dalam hotel serta harga dari masing-masing kamar tersebut.

Entry Data Kamar

HOTEL VICTORY
Jl. Raya Junggo 107 Kota Batu - Malang
Telp. (0341) 899011, 899006 Fax. (0341) 899012

Entry Data Kamar

Kode Tipe:

Nama Tipe:

Bed Size:

Harga Kamar:

TAMBAH HAPUS
SIMPAN KELUAR
ENTRY NO KAMAR

	kode tipe	tipe kamar	bed_size	harga_kamar
☐	Fa004	FAMILY	4 pax	400000
☐	Jr001	JUNIOR	2pax	250000
☐	Se005	SUITE	2pax	600000
☐	St002	STANDARD	2 pax	350000
☒	Su003	SUPERIOR	4 pax	500000

Gambar 4.16 Form Entry Data Kamar

4.6.2 Form Entry Data Karyawan

Form Entry Data Karyawan form yang digunakan untuk register atau menginputkan data pribadi para karyawan Hotel Victory Batu. Dimana setiap karyawan wajib memberikan informasi tentang data pribadinya.

Entry Data Karyawan

HOTEL VICTORY
 Jl. Raya Jungsro 197 Kota Batu - Malang
 Telp. (0341) 893011, 892985 Fax. (0341) 693012

Entry Data Karyawan

NIK: 20023460

Nama: Andrias

Tempat lahir: Jombang Tanggal Lahir: 8/23/2010

Jenis Kelamin: ☒ Laki-Laki ☐ Perempuan

Alamat: Jl Perusahaan gg V no.88A

Kota: Malang

No. Telp: 08165482893 Role ID: Admin

TAMBAH SIMPAN HAPUS USER ID

nik	nama_karyawan	tmp_lahir	tgl_lahir	jenis_kelamin	alamat	kota	no_telp	role
0000001	Heri	Jakarta	12/14/1969	L	Jl. A. Yani 90	Surabaya	085263782881	Own
2001345	Heru Kuswanto	Jombang	1/23/1970	L	Jl. Slamet Riyad	Malang	081334515409	Rece
2001345	Dwi Agus SC	Solo	7/8/1978	L	Jl. Santal no. 1	Malang	085756798762	Rece
2002346	Andrias	Jombang	8/23/2010	L	Jl. Perusahaan	Malang	08165482893	Admi

Gambar 4.17 Form Entry Data Karyawan



BAB V

PENUTUP

5.1. Kesimpulan

Berdasarkan hasil pengujian studi pustaka, analisa, desain, pembuatan sistem informasi serta pengujian Sistem Informasi Front Office ini dapat diambil beberapa kesimpulan antara lain :

1. Dengan fasilitas sistem pelaporan akutansi pada aplikasi ini, maka karyawan tidak perlu lagi membuat laporan akutansi secara manual karena pada saat terjadi proses registrasi maupun reservasi tamu serta transaksi restoran secara otomatis akan membuat laporan akutansi dengan sendirinya.
2. Dengan sistem baru ini semua dokumen atau berkas – berkas disimpan dalam *database* komputer sehingga kemungkinan terjadinya kerusakan atau kehilangan dokumen – dokumen tersebut dapat diminimalisir. Bahkan dokumen – dokumen tersebut dapat dicetak jika suatu saat diperlukan.
3. Kegiatan yang terjadi di setiap bagian pada Hotel Victory akan selalu terpantau melalui Sistem Informasi Front Office ini.
4. Masing-masing pengguna sistem ini dibatasi hak aksesnya yang ditentukan oleh pimpinan hotel.

5.2. Saran

Sistem Informasi Front Office ini ini masih dapat dikembangkan lebih jauh lagi karena dalam pembuatannya masih banyak menggunakan batasan karena

pertimbangan luasnya sistem dan sumber daya manusia yang akan menggunakan sistem ini. Untuk pengembangan lebih lanjut dari aplikasi ini dapat ditambahkan beberapa fasilitas lain seperti fasilitas untuk absensi kepegawain yang dapat mengontrol data beserta aktifitas pegawai.

DAFTAR PUSTAKA

- [1]. Kristanto, Andri. *Perancangan Sistem Informasi dan Aplikasinya*, Penerbit Gava Media, Yogyakarta, 2008
- [2]. Wahyono, Teguh. *Sistem Informasi : Konsep dasar, Analisis Desain dan Implementasi*, Penerbit Graha Ilmu, Yogyakarta, 2004
- [3]. Subari, Yuswanto. *Mengolah Databse dengan SQL Server 2000*, Prestika Pustaka Publisser, Jakarta, 2005
- [4]. *Tip dan Trik Pemrograman Delphi 7.0*, Penerbit Andi dan Wahana Komputer, Yogyakarta, 2007
- [5]. Martina, Inge, Ir. *Database Client/Server Menggunakan Delphi*, Elex Media Komputindo, Jakarta, 2002
- [6]. <http://world-of-programmer.blogspot.com/2010/02/cara-membuar-erd-entity-relational.html>
- [7]. <http://komputer-one.blogspot.com/>
- [8]. http://tri_s.staff.gunadarma.ac.id/Downloads/files/7067/konsep+dasar+SI.pdf



LAMPIRAN



PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG
INSTITUT TEKNOLOGI NASIONAL MALANG

**FAKULTAS TEKNOLOGI INDUSTRI
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
PROGRAM PASCASARJANA MAGISTER TEKNIK**

BNI (PERSERO) MALANG
BANK NIAGA MALANG

Kampus I : Jl. Bendungan Sigura-gura No. 2 Telp. (0341) 551431 (Hunting), Fax. (0341) 553015 Malang 65145
Kampus II : Jl. Raya Karanglo, Km 2 Telp. (0341) 417636 Fax. (0341) 417634 Malang

**BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI**

NAMA : ANDRIAS KRISTANTO WIDIGDO
NIM : 05.12.526
JURUSAN/KONSENTRASI : TEKNIK ELEKTRO/KOMPUTER DAN INFORMATIKA
JUDUL SKRIPSI : **PERANCANGAN DAN PEMBUATAN SISTEM INFORMASI
FRONT OFFICE PADA HOTEL VICTORY DI BATU-
MALANG**

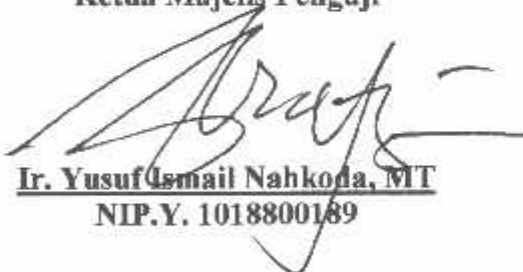
Dipertahankan dihadapan penguji skripsi jenjang program sarjana satu (S-1) pada :

Hari/Tanggal : Selasa, 24 Agustus 2010

Dengan nilai : 83,45(A) *af*

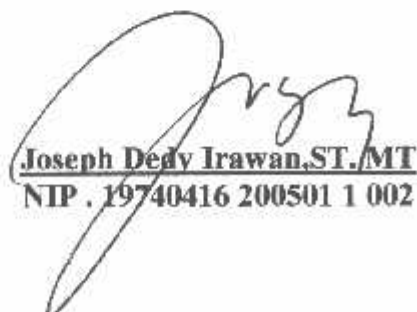
PANITIA UJIAN SKRIPSI

Mengetahui,
Ketua Majelis Penguji


Ir. Yusuf Ismail Nahkoda, MT
NIP.Y. 1018800189

ANGGOTA PENGUJI

Dosen Penguji I


Joseph Dedy Irawan, ST, MT
NIP. 19740416 200501 1 002

Dosen Penguji II


Sandy Nataly Mantja, SKom.



FORMULIR PERBAIKAN SKRIPSI

Nama : **ANDRIAS KRISTANTO WIDIGDO**
Nim : **05.12.526**
Jurusan : **T. Elektro S-1**
Konsentrasi : **T. Komputer dan Informatika**
Masa Bimbingan : **14 Juni s/d 14 Desember 2010**
Judul Skripsi : **PERANCANGAN DAN PEMBUATAN SISTEM INFORMASI
FRONT OFFICE PADA HOTEL VICTORY DI BATU-MALANG**

Tanggal	Penguji	Uraian	Paraf
24 Agustus 2010	I	1. Pesan tanggal mundur masih bisa. 2. Pesan tanggal 7-8 dan 10-11 kamar yang sama tidak bisa. 3. Tidak terjadi perubahan warna status jika pesanan dibatalkan, ceck out juga. 4. Demo ulang program. 5. Judul Administrasi Perhotelan diganti Front Office.	
	II	1. Untuk laundry keynya berdasarkan nomor kamar, otomatis muncul nama tamu. 2. Laundry hotel per-item untuk harga. 3. Perlu searching untuk data tamu. 4. Di Form cetak harus ada tanggal masuk, keluar dan jumlah hari.	

Disetujui,

Dosen Penguji I

Dosen Penguji II

Joseph Dedy Irawan, ST. MT
NIP. 19740416 200501 1 002

Sandy Nataly Mantja, SKom.

Mengetahui,

Dosen Pembimbing I

Dosen Pembimbing II

Ir. Yusuf Ismail Nakhoda, MT.
NIP. Y. 1018806189

Febriana Santi W, S.Kom, M.Kom.



INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO

Formulir Perbaikan Ujian Skripsi

Dalam pelaksanaan Ujian Skripsi Janjang Strata 1 Jurusan Teknik Elektro Konsentrasi T. Energi Listrik / T. Elektronika / T. Infokom, maka perlu adanya perbaikan skripsi untuk mahasiswa :

NAMA : ANDRINS IC
NIM : 0512526
Perbaikan meliputi :

- 1) PESAN TANGGAL MONDOK MASIH BISA ?
- 2) PESAN TGL 7-8 DAN 10-12 KAMAR
76 SAMA TIDAK BISA
- 3) TIDAK TERJADI PERUBAHAN WARNA STATUS
JIKA PESANAN DIBATALKAN, CHECK OUT JODAH
- 4) PERUBAHAN PROGRAM
- 5) JUDUL ADMINISTRASI PERHOTELAN →
FRONT OFFICE ??

Malang, 24 - 8 - 2000



INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO

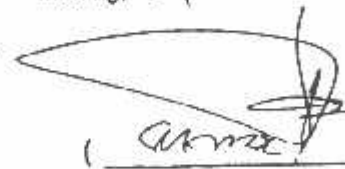
Formulir Perbaikan Ujian Skripsi

Dalam pelaksanaan Ujian Skripsi Janjang Strata 1 Jurusan Teknik Elektro Konsentrasi T. Energi Listrik / T. Elektronika / T. Infokom, maka perlu adanya perbaikan skripsi untuk mahasiswa :

NAMA : ANDHARA KARTANTO
NIM : 071216
Perbaikan meliputi :

1. UTK COUNTRY KEY NYA BELDASARKAN NO EAMAR, OTOMATIC MONITOR MAMPA TAMU
2. LAUNDRY HOTEL PER-ITEM UTK HARSA
3. ~~PER~~ PERW SEARCHING UTK DATA TAMU
4. DI FORM CETAK HAS ADA TEL NARJE, PERWA JUMLAH HARI.

Malang, 24.8. 2006

()



PERMOHONAN PERSETUJUAN SKRIPSI

Yang betanda tangan dibawah ini :

Nama : ANDRIAS KRISTANTO W.
NIM : 0512526
Semester : IX (SEMBILAN)
Fakultas : Teknologi Industri
Jurusan : Teknik Elektro S-1
Konsentrasi : **TEKNIK ELEKTRONIKA**
TEKNIK ENERGI LISTRIK
TEKNIK KOMPUTER DAN INFORMATIKA
Alamat : Jl. PERUSAHAAN GG. D NO. 88A LOSAWI - SINGOSARI

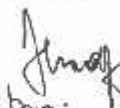
Dengan ini kami mengajukan permohonan untuk mendapatkan persetujuan untuk membuat **SKRIPSI Tingkat Sarjana**. Untuk melengkapi permohonan tersebut, bersama kami lampirkan persyaratan-persyaratan yang harus dipenuhi.

Adapun persyaratan-persyaratan pengambilan **SKRIPSI** adalah sebagai berikut :

1. Telah melaksanakan semua praktikum sesuai dengan konsentrasinya (.....)
2. Telah lulus dan menyerahkan Laporan Praktek Kerja (.....)
3. Telah lulus seluruh mata kuliah keahlian (MKB) sesuai konsentrasinya (.....)
4. Telah menempuh mata kuliah ≥ 134 sks dengan IPK ≥ 2 dan tidak ada nilai E (.....)
5. Telah mengikuti secara aktif kegiatan seminar skripsi yang diadakan Jurusan (.....)
6. Memenuhi persyaratan administrasi (.....)

Demikian permohonan ini untuk mendapatkan penyelesaian lebih lanjut dan atas perhatiannya kami ucapkan terima kasih.

Telah diteliti kebenaran data tersebut diatas
Recording Teknik Elektro


(.....)

Malang, ..NOVEMBER....2009

Pemohon


(ANDRIAS KRISTANTO W.)

Disetujui
Ketua Jurusan Teknik Elektro


Ir. F. Yudi Limpraptono, MT
NIP. P. 1039500274

Mengetahui
Dosen Wali


(.....)

Catatan :

Bagi mahasiswa yang telah memenuhi persyaratan mengambil SKRIPSI agar membuat proposal dan mendapat persetujuan dari Ketua Jurusan/Sekretaris Jurusan T. Elektro S-1

1. 100 437 / 138 - 3.17
2.
3.

Lampiran : 1 (satu) berkas
Pembimbing Skripsi

Kepada : Yth. Ibu Ir. TH. Mimien Mustikawati, MT
Dosen Institut Teknologi Nasional
MALANG

Yang bertanda tangan di bawah ini .

Nama : ANDRIAS KRISTANTO WIDIGDO
Nim : 0512526
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Komputer dan Informatika

Dengan ini mengajukan permohonan kiranya Ibu bersedia menjadi Dosen Pembimbing Utama / Pendamping *), untuk penyusunan Skripsi dengan judul (proposai terlampir) :

**PERANCANGAN DAN PEMBUATAN SISTEM INFORMASI
ADMINISTRASI PERHOTELAN PADA HOTEL VICTORY
DI BATU MALANG BERBASIS CLIENT SERVER**

Adapun tugas tersebut sebagai salah satu syarat untuk merempuh Ujian Akhir Sarjana Teknik.

Demikian permohonan kami dan atas kesediaan Ibu kami ucapkan terima kasih.

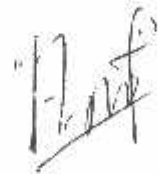
Malang, 7 Desember 2009

Mengetahui
Ketua Jurusan Teknik Elektro



Ir. F. Yudi Limpraptono, MT
NIP.Y. 1039 500274

Hormat kami



Andrias Kristanto Widigdo

*)coret yang tidak perlu

Form S-3 a

INSTITUT TEKNOLOGI NASIONAL
Jl. Raya Karanglo Km 2
MALANG

Lampiran : 1 (satu) berkas
Pembimbing Skripsi

Kepada : Yth. Ibu Febriana Santi W, S.Kom, M.Kom
Dosen Institut Teknologi Nasional
MALANG

Yang bertanda tangan di bawah ini :

Nama : ANDRIAS KRISTANTO WIDIGDO
Nim : 0512526
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Komputer dan Informatika

Dengan ini mengajukan permohonan kiranya Ibu bersedia menjadi Dosen Pembimbing ~~Utama~~ / Pendamping *), untuk penyusunan Skripsi dengan judul (proposal terlampir) :

**PERANCANGAN DAN PEMBUATAN SISTEM INFORMASI
ADMINISTRASI PERHOTELAN PADA HOTEL VICTORY
DI BATU MALANG BERBASIS CLIENT SERVER**

Adapun tugas tersebut sebagai salah satu syarat untuk menempuh Ujian Akhir Sarjana Teknik.

Demikian permohonan kami dan atas kesediaan Ibu kami ucapkan terima kasih.

Malang, 7 Desember 2009

Mengetahui
Ketua Jurusan Teknik Elektro



Ir. F. Yudi Limpraptono, MT
NIP.Y. 1039 500274

Hormat kami



Andrias Kristanto Widigdo

*)coret yang tidak perlu

Form S-3 a

PERNYATAAN KESEDIAAN DALAM PEMBIMBINGAN SKRIPSI

Sesuai permohonan dari mahasiswa :

Nama : ANDRIAS KRISTANTO WIDIGDO
Nim : 0512526
Semester : 9 (Sembilan)
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Komputer dan Informatika

Dengan ini Menyatakan bersedia / tidak bersedia*) Membimbing Skripsi dari mahasiswa tersebut , dengan judul:

**PERANCANGAN DAN PEMBUATAN SISTEM INFORMASI
ADMINISTRASI PERHOTELAN PADA HOTEL VICTORY
DI BATU MALANG BERBASIS CLIENT SERVER**

Demikian surat Pernyataan ini kami buat agar dapat dipergunakan seperlunya.

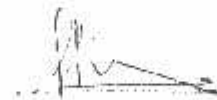
Malang, 7 Desember 2009

Kami yang membuat pernyataan,

Catatan :

Setelah disetujui agar formulir ini
Diserahkan mahasiswa/i yang bersangkutan
Kepada Jurusan untuk diproses lebih lanjut.

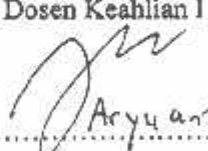
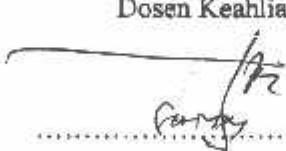
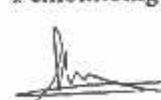
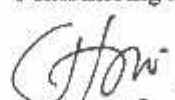
*)coret yang tidak perlu



Ir. TH. Mimien Mustikawati, MT
NIP.P. 1039 000352



BERITA ACARA SEMINAR PROPOSAL SKRIPSI
PROGRAM STUDI TEKNIK ELEKTRO S1

KONSENTRASI		TEKNIK INFORMATIKA DAN KOMPUTER			
1.	Nama Mahasiswa	ADRIAS KRISTANTO W		NIM	0512526
2.	Keterangan	Tanggal		Waktu	Tempat / Ruang
	Pelaksanaan	19 DESEMBER 2009			
3.	Spesifikasi Judul (berilah tanda silang *)				
	a.	Sistem Tenaga Elektrik	e.	Embbded System	☒ i. Sistem Informasi
	b.	Konversi Energi	f.	Antar Muka	j. Jaringan Komputer
	c.	Sistem Kendali	g.	Elektronika Telekomunikasi	k. Web
	d.	Tegangan Tinggi	h.	Elektronika Instrumer'tasi	l. Algoritma Cerdas
4.	Judul Proposal yang diseminarkan Mahasiswa	PERANCANGAN DAN PEMBUATAN SISTEM INFORMASI ADMINISTRASI PERHOTELAN PADA HOTEL VICTORY DI BATU MALANG BERBASIS CLIENT SERVER			
5.	Perubahan Judul yang diusulkan oleh Kelompok Dosen Keahlian				
6.	Catatan :				
7.	Catatan :				
	Persetujuan Judul Skripsi				
	Disetujui, Dosen Keahlian I		Disetujui, Dosen Keahlian II		
	 Aryanto				
	Mengetahui, Ketua Jurusan		Disetujui, Calon Dosen Pembimbing		
 Ir. F. Yudi Limpraptono, MT NIP. Y. 1039500274		Pembimbing I		Pembimbing II	
		 M. Mubandari		 Febriana S.W	

Keterangan :

*) dilingkari a, b, c, ... sesuai dengan bidang keahlian



FORMULIR BIMBINGAN SKRIPSI

Nama : ANDRIAS KRISTANTO WIDIGDO
NIM : 05.12.526
Masa Bimbingan : 14 Juni 2010 s/d 14 Desember 2010
Judul Skripsi : PERANCANGAN DAN PEMBUATAN SISTEM INFORMASI ADMINISTRASI PERHOTELAN PADA HOTEL VICTORY DI BATU MALANG BERBASIS CLIENT SERVER

No	Tanggal	Uraian	Paraf Pembimbing
1	19 Juli 2010	Revisi tampilan dan warna pada aplikasi	
2	26 Juli 2010	Bab I dan Bab II, Perbaiki Latar Belakang Dan Batasan Masalah	
3	30 Juli 2010	Bab I dan Bab II Acc, Bab III revisi desain sistem	
4	2 Agustus 2010	Perbaiki tata letak tombol dan komponen pada program	
5	3 Agustus 2010	Bab III Acc, revisi print screen gambar Bab IV	
6	7 Agustus 2010	Revisi tentang perbandingan program	
7	10 Agustus 2010	Bab IV Acc, Bab V revisi Kesimpulan dan Saran	
8	12 Agustus 2010	Bab V Acc, Seminar Hasil	
9			
10			

Malang,
Dosen Pembimbing

(Ir. Yusuf Ismail Nakhoda, MT)
NIP.Y. 1018800189

FORM S-4b



FORMULIR BIMBINGAN SKRIPSI

Nama : ANDRIAS KRISTANTO WIDIGDO
NIM : 05.12.526
Masa Bimbingan : 14 Desember 2009 s/d 14 Juni 2010
Judul Skripsi : PERANCANGAN DAN PEMBUATAN SISTEM INFORMASI
ADMINISTRASI PERHOTELAN PADA HOTEL VICTORY DI BATU
MALANG BERBASIS CLIENT SERVER

No	Tanggal	Uraian	Paraf Pembimbing
1	28 - 4 - 10	Perbaiki Bab I (+ ERP) : ACC Bab I	
2			
3			
4			
5			
6			
7			
8			
9			
10			

Malang,
Dosen Pembimbing

(Febriana Santi W, S.Kom, M.Kom)



FORMULIR BIMBINGAN SKRIPSI

Nama : ANDRIAS KRISTANTO WIDIGDO
NIM : 05.12.526
Masa Bimbingan : 14 Juni 2010 s/d 14 Desember 2010
Judul Skripsi : PERANCANGAN DAN PEMBUATAN SISTEM INFORMASI ADMINISTRASI PERHOTELAN PADA HOTEL VICTORY DI BATU MALANG BERBASIS CLIENT SERVER

No	Tanggal	Uraian	Paraf Pembimbing
1	6-8-2010	Perbaiki DFD, ACC Bab III + Bab IV	
2			
3	9-8-2010	ACC Bab V, Lanjutkan Daftar Pustaka	
4	12-8-2010	ACC Seminar Hasil	
5			
6			
7			
8			
9			
10			

Malang,
Dosen Pembimbing

(Febriana Santi W, S.Kom, M.Kom)

```

{
Form Utama Aplikasi
}
//
Perancangan dan Pembuatan Sistem Informasi Administrasi Perhotelan pada Hotel Victory
di Batu-Malang Berbasis Client-Server
Dibuat pada : Agustus 2010
Oleh : christianh_newbie@yahoo.co.id
}
//
unit Utama;

interface

uses
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
Dialogs, StdCtrls, ComCtrls, HxCtrls, Menus, DB, ADODB, jpeg;

type
TFormUtama = class(TForm)
Image1: TImage;
panelKasir: TPanel;
MonthCalendar1: TMonthCalendar;
GroupBox1: TGroupBox;
Label1: TLabel;
Label2: TLabel;
btn_login: TButton;
txt_password: TEdit;
txt_user: TEdit;
btn_reset: TButton;
MainMenu1: TMainMenu;
MainMenu2: TMainMenu;
main_reg: TMenuItem;
Pelaporan1: TMenuItem;
LapDataUmum1: TMenuItem;
LapPencarianKamar1: TMenuItem;
LapPendafataran1: TMenuItem;
LapRestoran1: TMenuItem;
LapPemasukan1: TMenuItem;
porHari1: TMenuItem;
perBulan1: TMenuItem;
prTahun1: TMenuItem;
LapRugiLabat1: TMenuItem;
Akses1: TMenuItem;
EntryDataKamar1: TMenuItem;
EntrydataKrywn: TMenuItem;
Logout1: TMenuItem;
Keluar1: TMenuItem;
main_resto: TMenuItem;
main_transaksi: TMenuItem;

```

```

main_biayaFas: TMenuItem;
main_biayaTelp: TMenuItem;
main_Tagihan: TMenuItem;
Lbl_Status: TLabel;
ADOQ_user: TADOQuery;
ADOQ_usutrik: TStringField;
ADOQ_username: TStringField;
ADOQ_userpassword: TStringField;
ADOQ_userrole_ID: TStringField;
Image2: TImage;
ADOConnection_utama: TADOConnection;
procedure main_regClick(Sender: TObject);
procedure Keluar1Click(Sender: TObject);
procedure btn_resetClick(Sender: TObject);
procedure main_restoClick(Sender: TObject);
procedure main_gudangClick(Sender: TObject);
procedure main_ghn_tamuClick(Sender: TObject);
procedure EntryDataKrywnClick(Sender: TObject);
procedure EntrydataKrywnClick(Sender: TObject);
procedure main_biayaFasClick(Sender: TObject);
procedure main_biayaTelpClick(Sender: TObject);
procedure main_TagihanClick(Sender: TObject);
procedure LapRugiLabat1Click(Sender: TObject);
procedure LapPencarianKamar1Click(Sender: TObject);
procedure btn_loginClick(Sender: TObject);
procedure FormCreset(Sender: TObject);
procedure Logout1Click(Sender: TObject);
procedure LapPendafataran1Click(Sender: TObject);
procedure LapDataTamu1Click(Sender: TObject);
procedure LapRestoran1Click(Sender: TObject);
private
{ Private declarations }
public
procedure bersih;
procedure aksesFormfalse;

procedure aksesFormResto;
procedure aksesHouseKeeping;
procedure aksesFormAdmin;
procedure aksesFormOwner;
procedure aksesFormReception;

{ Public declarations }
end;

var
FormUtama: TFormUtama;

implementation

```

```

uses Reg, Resto, Gudang, TagihanTamu, EntryDataKamar, EntryDataKaryawan, BiayaFas, BiayaTelp,
RugiTela, If_frm_RPT_DataTamu, U_frm_rpt_reservasi, U_frm_rpt_Ditriamu, U_frm_rpt_resto,
{SR * .dfm}

```

```

procedure TFormUtama.main_regClick(Sender: TObject);
begin
    frm_Registrasi.Show;
end;

```

```

procedure TFormUtama.KeluarClick(Sender: TObject);
begin
    Close;
end;

```

```

procedure TFormUtama.btn_resetClick(Sender: TObject);
begin
    bersih;
end;

```

```

procedure TFormUtama.bersih;
begin
    txt_user.Text := '';
    txt_password.Text := '';
end;

```

```

procedure TFormUtama.main_gudangClick(Sender: TObject);
begin
    frm_gudang.Show;
end;

```

```

procedure TFormUtama.main_tglu_tamuClick(Sender: TObject);
begin
    frm_tagihanTamu.Show;
end;

```

```

procedure TFormUtama.main_restoClick(Sender: TObject);
begin
    frm_restaurant.Show;
end;

```

```

procedure EntryDataKamarClick(Sender: TObject);
begin
    frm_entryDataKamar.Show;
end;

```

```

procedure EntrydataKaryawanClick(Sender: TObject);

```

```

begin
    frm_entryDataKaryawan.Show;
end;

```

```

procedure TFormUtama.aksesformfalse;
begin
    main_reg.Enabled:=False;
    main_resto.Enabled:=False;
    main_biayaFas.Enabled:=False;
    main_biayaTelp.Enabled:=False;
    main_Tagihan.Enabled:=False;
    EntryDataKamar.Enabled:=False;
    EntrydataKaryawan.Enabled:=False;
    LapDataTamu.Enabled:=False;
    LapPemesananKamar.Enabled:=False;
    LapPendatutamu.Enabled:=False;
    LapRestauran.Enabled:=False;
    LapRugiTela.Enabled:=False;
    LapPemasukan.Enabled:=False;
end;

```

```

procedure TFormUtama.EntryDataKamarClick(Sender: TObject);
begin
    frm_entryDataKamar.Show;
end;

```

```

procedure TFormUtama.EntrydataKaryawanClick(Sender: TObject);
begin
    frm_entryDataKaryawan.Show;
end;

```

```

procedure TFormUtama.main_biayaFasClick(Sender: TObject);
begin
    biaya_fasilitas.Show;
end;

```

```

procedure TFormUtama.main_biayaTelpClick(Sender: TObject);
begin
    biaya_telp.Show;
end;

```

```

procedure TFormUtama.main_TagihanClick(Sender: TObject);
begin
    frm_tagihanTamu.Show;
end;

```

```

procedure TFormUtama.LapRugiTelaClick(Sender: TObject);

```

```

begin
    frm_rugi1.aht.Show;
end;

procedure TForm1 Utama.LapPemesananKamar1.Click(Sender: TObject);
begin
    frm_rpt_reserv_QR_rpt_reserv.Preview;
end;

procedure TForm1 Utama.btm_login.Click(Sender: TObject);
var
    s,s1:s String;
begin
    if (txt_user.Text="" or (txt_password.Text="")) then begin
        MessageDlg('Cek Username dan Password Anda',mtConfirmation,[mbOK],0);
    end else begin
        s:='SELECT * FROM T_user WHERE username like :pID AND password like :pWD';
        with ADOQ_user do begin
            SQL.Clear;
            SQL.Add(s);
            Parameters.ParamValues['pID']:='%'+txt_user.Text+'%';
            Parameters.ParamValues['pWD']:='%'+txt_password.Text+'%';
            ExecSQL;
            Close;
            Open;
            end;
        if (txt_user.Text='administrator' and (txt_password.Text='1234')) then begin
            FormUtama.aksesFormAdmin;
            txt_password.Text:='';
            txt_user.Text:='';
        end else
        if (ADOQ_user.RecordCount=1) then begin
            MessageDlg('User dan Password ditemukan!!!',mtInformation,[mbOK],0);
            //frm_loading.Show;
            //frm_loading.Timer1.Enabled:=True;
            //frm_Login.Hide;

            //Transfer data ke dalam form utama
            Lbl_Status.Caption:='STATUS ' + ADOQ_usr.username.Value + ' ' +
            ADOQ_usrrole_ID.Value;

            //Pantangan Status Login
            sts:=ADOQ_usrrole_ID.Value;
            bersih;

            if (sts='Owner') then

```

```

        aksesFormOwner
    else if (sts='Admin') then
        aksesFormAdmin
    else if (sts='Receptionist') then
        aksesFormReception
    else if (sts='Restaurant') then
        aksesFormResto
    else if (sts='House Keeping') then
        aksesHouseKeeping
    else
        aksesFormHlse
    end else begin
        MessageDlg('Data Tidak Ditemukan!!!',mtError,[mbOK],0);
        bersih;
    end;
end;
end;

procedure TForm1 Utama.aksesFormResto;
begin
    main_reg.Enabled:=False;
    main_resto.Enabled:=True;
    main_biayaFas.Enabled:=False;
    main_biayaTelp.Enabled:=False;
    main_tugasan.Enabled:=False;
    EntryDataKaryawan.Enabled:=False;
    EntryDataKamar1.Enabled:=False;
    LapDataTamu1.Enabled:=True;
    LapPemesananKamar1.Enabled:=True;
    LapPendaftaran1.Enabled:=True;
    LapRestaurant1.Enabled:=True;
    LapRugiLabo1.Enabled:=True;
    LapPemasukan1.Enabled:=True;
end;

procedure TForm1 Utama.aksesHouseKeeping;
begin
    main_reg.Enabled:=False;
    main_resto.Enabled:=False;
    main_biayaFas.Enabled:=True;
    main_biayaTelp.Enabled:=True;
    main_tugasan.Enabled:=False;
    EntryDataKamar1.Enabled:=False;
    EntryDataKaryawan.Enabled:=False;
    LapDataTamu1.Enabled:=True;
    LapPemesananKamar1.Enabled:=True;

```

```

LapPendafarian1.Enabled:=True;
LapRestoran1.Enabled:=True;
LapRugiLaba1.Enabled:=True;
LapPemasukan1.Enabled:=True;
end;

procedure TFormUtama aksesFormAdmin;
begin
    main_reg.Enabled:=True;
    main_resto.Enabled:=True;
    main_biayaFas.Enabled:=True;
    main_biayaTelp.Enabled:=True;
    main_tagihan.Enabled:=True;
    EntryDataKamar1.Enabled:=True;
    EntrydataKrywn.Enabled:=True;
    LapDataTamu1.Enabled:=True;
    LapPencapaianKamar1.Enabled:=True;
    LapPendafarian1.Enabled:=True;
    LapRestoran1.Enabled:=True;
    LapRugiLaba1.Enabled:=True;
    LapPemasukan1.Enabled:=True;
end;

procedure TFormUtama aksesFormOwner;
begin
    main_reg.Enabled:=True;
    main_resto.Enabled:=True;
    main_biayaFas.Enabled:=True;
    main_biayaTelp.Enabled:=True;
    main_tagihan.Enabled:=True;
    EntryDataKamar1.Enabled:=True;
    EntrydataKrywn.Enabled:=True;
    LapDataTamu1.Enabled:=True;
    LapPencapaianKamar1.Enabled:=True;
    LapPendafarian1.Enabled:=True;
    LapRestoran1.Enabled:=True;
    LapRugiLaba1.Enabled:=True;
    LapPemasukan1.Enabled:=True;
end;

procedure TFormUtama aksesFormReception;
begin
    main_reg.Enabled:=True;
    main_resto.Enabled:=False;
    main_biayaFas.Enabled:=False;
    main_biayaTelp.Enabled:=False;
    main_tagihan.Enabled:=False;
    EntryDataKamar1.Enabled:=True;
    EntrydataKrywn.Enabled:=False;

```

```

LapDataTamu1.Enabled:=True;
LapPencapaianKamar1.Enabled:=True;
LapPendafarian1.Enabled:=True;
LapRestoran1.Enabled:=True;
LapRugiLaba1.Enabled:=True;
LapPemasukan1.Enabled:=True;
end;

procedure TFormUtama FormCreate(Sender: TObject);
begin
    MonthCalendar1.Date:=Now;
    aksesFormfalse;
end;

procedure TFormUtama Logout1Click(Sender: TObject);
begin
    aksesFormfalse;
    MessageDlg('Anda telah Logout dari Aplikasi ini' mtConfirmation,mbOK,0);
    Refresh;
end;

procedure TFormUtama LapPendafarian1Click(Sender: TObject);
begin
    frm_rpt_DataTamu_QR_rpt_DTamuPreview;
end;

procedure TFormUtama LapDataTamu1Click(Sender: TObject);
begin
    frm_rpt_Dftr_tamu_QR_rpt_DTamuPreview;
end;

procedure TFormUtama LapRestoran1Click(Sender: TObject);
begin
    frm_qr_resto_QR_rpt_restoPreview;
end;

end.
{
//-----
}
Form Registrasi
//-----
}

procedure TForm_Registrasi simpanClick(Sender: TObject);
var
    cek:Boolean;
begin

```

```

if (no_daftar.Text="" or (nama.Text="" or (alamat.Text="" or (no_ID.Text="" then
begin
    MessageDlg('Maaf Data Masih Ada yang Kosong, Harap Di isi Terlebih  

    Dahulu', mtConfirmation, [mbOK], 0);
end
else
begin
    if (edi=False) then begin
        cek:=False;
        if (not cek=cekID(no_daftar.Text)) then begin
            ShowMessage('No Pemesanan Sudah Ada');
            Exit;
        end;
        ADOdataTamu.Append;
        ADOdetail.Append;
    end else
        ADOdataTamu.Edit;
        ADOdetail.Edit;

    with ADOdataTamu do begin
        FieldByName('no_pesanan').Value := Trim(no_pesanan.Text);
        FieldByName('no_Pdaftar').Value := Trim(no_daftar.Text);
        FieldByName('nama').Value := Trim(nama.Text);
        FieldByName('alamat').Value := Trim(alamat.Text);
        FieldByName('tempat_lahir').Value := Trim(tempat_lahir.Text);
        FieldByName('tgl_lahir').Value := (tgl_lahir.Date);

    if RadioLaki.Checked = True then
        begin
            FieldByName('jns_kelamin').Value := 'L';
        end
        else
        begin
            FieldByName('jns_kelamin').Value := 'P';
        end;

        FieldByName('ID').Value := Trim(ID_cand.Text);
        FieldByName('no_ID').Value := Trim(no_ID.Text);
        FieldByName('room').Value := Trim(no_kamar.Text);
        FieldByName('cek_in').Value := (cek_in.DateTime);
        FieldByName('cek_out').Value := (cek_out.DateTime);
        FieldByName('pugas_penerima').Value := Trim(pgs_penerima.Text);
        Post;
    end;

    with ADOdetail do begin
        FieldByName('no_Pdaftar').Value := Trim(no_daftar.Text);

```

```

        FieldByName('tgl_update').Value:=FormatDateTime('dd/mm/yyyy', Date);
        FieldByName('deposit').Value := Trim(deposit.Text);
        FieldByName('tertutang').Value := Trim(tertutang.Text);
        FieldByName('kembali').Value := Trim(kembali.Text);
        FieldByName('total_akhir_bayar').Value := Trim(total_bayar.Text);
        Post;
    end;

    //syntax menghapus data Reserv
    //if ADOreserv.RecordCount > 0 then
    if stmpian.Caption='SIMPAN' then
        ADOreserv.Delete;
        //ADOreserv.Requery();

    Refresh;
    ShowMessage('Data Berhasil tersimpan!!');
    bensih;
    edi:=False;
    KamarKosong;
    KamarPesana;
    KamarTersewa;
end;
end;

procedure Tfrm_Registrasi.keluarClick(Sender: TObject);
begin
    Close;
end;

procedure Tfrm_Registrasi.FormCreate(Sender: TObject);
begin
    ADOdataTamu.Active:=True;
    ADOreserv.Active:=True;
    ADOQ_TypeK.Active:=True;
    ADOdetail.Active:=True;
    edi:=False;
    cek_in.Date:=Now;
    cek_out.Date:=Now;
    Res_tglPsn.Date:=Now;
    Res_cekIn.Date:=Now;
    ADOreserv.Requery();

    KamarKosong;
    KamarTersewa;
    KamarPesana;
end;

function Tfrm_Registrasi.cekID(ID: string): boolean;

```

```

begin
  if(Not ADOdataTamu Locate('no_pdaftar',ID,[!ofPartialKey,loc'ascinsensitive!])) then
    Result:=False
  else
    Result:=True;
  end;
end;

procedure Tfrm_Registrasi.DBGrid2CellClick(Column: TColumn);
begin
  edt:=True;
  with ADOreserv do begin
    no_pesun.Text := FieldByName('no_pesun').Value;
    nama.Text := Trim(FieldValues['nama']);
    alamat.Text := Trim(FieldValues['alamat']);
    no_kamar.Text := Trim(FieldValues['room']);
    cek_in.Date := (FieldValues['cek_in']);
    deposit.Text := IntToStr(FieldValues['deposit']);
  end;
  simpan.Caption:=SIMPAN_;
end;
edt:=False;
end;

procedure Tfrm_Registrasi.DBGrid1CellClick(Column: TColumn);
var
  jnsKlrm :String;
begin
  edt:=True;
  With ADOdataTamu do begin
    no_pesun.Text := Trim(FieldValues['no_pesun']);
    no_dafar.Text := Trim(FieldValues['no_pdaftar']);
    nama.Text := Trim(FieldValues['nama']);
    alamat.Text := Trim(FieldValues['alamat']);
    tempat_lahir.Text:= Trim(FieldValues['tempat_lahir']);
    tg_lahir.Date := (FieldValues['tg_lahir']);
  end;
  jnsKlrm:= Trim(FieldValues['jns_kelamin']);
  if jnsKlrm = '1' then
    RadioLaki.Checked:=True
  else
    RadioPerempuan.Checked:=True;
  ID_card.Text := Trim(FieldValues['ID']);
  no_ID.Text := Trim(FieldValues['no_ID']);
  no_kamar.Text := Trim(FieldValues['room']);
  cek_in.Date:= FieldValues['cek_in'];
  cek_out.DateTime:= FieldValues['cek_out'];
  pgs_penerima.Text:= Trim(FieldValues['pgas_penerima']);

```

```

end;

//cari pada T detailBarang
if(length(no_dafar.Text)>0)then begin
  ADOdetail.Locate('no_pdaftar',no_dafar.Text,[!ofPartialKey,loc'CaseInsensitive!]);
  refresh;
  deposit.Text := ADOdetail.FieldValues['deposit'];
  terhitung.Text := ADOdetail.FieldValues['terhitung'];
  kembali.Text := ADOdetail.FieldValues['kembali'];
  total_bayar.Text:= ADOdetail.FieldValues['total_akhir_bayar'];
end;

end;

procedure Tfrm_Registrasi.bersih;
begin
  //membersihkan Form Registrasi
  no_pesun.Text:= ''; no_dafar.Text:= '';
  nama.Text:= ''; alamat.Text:= '';
  tempat_lahir.Text:= ''; no_ID.Text:= '';
  no_kamar.Text:= ''; pgs_penerima.Text:= '';
  Tipe_kamar.Text:= ''; INI_bayar.Text:=0;
  terhitung.Text :=0;kembali.Text:=0;
  total_bayar.Text:=0;deposit.Text:=0;

  //membersihkan Form Reservation
  Res_noPesun.Text:= '';Res_nama.Text:= '';
  Res_alamat.Text:= '';Res_room.Text:= '';
  //Res_tipeKmr.Text:= '';Res_jmlhKmr.Text:= '';
  Resv_Ilarga.Text:=0;

  lbl_deposit.Caption :=0;
  lbl_terhitung.Caption :=0;
  lbl_getByr.Caption :=0;

end;

procedure Tfrm_Registrasi.lambuhClick(Sender: TObject);
begin
  bersih;
  edt:=False;
  ADOreserv.Requery;
  ADOdetail.Requery;
  ADOdataTamu.Requery;

  no_dafar.Text:=FormatDateTime('yyyy-mm-dd hh:mm:ss',now);
  //KamarKosong
  simpan.Caption:=SIMPAN;
end;

```

```

procedure Tfrm_Registrasi.no_kamar_kamar, indX FROM v_TipeKamar where
var
s:String;
begin
s:='SELECT no_kamar,tipe_kamar,bod_size,harga_kamar,indX FROM v_TipeKamar where
no_kamar LIKE 'pNO';
with ADOQ_TipeK as do begin
sql.Clear;
sql.Add(s);
Parameters.ParamValues['pNO']:='%'+no_kamar.Text+'%';
ExecSQL;
Close;
Open;
end;
if ADOQ_TipeK.RecordCount>0 then begin
txt_harga.Text:=ADOQ_TipeK.FieldValues[harga_kamar];
Tipe_kamar.Text:=ADOQ_TipeK.FieldValues[tipe_kamar];
end;
HitungHari;
end;

procedure Tfrm_Registrasi.btn_tghantamuClick(Sender: TObject);
begin
if no_daftar.Text='' then begin
MessageDlg('Maaf No Pelanggan Tidak Tersedia',mInformation,[mbOK],0);
Exit;
end
else begin
frm_tagihanTamu.Show;
frm_tagihanTamu.no_daftar.Text:=no_daftar.Text;
frm_tagihanTamu.no_bayar.Text:=FormatDateTime('yyyy-mm-dd hh:mm:ss',now);
end;
end;

{var
X,I: Integer;
begin
for I:=StrToInt(deposit.Text) downto StrToInt(terhutang.Text) do begin
if (Components[0].ClassType=TPanel) then
(Components[0] as TPanel).Color:=clAqua;
end;}
end;

procedure Tfrm_Registrasi.HitungHari;
var
s,s2:TDate;
T,H1,H2: Word;

```

```

Hari:Integer;
Dep, Ter, Kemb, TolbYr: integer;
begin
s:=cock_in.Date;
s2:=cock_out.Date;
DecoDelDateDay(s,T,H1);
DecoDelDateDay(s2,T,H2);
Hari:=H2-H1;
if Hari=0 then
total_bayar.Text:=IntToStr(((H2-H1)+1)*StrToInt(txt_harga.Text))
else
total_bayar.Text:=IntToStr(((H2-H1)*StrToInt(txt_harga.Text));
TolbYr:=StrToInt(total_bayar.Text);
Dep:=StrToInt(deposit.Text);
Kemb:=0;
Ter:=0;
lbl_deposit.Caption:=FormatDateTime('###0.00',StrToInt(deposit.Text));
lbl_terhutang.Caption:=FormatDateTime('###0.00',StrToInt(terhutang.Text));
lbl_totByr.Caption:=FormatDateTime('###0.00',StrToInt(total_bayar.Text));
Ter:=TolbYr-Dep;
if Ter>0 then begin
terhutang.Text:=IntToStr(Ter);
kembali.Text:=IntToStr(0);
end
else begin
Kemb:=Dep-TolbYr;
kembali.Text:=IntToStr(Kemb);
terhutang.Text:=IntToStr(0);
end;
end;

procedure Tfrm_Registrasi.cock_inChange(Sender: TObject);
begin
HitungHari;
end;

procedure Tfrm_Registrasi.cock_outChange(Sender: TObject);
begin
HitungHari;
end;

```

```

procedure TForm_Registrasi.FormActivate(Sender: TObject);
begin
  ADOdataTamu.Query;
  ADOreserv.Query;
  ADOQ_TipeK.Query;
  ADOdetail.Query;

```

```

end;

```

```

procedure TForm_Registrasi.btn_Simpan_resvClick(Sender: TObject);

```

```

var
  cek:Boolean;
begin
  if (Res_noPsn.Text='') or (Res_nama.Text='') or (Res_alamat.Text='') or (Res_room.Text='') then
  begin
    MessageDlg('Maaf Data Masih Ada yang Kosong, Harap Di isi Terlebih  
Dahulu', mtConfirmation, [mbOK], 0);
  end
  else
  begin
    if (cek=False) then begin
      cek:=False;
      if (not cek=oldNoPesunResv(Res_noPsn.Text)) then begin
        ShowMessage('No Pemesanan Sudah Ada');
        Exit;
      end;
      ADOreserv.Append;
    end else
      ADOreserv.Edit;
  end

```

```

  with ADOreserv do begin
    FieldByName('no_pesun').Value := Trim(Res_noPsn.Text);
    FieldByName('nama').Value := Trim(Res_nama.Text);
    FieldByName('alamat').Value := Trim(Res_alamat.Text);
    FieldByName('room').Value := Trim(Res_room.Text);
    FieldByName('tgl_pesun').Value := (Res_tglPsn.Date) Time;
    FieldByName('cek_in').Value := (Res_cekIn.Date) Time;
    FieldByName('depoexit').Value := (depoexit.Text);
    FieldByName('tgl_update').Value := Now;
    Post;
  end;

```

```

  Refresh;
  ShowMessage('Data Berhasil tersimpan!!!');
  bersih;
  cek:=False;
end;

```

```

KamarKosong;
KamarPesun;
KamarTersewa;

```

```

end;

```

```

function TForm_Registrasi.cekNoPesunResv(No: string) boolean;
begin
  if (Not ADOreserv.Locate('no_pesun', No, [ofPartialKey, ofCaseInsensitive])) then
    Result:=False
  else
    Result:=True;
  end;

```

```

procedure TForm_Registrasi.Tab_ReservasiShow(Sender: TObject);
begin
  tertututug.Enabled := False;
  kembeli.Enabled := False;
  total_bayar.Enabled := False;
  btn_tghnTamu.Enabled := False;
end;

```

```

procedure TForm_Registrasi.Res_roomChange(Sender: TObject);
var
  s:String;
begin
  s:=SELECT 'no_kamar tipe_kamar, bed_size, harga_kamar, idx FROM v_TipeKamar where  
no_kamar LIKE 'pNO';
  with ADOQ_TipeK do begin
    sql.Clear;
    sql.Add(s);
    Parameters.ParamValues['pNO']:=%+ Res_room.Text +'%';
    ExecSQL;
    Close;
    Open;
  end;
  if ADOQ_TipeK.RecordCount>0 then begin
    resv_Harga.Text := ADOQ_TipeK.FieldValues['harga_kamar'];
    Res_tipeKmr.Text := ADOQ_TipeK.FieldValues['tipe_kamar'];
  end;
end;

```

```

procedure TForm_Registrasi.Tab_RegistrasiShow(Sender: TObject);
begin
  tertututug.Enabled := True;
  kembeli.Enabled := True;
  total_bayar.Enabled := True;
  btn_tghnTamu.Enabled := True;
end;

```

```

procedure Tfrm_Registrasi.Cntd_DataReservasiClick(Column: TColumn);
begin
  edit:=True;
  With ADOreserv do begin
    Res_noPsn.Text:=FieldValues[no_pesan];
    Res_nama.Text:=FieldValues[nama];
    Res_alamat.Text:=FieldValues[alamat];
    Res_room.Text:=FieldValues[room];
    Res_cockInDate:=FieldValues[cock_in];
    Res_tglTsa.Date:=FieldValues[tgl_pesan];
    deposit.Text:=FieldValues[deposit];
  end;
  txt_CanData.Text:=FieldValues[nama];
end;

procedure Tfrm_Registrasi.txt_CanDataChange(Sender: TObject);
begin
  if(length(txt_CanData.Text)>0)then begin
    ADOreserv Locate('nama',txt_CanData.Text,[loPartialKey,loCaseInsensitive]);
    refresh;
  end;
end;

procedure Tfrm_Registrasi.btn_Hapus_resvClick(Sender: TObject);
begin
  if(length(txt_CanData.Text)>0)then begin
    ADOreserv Locate('nama',txt_CanData.Text,[loPartialKey,loCaseInsensitive]);
    if(Application.MessageBox('Anda Yakin menghapus record ini?', 'Peringatan', MB_YESNO or
      MB_ICONQUESTION)=IDYES)then begin
      ADOreserv Delete;
      edit:=false;
    end;
    refresh;
  end;
  edit:=false;
  refresh;
end;

procedure Tfrm_Registrasi.btn_Bersih_resvClick(Sender: TObject);
begin
  bersih;
  edit:=false;
end;

procedure Tfrm_Registrasi.btn_hapus_regClick(Sender: TObject);
begin
  edit:=false;

```

```

  if(edit=False) then begin
    ADO_cockout.Append;
  end;
  with ADO_cockout do begin
    FieldByName('no_Pdaftar').Value := Trim(no_daftar.Text);
    FieldByName('nama_lengkap').Value := Trim(nama.Text);
    FieldByName('alamat').Value := Trim(alamat.Text);
    FieldByName('tempat_lahir').Value := Trim(tempat_lahir.Text);
    FieldByName('tgl_lahir').Value := FormatDateTime('dd/mm/yyyy', (tgl_lahir.Date));
  end;
  if Radio1.Checked = True then
  begin
    FieldByName('jns_kelamin').Value = 'L';
  end
  else
  begin
    FieldByName('jns_kelamin').Value = 'P';
  end;
  FieldByName('ID').Value := Trim(ID_card.Text);
  FieldByName('no_ID').Value := Trim(no_ID.Text);
  FieldByName('room').Value := Trim(no_kamar.Text);
  FieldByName('cock_in').Value := (cock_in.DateTime);
  FieldByName('cock_out').Value := (cock_out.DateTime);
  FieldByName('tgl_update').Value:=Now;
  FieldByName('tot_bayar').Value := Trim(total_bayar.Text);
  Post;
  ShowMessage('Tamu telah Check Out...!!!');
end;

//syntax menghapus data Reserv
//if ADOreserv.RecordCount > 0 then
  ADOdataTamu.Delete;
  ADOdataTamu.Requery();
end;

Refresh;
//ShowMessage('Tamu sudah CheckOut!!!');
bersih;
edit:=False;
KamarKosong;
KamarPesan;
KamarTersewa;
end;
end;

```

```

procedure Tfrm_Registrasi.KamarKosong;
var
  I : Integer;
begin
  for I:=0 to ComponentCount-1 do begin
    if (Components[I].ClassType=TPanel) then
      (Components[I] as TPanel).Color := clYellow;
    end;
  end;
end;

procedure Tfrm_Registrasi.KamarTersewa;
var
  X,I : Integer;
begin
  ADOdataTamu.Request;
  ADOt_kamarTersewa.Request;
  for I:=ADOt_kamarTersewa.RecordCount downto 0 do begin
    if (Components[(ADOt_kamarTersewa.X.Value)] ClassType = TPanel) then
      (Components[(ADOt_kamarTersewa.X.Value)] as TPanel).Color := clRed;
    ADOt_kamarTersewa.Next;
  end;
  {for I:=100 downto 89 do begin
    if (Components[I].ClassType = TPanel) then
      (Components[I] as TPanel).Color := clRed;
    end;}
  end;

procedure Tfrm_Registrasi.KamarPesan;
var
  X,I : Integer;
begin
  ADOreserv.Request;
  ADOt_kamarPesan.Request;
  for I:=ADOt_kamarPesan.RecordCount downto 0 do begin
    if (Components[(ADOt_kamarPesan.X.Value)] ClassType = TPanel) then
      (Components[(ADOt_kamarPesan.X.Value)] as TPanel).Color := clAqua;
    ADOt_kamarPesan.Next;
  end;
end;

procedure Tfrm_Registrasi.btn_tambahClick(Sender: TObject);
begin
  bersih;
  Res_noPsn.Text:=FormatDateTime('yy yy dd mm ss mm hh',now);
end;

procedure Tfrm_Registrasi.depositChange(Sender: TObject);
var
  lbl_deposit.Caption:=FormatText('#,##0.00',StrToFloat(deposit.Text));
end;

procedure Tfrm_Registrasi.terhutangChange(Sender: TObject);
begin
  lbl_terhutang.Caption:=FormatText('#,##0.00',StrToFloat(terhutang.Text));
end;

procedure Tfrm_Registrasi.totol_bayarChange(Sender: TObject);
begin
  lbl_totolByr.Caption:=FormatText('#,##0.00',StrToFloat(totol_bayar.Text));
end;

procedure Tfrm_Registrasi.Button1Click(Sender: TObject);
begin
  data_tamu.Show;
end;

procedure Tfrm_Registrasi.DBGrid3CellClick(Column: TColumn);
var
  jnsKlmm:String;
begin
  edit:=True;
  With ADO_cekout do begin
    txt_nodflrTamuCO.Text := FieldValues[no_pisafar];
    txt_nml_gkapCO.Text := FieldValues[nama_jengkap];
    txt_almtAsiCO.Text := FieldValues[alamat];
    txt_tupjLahirCO.Text := FieldValues[tempat_lahir];
    txt_tglLhrCO.Date := (FieldValues[tgl_lahir]);

    jnsKlmm := FieldValues[jns_kelamin];
    if jnsKlmm = 'L' then begin
      RadioLaki_CO.Checked := True;
    end else
      RadioCwek_CO.Checked := True;

    txt_IDCO.Text := Trim(FieldValues[ID]);
    txt_nolIDCO.Text := Trim(FieldValues[no_ID]);
    txt_roomCO.Text := Trim(FieldValues[room]);
    txt_cekInCO.Date := FieldValues[cek_in];
    txt_cekOutCO.Date := FieldValues[cek_out];

    txt_totByrCO.Text := Trim(FieldValues[tot_bayar]);

  end;
end;

```

```

procedure Tfrm_Registrasi.btn_simpanClick(Sender: TObject);
var

```

```

cek:Boolean;
begin
if (txt_nodfrTamuCO.Text='') or (txt_nmlgkapCO.Text='') or (txt_almtAsiCO.Text='') or
(txt_nollXCO.Text='') then
begin
MessageDlg('Maaf Data Masih Ada yang Kosong. Harap Di isi Terlebih
Dahulu.',mfcConfirmation,[mbOK]0);
end
else
begin
if (ctrl=False) then begin
cek:=False;
if (not cek=cekID(txt_nodfrTamuCO.Text)) then begin
ShowMessage('No Pertemuan Sudah Ada');
Exit;
end;
ADO_cekout.Appeal;
end else
ADO_cekout.Edit;
end
end
end

```

```

with ADO_cekout do begin
FieldByName('no_Pendaftaran').Value := Trim(txt_nodfrTamuCO.Text);
FieldByName('nama_lengkap').Value := Trim(txt_nmlgkapCO.Text);
FieldByName('alamat').Value := Trim(txt_almtAsiCO.Text);
FieldByName('tempat_lahir').Value := Trim(txt_tmplahirCO.Text);
FieldByName('tgl_lahir').Value := (txt_tglLhrCO.Date);

```

```

if RadioLaki_CO.Checked = True then
begin
FieldByName('jns_kelamin').Value := 'L';
end
else
begin
FieldByName('jns_kelamin').Value := 'P';
end;

```

```

FieldByName('ID').Value := Trim(txt_IDCO.Text);
FieldByName('no_ID').Value := Trim(txt_nolDCCO.Text);
FieldByName('room').Value := Trim(txt_roomCO.Text);
FieldByName('cek_in').Value := (txt_cekInCO.Date Time);
FieldByName('cek_out').Value := (txt_cekOutCO.Date Time);
FieldByName('tot_bayar').Value := Trim(txt_totByrCO.Text);
FieldByName('tgl_update').Value := Now;
Post;
end;

```

```

//syntax menghapus data Reserv
//if ADOreserv.RecordCount > 0 then
//if simpan.Caption=SIMPAN_ ' then
//ADOreserv.Delete;
//ADOreserv.Requery();

Refresh;
ShowMessage('Data Berhasil tersimpan!!!');
bersih;

end;

end;

procedure Tfrm_Registrasi.btn_cetakClick(Sender: TObject);
begin
frm_qrt_cekout.QR_rpt_cekout.Preview;
end;

end;

{
//
Form Restaurant
//
}

procedure Tfrm_restaurant.hersih;
begin
/no_trans_resto.Text := '';
txt_nodaftar.Text := '';
txt_nama.Text := '';
txt_ruam.Text := '';
txt_bayk1.Text := '0';
txt_nmlMkn1.Text := '';
txt_bargal.Text := '0';
txt_totMenu1.Text := '0';
txt_totResto.Text := '0';
end;

procedure Tfrm_restaurant.btn_simpanClick(Sender: TObject);
var
cek:Boolean;
begin
if (no_trans_resto.Text='') then
begin

```

```

        MessageBox("Maaf no Transaksi Restoran masih Kosong, Harap Di isi Terlebih  

        Dahulu",miConfirmation,mbOK);
    end
    else
        begin
            hitungtotResto;
            if(edt=False) then begin
                ADOresto.Append;
            end else
                ADOresto.Edit;

            with ADOresto do begin
                FieldByName('no_trans_resto').Value := Trim(no_trans_resto.Text);
                FieldByName('no_pdaftar').Value := Trim(txt_nodaftar.Text);
                FieldByName('nama').Value := Trim(txt_nama.Text);
                FieldByName('room').Value := Trim(txt_room.Text);
                FieldByName('um_menu').Value := Trim(txt_nmMka1.Text);
                FieldByName('hrg_menu').Value := Trim(txt_harga1.Text);
                FieldByName('bny_menu').Value := Trim(txt_bnyk1.Text);
                FieldByName('total_biaya_resto').Value:= Trim(txt_totMenu1.Text);
                FieldByName('tgl_update').Value := Now;
            Post;
            end;

            Refresh;
            ShowMessage("Data Berhasil tersimpan!!!");
            //bersih;
            pesankosong;

            edt:=False;
            end;
        end;

        procedure Tfrm_restaurant.FormCreate(Sender: TObject);
        begin
            ADOresto.Active:=True;
            end;

        procedure Tfrm_restaurant.bn_tambahClick(Sender: TObject);
        begin
            bersih;
            no_trans_resto.Text:=FormatDateTime('yyyymmddhhmmss',now);
            end;

        procedure Tfrm_restaurant.bn_keluarClick(Sender: TObject);
        begin

```

```

        Close;
        end;

        procedure Tfrm_restaurant.hitungtotResto;
        var
            jml : Integer;
            totResto, totSml,hrg,totAkhr : Currency;
        begin
            jml :=StrToInt(txt_bnyk1.Text);
            hrg :=StrToCurr(txt_harga1.Text);
            totResto := jml * hrg;

            totSml := StrToInt(lbl_totByrResto.Caption);
            totAkhr := totSml + totResto;

            lbl_totByrResto.Caption := CurrToStr(totAkhr);
            //lbl_totByrResto.Caption := FormatFloat('#0.00',StrToInt(txt_totSmlr.Text));

            txt_totSmlr.Text := CurrToStr(totResto);
        end;

        procedure Tfrm_restaurant.hitungClick(Sender: TObject);
        begin
            hitungtotResto;
            end;

        procedure Tfrm_restaurant.hitungMkn;
        begin
        end;

        procedure Tfrm_restaurant.hitungMnurn;
        begin
        end;

        procedure Tfrm_restaurant.DBGrid1CellClick(Column: TColumn);
        begin
            edt:=True;
            with ADOresto do begin
                no_trans_resto.Text := Trim(FieldValues['no_trans_resto']);
                txt_nodaftar.Text := Trim(FieldValues['no_pdaftar']);
                txt_nama.Text := Trim(FieldValues['nama']);
                txt_room.Text := Trim(FieldValues['room']);
                txt_totResto.Text := Trim(FieldValues['total_biaya_resto']);
            end;
        end;
    end;

```

```

begin
If txt_bnyk1.Text = " then begin
txt_bnyk1.Text := 0;
end else begin
txt_totMenu1.Text := IntToStr(StrToInt(txt_bnyk1.Text) * StrToInt(txt_harga1.Text));
lbl_jmlMknCaption := FormatFloat('#,##0.00', StrToInt(txt_totMenu1.Text));
end;
end;

procedure Tfrm_restaurant.txt_roomChange(Sender: TObject);
var
s:String;
begin
if (length(txt_room.Text)>0) then begin
ADOQ_reg.Locate('no_pdaftar', txt_room.Text, [loPartialKey, loCaseInsensitive]);
refresh;
end;

s:=SELECT 'no_pdaftar, nama, room FROM T_Reg where room LIKE ' + txt_pRoom';
with ADOQ_reg do begin
sql.Clear;
sql.Add(s);
Parameters.ParamValues[pRoom]:= % + txt_room.Text + %;
ExecSQL;
Close;
Open;
end;
if ADOQ_reg.RecordCount>0 then begin
txt_nodaftar.Text := ADOQ_reg.FieldValues[no_pdaftar];
txt_nama.Text := ADOQ_reg.FieldValues[nama];
end;
end;

procedure Tfrm_restaurant.pesankosong;
begin
txt_kodeMkn1.Text:= '';
txt_mnMkn1.Text:= '';
txt_bnyk1.Text:=0;
txt_harga1.Text:=0;
txt_totMenu1.Text:=0;
end;

procedure Tfrm_restaurant.txt_bnyk1KeyPress(Sender: TObject);

```

```

procedure Tfrm_restaurant.btn_menuMknClick(Sender: TObject);
begin
frm_pktMkn.Show;
end;

function Tfrm_Registrasi.cekID(ID: string): boolean;
begin
if (Not ADODataTamu1.Locate('no_pdaftar', ID, [loPartialKey, loCaseInsensitive])) then
Result:=False
else
Result:=True;
end;

procedure Tfrm_restaurant.txt_kodeMkn1Change(Sender: TObject);
var
buy, lag, jml: Integer;
s:String;
begin
if (length(txt_kodeMkn1.Text)>0) then begin
ADOmenu.Locate('kode', txt_kodeMkn1.Text, [loPartialKey, loCaseInsensitive]);
refresh;
end;

s:=SELECT kode, [nama_mkn/mnum], harga FROM T_kodeMkn where kode LIKE ' + txt_kodeMkn1.Text + %;
with ADOQ_menuMkn do begin
sql.Clear;
sql.Add(s);
Parameters.ParamValues[pKode]:= % + txt_kodeMkn1.Text + %;
ExecSQL;
Close;
Open;
end;
if ADOQ_menuMkn.RecordCount>0 then begin
txt_mnMkn1.Text := ADOQ_menuMkn.FieldValues[nama_mkn/mnum];
txt_harga1.Text := ADOQ_menuMkn.FieldValues[harga];
end;

jml:=StrToInt(txt_totMenu1.Text);
buy:=StrToInt(txt_bnyk1.Text);
hrg:=StrToInt(txt_harga1.Text);

jml:=s buy * hrg;

txt_totMenu1.Text := IntToStr(jml);
end;

procedure Tfrm_restaurant.txt_bnyk1Change(Sender: TObject);

```

```

var Key: Char;
begin
  if key = chr(13) then begin
    if (txt_kodemakml.Text='') then begin
      ShowMessage('Data Belum Terisi !!!');
    end else
      if (txt_bnyk1.Text='') then begin
        ShowMessage('Jumlah Barang Belum Terisi !!!');
      end else
        if (txt_bnyk1.Text='0') then
          begin
            hitungtotRear;
            btn_simpan.Click;
            pesan kosong;
          end;
        end;
      end;
    end;
  end;
end;

```

```
procedure Tfrm_restaurant.btn_cetakClick(Sender: TObject);
```

```

var
  s:String;
begin
  s:=SELECT = From T_resto where no_trains_resto LIKE pNo;
  {with ADOQ_reg do begin
    sql.Clear;
    sql.Add(s);
    Parameters.ParamValues['pRoom'] := %s+ txt_room.Text + %s;
    ExecSQL;
    Close;
    Open;
  end;
  if ADOQ_reg.RecordCount>0 then begin
    txt_nodeflar.Text = ADOQ_reg.FieldValues['no_P3aflar'];
    txt_namaa.Text := ADOQ_reg.FieldValues['namaa'];
  end;
end;

```

```

8 with frm_qr_resto.ADOQ_resto do begin
    sql.Clear;
    sql.Add(s);
    Parameters.ParamValues["NO"]:= '%'+no_trans_resto.Text+'%';
    ExecSQL;
    Close;
    Open;
    end;
    frm_qr_resto.QR_rpt_resto.Preview;
end;

```

```
procedure Tfrm_restaurant.txt_nodafiarChange(Sender: TObject);
var
  s:String;
```

```
begin
  if (length(txt_nodeflar.Text) > 0) then begin
    ADXQ_reg.Locate("no_Pdaflar.txt_nodeflar.Text.[loPartialKey.toCaseInsensitive]");
    refresh;
  end;
end;
```

```

s:=SELECT no_pdaflar, name, room FROM T_Reg where no_pdaflar LIKE :pNo;
with ADOQ_reg do begin
    sql Clear;
    sql.Add(s);
    Parameters.ParamValues[pNo]:="%" + lst_nodeflar.Text + "%";
    ExecSQL;
    Close;
    Open;
end;
if ADOQ_reg.RecordCount>0 then begin
    lst_room.Text := ADOQ_reg.FieldValues[room];
    txt_name.Text := ADOQ_reg.FieldValues[name];
end;

```

end

and

Form Transaksi Bina Fasilitas

 $\sqrt{1}$

44

procedure Thiava fasilitas.bersih:

```
begin
  txt_notranFas.Text := '';
  txt_noDfr.Text := '';
  txt_kodeFas.Text := '';
  txt_namaFas.Text := '';
  txt_lrgFas.Text := '0';
  txt_jmlhPakai.Text := '0';
  txt_totbiayaFas.Text := '0';
end;
```

```
procedure Thiava fasilitas.btn_simpanClick(Sender: TObject);
```

var

cek-1

```
begin
    if (txt_namPak.Text="" or (x1_nuDir.Text="" or (txt_kodeFas.Text="" or (txt_namFas.Text="" or
        txt_jmlPaket.Text="" then
```

```

begin
    MessageDlg('Maaf Data Masih Ada yang Kosong, Harap Di isi Terlebih  
Dahulu.', mtConfirmation, [mbOK], 0);
end
else
begin
    if (edit=False) then begin
        cek:=False;
        if (not cek=cekID(txt_noDftr.Text)) then begin
            ShowMessage('Nomor Pendaftaran belum terisi Harap di isi terlebih dulu');
            Exit;
        end;
        ADOfasilitas.Append;
    end else
        ADOfasilitas.Edit;
    with ADOfasilitas do begin
        FieldByName('no_trans_fas').Value := Trim(txt_notranFas.Text);
        FieldByName('no_Pdaftar').Value := Trim(txt_noDftr.Text);
        FieldByName('kode_fas').Value := Trim(txt_kodeFas.Text);
        FieldByName('nama_fas').Value := Trim(txt_namFas.Text);
        FieldByName('jumlah_pakai').Value := Trim(txt_jmlhPakan.Text);
        FieldByName('total_biaya_fas').Value := Trim(txt_totbiayaFas.Text);
        FieldByName('tgl_update').Value := Now;
    end;
    Refresh;
    ShowMessage('Data Berhasil tersimpan!');
    bersih;
    edit:=false;
end;
end;

procedure Tbaya_fasilitas.btn_tambahClick(Sender: TObject);
begin
    bersih;
    txt_notranFas.Text:=FormatDateTime('yyyy dmm hhamss', now);
end;

procedure Tbaya_fasilitas.btn_keluarClick(Sender: TObject);
begin
    Close;
end;

procedure Tbaya_fasilitas.btn_hrgFasClick(Sender: TObject);
begin
    frm_hrgFas.Show;
end;

```

```

end;

function Tbaya_fasilitas.ccekID(ID: string): boolean;
begin
    if (Not ADOfasilitas.Locate('no_Pdaftar', ID, [loPartialKey, loCaseInsensitive])) then
        Result:=False
    else
        Result:=True;
    end;

procedure Tbaya_fasilitas.txt_kodeFasChange(Sender: TObject);
var
    s:String;
begin
    if (length(txt_kodeFas.Text)>0) then begin
        ADOhrgFas.Locate('kode_fas', txt_kodeFas.Text, [loPartialKey, loCaseInsensitive]);
        refresh;
    end;

    s:='SELECT kode_fas, nama_fas, harga_fas FROM T_hrgFas where kode_fas LIKE :pKode';
    with ADOQ_hrgFas do begin
        sql.Clear;
        sql.Add(s);
        Parameters.ParamValues['pKode']:='%'+txt_kodeFas.Text+'%';
        ExecSQL;
        Close;
        Open;
    end;
    if ADOQ_hrgFas.RecordCount>0 then begin
        txt_namFas.Text := ADOQ_hrgFas.FieldValues['nama_fas'];
        txt_hrgFas.Text := ADOQ_hrgFas.FieldValues['harga_fas'];
    end;
end;

procedure Tbaya_fasilitas.DB_FasCellClick(Column: TColumn);
begin
    edit:=True;
    With ADOfasilitas do begin
        txt_notranFas.Text := Trim(FieldValues['no_trans_fas']);
        txt_noDftr.Text := Trim(FieldValues['no_Pdaftar']);
        txt_kodeFas.Text := Trim(FieldValues['kode_fas']);
        txt_namFas.Text := Trim(FieldValues['nama_fas']);
        txt_jmlhPakai.Text := Trim(FieldValues['jumlah_pakai']);
        txt_totbiayaFas.Text := Trim(FieldValues['total_biaya_fas']);
    end;
end;

```

```
procedure Thiaya_fasilitas.hgBiayaFas;
var
    hrgFas, tolPakai, totFas : Integer;
begin
    totFas:=StrToInt(txt_totbiayaFas.Text);
    tolPakai:=StrToInt(txt_jmlhPakai.Text);
    hrgFas:=StrToInt(txt_hrgfas.Text);
    totFas := hrgFas * tolPakai;
    txt_totbiayaFas.Text := IntToStr(totFas);
    lbl_totBiayaFas.Caption := FormatFloat('0.##0.00',StrToInt(txt_totbiayaFas.Text));
end;

procedure Thiaya_fasilitas.hitungClick(Sender: TObject);
begin
    hgBiayaFas;
end;

end;

{
//=====
Form Transaksi Biaya Telephone
//=====
}

procedure Thiaya_telp.bersih;
begin
    txt_notranTelp.Text:='';
    txt_noDflar.Text:='';
    txt_lmaTelp.Text:='';
    txt_tujuan.Text:='';
    txt_biayaTelp.Text:='';
end;

procedure Thiaya_telp.btn_simpanClick(Sender: TObject);
var
    cek:Boolean;
begin
    cek:Boolean;
    if (txt_notranTelp.Text='') or (txt_noDflar.Text='') or (txt_lmaTelp.Text='') or (txt_tujuan.Text='') or
    (txt_biayaTelp.Text='') then
    begin
        MessageDlg('Maaf Data Masih Ada yang Kosong, Harap Di isi Terlebih
        Dahulu',mtConfirmation,[mbOK][0];
    end
end;
```

```
else
begin
    if(cek=False) then begin
        cek:=False;
        if(not cek=cekID(txt_noDflar.Text)) then begin
            ShowMessage('Nomor Pendaftaran belum terisi Harap di isi terlebih dulu');
            Exit;
        end;
        A[X]telp.Append;
    end else
        ADOtelp.Edit;
end;

with ADOtelp do begin
    FieldByName('no_trans_telp').Value := Trim(txt_notranTelp.Text);
    FieldByName('no_pdaflar').Value := Trim(txt_noDflar.Text);
    FieldByName('lma_telp').Value := Trim(txt_lmaTelp.Text);
    FieldByName('tujuan').Value := Trim(txt_tujuan.Text);
    FieldByName('total_biaya_telp').Value := Trim(txt_biayaTelp.Text);
    FieldByName('tgl_update').Value := Now;
end;

Post;

end;

Refresh;
ShowMessage('Data Berhasil tersimpan!');
bersih;
cek:=False;
end;

function Thiaya_telp.cekID(ID: string): boolean;
begin
    if(Not ADOtelp.Locate('no_pdaflar',ID,[loPartialKey,loCaseInsensitive])) then
        Result:=False
    else
        Result:=True;
    end;
end;

procedure Thiaya_telp.btn_tambahClick(Sender: TObject);
begin
    bersih;
    txt_notranTelp.Text:=FormatDateTime('yyyy ddmm hhmmss',now);
end;

procedure Thiaya_telp.Button4Click(Sender: TObject);
begin
    Close;
end;
```

```
procedure Tbiaya_telp.DBGrid1Click(Column: TColumn; TColumn):
```

```
begin
```

```
edi:=True;
```

```
With ADOtelp do begin
```

```
txt_notranTelp.Text:=FieldValues[no_trans_telp];
```

```
txt_notdftar.Text:=FieldValues[no_Pendaftaran];
```

```
txt_luasTelp.Text:=FieldValues[luas_telp];
```

```
txt_tujuan.Text:=FieldValues[tujuan];
```

```
txt_biayaTelp.Text:=FieldValues[total_biaya_telp];
```

```
end;
```

```
end;
```

```
end
```

```
{  
//  
Form Transaksi Tagihan Tamu  
//  
}
```

```
procedure Tfrm_tagihanTamu.bersih;
```

```
begin
```

```
no_bayar.Text:='', no_daftar.Text:='',
```

```
nama.Text:='', alamat.Text:='',
```

```
ID.Text:='', no_ID.Text:='',
```

```
{if RadioTunai.Checked = True then begin
```

```
end; }
```

```
biaya_kmr.Text:='0'; biaya_fas.Text:='0';
```

```
biaya_resto.Text:='0'; biaya_telp.Text:='0';
```

```
total_biaya.Text:='0';
```

```
end;
```

```
procedure Tfrm_tagihanTamu.btn_keluarClick(Sender: TObject);
```

```
begin
```

```
Close;
```

```
end;
```

```
procedure Tfrm_tagihanTamu.ToPembayaran;
```

```
var
```

```
b_kmr,b_fas,b_tel,b_res,tot : Currency;
```

```
begin
```

```
b_kmr := StrToCurr(biaya_kmr.Text);
```

```
b_fas := StrToCurr(biaya_fas.Text);
```

```
b_tel := StrToCurr(biaya_telp.Text);
```

```
b_res := StrToCurr(biaya_resto.Text);
```

```
tot := (b_kmr + b_fas + b_tel + b_res);
```

```
total_biaya.Text := CurrToStr(tot);
```

```
end;
```

```
procedure Tfrm_tagihanTamu.FormCreate(Sender: TObject);
```

```
begin
```

```
ADODaftar.Active:=True;
```

```
ADODaftar.Reg.Active:=True;
```

```
ADODaftar.Fasilitas.Active:=True;
```

```
ADODaftar.Kamar.Active:=True;
```

```
ADODaftar.Telepon.Active:=True;
```

```
ADODaftar.Resto.Active:=True;
```

```
ADODaftar.Tagihan.Active:=True;
```

```
CarNotDaftar,
```

```
bersih;
```

```
end;
```

```
procedure Tfrm_tagihanTamu.btn_simpanClick(Sender: TObject);
```

```
var
```

```
cek:Boolean;
```

```
begin
```

```
if (no_daftar.Text='') or (nama.Text='') or (alamat.Text='') or (no_ID.Text='') or (no_bayar.Text='')
```

```
then
```

```
begin
```

```
MessageDlg('Maaf! Data Masih Ada yang Kosong, Harap Di isi Terlebih
```

```
Dahulu.',mtConfirmation,[mbOK],0);
```

```
end
```

```
else
```

```
begin
```

```
if (cek=False) then begin
```

```
cek:=False;
```

```
if (not cek=cekID(no_ID.Text)) then begin
```

```
ShowMessage('No Pembayaran Salah Ads');
```

```
Exit;
```

```
end;
```

```
ADODaftar.Append;
```

```
end else
```

```
ADODaftar.Edit;
```

```
with ADOtagihan do begin
```

```
FieldByName('no_bayar').Value := Trim(no_bayar.Text);
```

```
FieldByName('no_Pendaftaran').Value := Trim(no_daftar.Text);
```

```
FieldByName('nama').Value := (nama.Text);
```

```
FieldByName('alamat').Value := Trim(alamat.Text);
```

```

FieldByName('ID').Value := Trim(ID.Text);
FieldByName('no_ID').Value := Trim(no_ID.Text);
FieldByName('biaya_kamar').Value := (biaya_kmr.Text);
FieldByName('biaya_fasilitas').Value := Trim(biaya_fas.Text);
FieldByName('biaya_telp').Value := Trim(biaya_telp.Text);
FieldByName('biaya_resto').Value := Trim(biaya_resto.Text);
FieldByName('total_biaya').Value := Trim(total_biaya.Text);
Post;
end;

Refresh;
ShowMessage('Data Berhasil tersimpan!');
bersih;
exit := False;
end;

end;

procedure Tfrm_tagihanTamu.CariNoDaftar;
var
  s:String;
  ByaKamar,ByaTerhutang : Currency;
begin
  s:='SELECT no_Pdaftar,nama,alamat,ID,no_ID FROM T_Reg where no_Pdaftar LIKE :pNO';
  with ADOQ_Reg do begin
    sql.Clear;
    sql.Add(s);
    Parameters.ParamValues['pNO']:='%'+no_daftar.Text+'%';
    ExecSQL;
    Close;
    Open;
  end;
  if ADOQ_Reg.RecordCount>0 then begin
    nama.Text := ADOQ_Reg.FieldValues['nama'];
    alamat.Text := ADOQ_Reg.FieldValues['alamat'];
    ID.Text := ADOQ_Reg.FieldValues['ID'];
    no_ID.Text := ADOQ_Reg.FieldValues['no_ID'];
  end;
end;

procedure Tfrm_tagihanTamu.no_daftarChange(Sender: TObject);
begin
  ADOQ_Reg.Requery;ADOQ_Fasilitas.Requery;ADOQ_Kamar.Requery;
  ADOQ_Telepon.Requery;ADOQ_Resto.Requery;
  CariNoDaftar;
  CariByaKamar;
  CariByaFas;
  CariByaTelp;
  CariByaResto;

```

```

TotPembayaran;
end;

procedure Tfrm_tagihanTamu.CariByaKamar;
var
  s:String;
  ByaKamar,ByaTerhutang : Currency;
begin
  s:='SELECT no_Pdaftar,total_akhir_bayar,terhutang FROM T_detailBayar where no_Pdaftar LIKE :pNO';
  with ADOQ_Kamar do begin
    sql.Clear;
    sql.Add(s);
    Parameters.ParamValues['pNO']:='%'+no_daftar.Text+'%';
    ExecSQL;
    Close;
    Open;
  end;
  if ADOQ_Kamar.RecordCount>0 then begin
    ByaKamar := ADOQ_Kamar.FieldValues['total_akhir_bayar'];
    ByaTerhutang := ADOQ_Kamar.FieldValues['terhutang'];

    if ByaTerhutang > 0 then
      biaya_kmr.Text := CurrToStr(ByaTerhutang)
    else
      biaya_kmr.Text := CurrToStr(ByaKamar);
    end
  else begin
    biaya_kmr.Text:='0';
  end;
end;

end;

procedure Tfrm_tagihanTamu.CariByaFas;
var
  s:String;
begin
  s:='SELECT no_Pdaftar,total_biaya_fas FROM T_trans_fas where no_Pdaftar LIKE :pNO';
  with ADOQ_Fasilitas do begin
    sql.Clear;
    sql.Add(s);
    Parameters.ParamValues['pNO']:='%'+no_daftar.Text+'%';
    ExecSQL;
    Close;
    Open;
  end;
  if ADOQ_Fasilitas.RecordCount>0 then

```

```

        biaya_fas.Text := ADOQ_Fasilitas.FieldValues('total_biaya_fas')
    else
        biaya_fas.Text := '0';
    end;

procedure Tfrm_lagihanTamu.CariByaTelp;
var
    s:String;
begin
    s:='SELECT no_Pdaftar,total_biaya_telp FROM T_trans_telp where no_Pdaftar LIKE :pNO';
    with ADOQ_Telepon do begin
        sql.Clear;
        sql.Add(s);
        Parameters.ParamValues['pNO']:='%'+no_daftar.Text+'%';
        ExecSQL;
        Close;
        Open;
    end;

    if ADOQ_Telepon.RecordCount>0 then
        biaya_telp.Text := ADOQ_Telepon.FieldValues('total_biaya_telp')
    else
        biaya_telp.Text := '0';
    end;

procedure Tfrm_lagihanTamu.CariByaResto;
var
    s:String;
begin
    s:='SELECT no_Pdaftar,total_biaya_resto FROM T_resto where no_Pdaftar LIKE :pNO';
    with ADOQ_Resto do begin
        sql.Clear;
        sql.Add(s);
        Parameters.ParamValues['pNO']:='%'+no_daftar.Text+'%';
        ExecSQL;
        Close;
        Open;
    end;

    if ADOQ_Resto.RecordCount>0 then
        biaya_resto.Text := ADOQ_Resto.FieldValues('total_biaya_resto')
    else
        biaya_resto.Text := '0';
    end;

procedure Tfrm_lagihanTamu.btn_cetakClick(Sender: TObject);
var
    s:String;
    cek:Boolean;
begin

```

```

    if (no_daftar.Text='') or (nama.Text='') or (alamat.Text='') or (no_ID.Text='') or (no_bayar.Text='')
    then
        begin
            MessageDlg('Maaf Data Masih Ada yang Kosong, Harap Di isi Terlebih
            Dahulu.',mtConfirmation,mbOK,1,0);
        end
    else
        begin
            if(cek=False) then begin
                cek:=False;
                if(not cek=cekIDXno_ID.Text) then begin
                    ShowMessage('No Pembayaran Sudah Ada');
                    Exit;
                end;
                ADOlagihan.Append;
            end else
                ADOlagihan.Edit;

            with ADOlagihan do begin
                FieldByName('no_bayar').Value := Trim(no_bayar.Text);
                FieldByName('no_Pdaftar').Value := Trim(no_daftar.Text);
                FieldByName('nama').Value := (nama.Text);
                FieldByName('alamat').Value := Trim(alamat.Text);
                FieldByName('ID').Value := Trim(ID.Text);
                FieldByName('no_ID').Value := Trim(no_ID.Text);
                FieldByName('biaya_kamar').Value := (biaya_kmr.Text);
                FieldByName('biaya_fasilitas').Value := Trim(biaya_fas.Text);
                FieldByName('biaya_telp').Value := Trim(biaya_telp.Text);
                FieldByName('biaya_resto').Value := Trim(biaya_resto.Text);
                FieldByName('total_biaya').Value := Trim(total_biaya.Text);
            Post;
            end;

            Refresh;

            cdt:=False;
        end;

    frm_Rpt_Dftr_lagihan.ADOQ_lagihan.Requery();

    s:='SELECT * FROM T_cashier WHERE no_Pdaftar like no:
    with frm_Rpt_Dftr_lagihan.ADOQ_lagihan do begin
        SQL.Clear;
        SQL.Add(s);
        Parameters.ParamValues['no']:='%'+frm_lagihanTamu.no_daftar.Text+'%';
        ExecSQL;
        Close;
        Open;
    end;

```

```

frm_Rpt_DthTagihan_QR_Frm_Rpt_DthTagihan Preview;
//bersih;
end;

function Tfrm_lagihanTamu.cekID(ID, string): boolean;
begin
if(Not ADOAggihan.Locate('no_Pdaftar',ID,[loPartialKey,loCaseInsensitive])) then
    Result:=False
else
    Result:=True;
end;

procedure Tfrm_lagihanTamu.DBGrid1CellClick(Column TColumn);
begin
od:=True;
With ADOAggihan do begin
no_bayar.Text := Trim(FieldValues['no_bayar']);
no_dafar.Text := Trim(FieldValues['no_Pdaftar']);
nama.Text := Trim(FieldValues['nama']);
alamat.Text := Trim(FieldValues['alamat']);
ID.Text := Trim(FieldValues['ID']);
no_ID.Text := Trim(FieldValues['no_ID']);
biaya_kmr.Text := Trim(FieldValues['biaya_kamar']);
biaya_fas.Text := Trim(FieldValues['biaya_fasilitas']);
biaya_telp.Text := Trim(FieldValues['biaya_telp']);
biaya_resto.Text := Trim(FieldValues['biaya_resto']);
total_biaya.Text := Trim(FieldValues['total_biaya']);
end;

end;

end;

end;
{
//
}
Form Rugi Laba

//
}

procedure Tfrm_rugiLaba.btn_simpanClick(Sender: TObject);
var
cek:Boolean;
begin
if (tx_nolapRL.Text='') then
begin

```

```

    MessageDlg('Maaf Nomor Laporan Masih Kosong, Harap Di isi Terlebih  

Dahulu.',mtConfirmation,[mbOK],0);
end
else
begin
if(od=False) then begin
end;
ADO_RL.Append;
ADO_RL.Edit;

with ADO_RL do begin
FieldByName('no_lapRL').Value := Trim(tx_nolapRL.Text);
FieldByName('nsk_kmr').Value := Trim(tx_totDuitKmr.Text);
FieldByName('nsk_resto').Value := Trim(tx_totDuitResto.Text);
FieldByName('nsk_fas').Value := Trim(tx_totDuitFas.Text);
FieldByName('nsk_telp').Value := Trim(tx_totDuitTelp.Text);
FieldByName('kr_histrik').Value := Trim(tx_totLstrik.Text);
FieldByName('kr_air').Value := Trim(tx_totAir.Text);
FieldByName('kr_lain').Value := Trim(tx_krlain.Text);
FieldByName('tot_msk').Value := Trim(tx_totMasuk.Text);
FieldByName('tot_kr').Value := Trim(tx_totKluar.Text);
FieldByName('hasil_RL').Value := Trim(tx_rugikluar.Text);
FieldByName('tgl_update').Value := Now;

Post;
end;

//syntax manghapus data Rugi laba
//ADO_RL.Delete;

Refresh;
ShowMessage('Data Berhasil tersimpan!!!');
bersih;
edl:=False;

end;
end;

procedure Tfrm_rugiLaba.bersih;
begin
tx_nolapRL.Text := ''; tx_totDuitKmr.Text:=0;
tx_totDuitResto.Text:=0; tx_totDuitFas.Text:=0;
tx_totDuitTelp.Text :=0;
tx_totLstrik.Text :=0; tx_totAir.Text :=0;
tx_krlain.Text :=0; tx_totMasuk.Text :=0;
tx_totKluar.Text :=0; tx_rugikluar.Text :=0;

end;
end;

```

```

procedure Tfrm_rugiLaba.btn_tambahClick(Sender: TObject);
begin
  bersih;
  tx_nolapRL.Text:=FormatDateTime('yyyy.ddmm.mmhhss',now);
end;

function Tfrm_rugiLaba.cekID(string): boolean;
begin
  if (Not ADOdataTamu.Locate('no_Pdaftar',ID,[loPartialKey,loCaseInsensitive])) then
    Result:=False
  else
    Result:=True;
  end;
end;

procedure Tfrm_rugiLaba.hitungRugiLaba;
var
  msKmr, msRsto, msFas, msTelp, krSt, krAir, krL: Integer,
  totMsk, totKr, hslRL: integer;
begin
  msKmr := StrToInt(tx_totDuitKmr.Text);
  msRsto := StrToInt(tx_totDuitResto.Text);
  msFas := StrToInt(tx_totDuitFas.Text);
  msTelp := StrToInt(tx_totDuitTelp.Text);
  krSt := StrToInt(tx_totStrik.Text);
  krAir := StrToInt(tx_totAir.Text);
  krL := StrToInt(tx_kirLain.Text);

  totMsk := StrToInt(tx_totMasuk.Text);
  totKr := StrToInt(tx_totKuar.Text);
  hslRL := StrToInt(tx_rugiLaba.Text);

  totMsk := msKmr + msRsto + msFas + msTelp;
  tx_totMasuk.Text := IntToStr(totMsk);
  totKr := krSt + krAir + krL;
  tx_totKuar.Text := IntToStr(totKr);
  hslRL := totMsk - totKr;
  tx_rugiLaba.Text := IntToStr(hslRL);
end;

procedure Tfrm_rugiLaba.btn_hitungClick(Sender: TObject);
begin
  hitungRugiLaba;
end;

procedure Tfrm_rugiLaba.DBGrid1CellClick(Column: TColumn);
begin
  edit:=True;

```

```

With ADO_RL do begin
  tx_nolapRL.Text := Trim(FieldValues['no_lapRL']);
  tx_totDuitKmr.Text := Trim(FieldValues['msk_kmr']);
  tx_totDuitResto.Text:= Trim(FieldValues['msk_rsto']);
  tx_totDuitFas.Text := Trim(FieldValues['msk_fas']);
  tx_totDuitTelp.Text := Trim(FieldValues['msk_telp']);
  tx_totStrik.Text := Trim(FieldValues['kr_strik']);
  tx_totAir.Text := Trim(FieldValues['kr_air']);
  tx_kirLain.Text := Trim(FieldValues['kr_lain']);
  tx_totMasuk.Text := Trim(FieldValues['tot_msk']);
  tx_totKuar.Text := Trim(FieldValues['tot_kr']);
  tx_rugiLaba.Text := Trim(FieldValues['hasil_RL']);
end;
end;

procedure Tfrm_rugiLaba.FormCreate(Sender: TObject);
begin
  //MonthCalendar1.Date =Now;
end;

procedure Tfrm_rugiLaba.cariBisaKmr,
var
  s:String;
  lop:integer;
  BysaKamar : Currency;
begin
  BysaKamar:=0;
  s:=SELECT no_ID, tot_hayar, week_out, tgl_update FROM T_kamar WHERE (tgl_update >=
  CONVERT(DATETIME,'+' + FormatDateTime('yyyy-mm-dd hh:mm:ss',DTP_drTglDateTime)+'
  '+', 102)) AND (tgl_update <= CONVERT(DATETIME,'+' + FormatDateTime('yyyy-mm-dd
  hh:mm:ss',DTP_mpeTglDateTime)+'+', 102));
  with ADOQ_RL.Kamar do begin
    sql Clear;
    sql.Add(s);
    ExecSQL;
    Close;
    Open;
  end;
  if ADOQ_RL.Kamar.RecordCount>0 then begin
    for lop:=0 to ADOQ_RL.Kamar.RecordCount-1 do begin
      BysaKamar:=BysaKamar + ADOQ_RL.Kamar.FieldValues['tot_hayar'];
      ADOQ_RL.Kamar.Next;
    end;
    tx_totDuitKmr.Text:=CurToStr(BysaKamar);
  end
  else begin
    tx_totDuitKmr.Text:=0;
  end;
end;

```

end;

```
procedure Tfrm_rugilaba.DTP_drTglChange(Sender: TObject);
```

begin

canByaResto;

canByaKmr;

canByaFas;

canByaTelp;

//tx_nolapRL.Text:=FormatDateTime('dd/mm/yyyy hh:mm:ss',DTP_drTgl.DateTime);

end;

```
procedure Tfrm_rugilaba.DTP_nipeTglChange(Sender: TObject);
```

begin

canByaKmr;

canByaResto;

canByaFas;

canByaTelp;

end;

```
procedure Tfrm_rugilaba.canByaResto;
```

var

s:String;

lop:integer;

ByaResto:Currency;

begin

ByaResto:=0;

s:=SELECT no_transaksi, total_biaya_resto, tgl_update FROM T_resto WHERE (tgl_update >=

CONVERT(DATETIME, ' + '' + FormatDateTime('yyyy-mm-dd hh:mm:ss', DTP_drTgl.DateTime) + ''

+', 102)) AND (tgl_update <= CONVERT(DATETIME, ' + '' + FormatDateTime('yyyy-mm-dd

hh:mm:ss', DTP_nipeTgl.DateTime) + '' + ', 102));

with ADOQ_RLResto do begin

sql.Clear;

sql.Add(s);

ExecSQL;

Close;

Open;

end;

if ADOQ_RLResto.RecordCount>0 then begin

for lop:=0 to ADOQ_RLResto.RecordCount-1 do begin

ByaResto:=ByaResto + ADOQ_RLResto.FieldValues['total_biaya_resto'];

ADOQ_RLResto.Next;

end;

tx_toliduitResto.Text:=CurrToStr(ByaResto);

end

else begin

tx_toliduitResto.Text:=0;

end;

end;

```
procedure Tfrm_rugilaba.canByaFas;
```

var

s:String;

lop:integer;

ByaFas:Currency;

begin

ByaFas:=0;

s:=SELECT total_biaya_fas, tgl_update FROM T_trans_fas WHERE (tgl_update >=

CONVERT(DATETIME, ' + '' + FormatDateTime('yyyy-mm-dd hh:mm:ss', DTP_drTgl.DateTime) + ''

+', 102)) AND (tgl_update <= CONVERT(DATETIME, ' + '' + FormatDateTime('yyyy-mm-dd

hh:mm:ss', DTP_nipeTgl.DateTime) + '' + ', 102));

with ADOQ_RLFas do begin

sql.Clear;

sql.Add(s);

ExecSQL;

Close;

Open;

end;

if ADOQ_RLFas.RecordCount>0 then begin

for lop:=0 to ADOQ_RLFas.RecordCount-1 do begin

ByaFas:=ByaFas + ADOQ_RLFas.FieldValues['total_biaya_fas'];

ADOQ_RLFas.Next;

end;

tx_toliduitFas.Text:=CurrToStr(ByaFas);

end;

else begin

tx_toliduitFas.Text:=0;

end;

end;

```
procedure Tfrm_rugilaba.canByaTelp;
```

var

s:String;

lop:integer;

ByaTelp:Currency;

begin

ByaTelp:=0;

s:=SELECT total_biaya_telp, tgl_update FROM T_trans_telp WHERE (tgl_update >=

CONVERT(DATETIME, ' + '' + FormatDateTime('yyyy-mm-dd hh:mm:ss', DTP_drTgl.DateTime) + ''

+', 102)) AND (tgl_update <= CONVERT(DATETIME, ' + '' + FormatDateTime('yyyy-mm-dd

hh:mm:ss', DTP_nipeTgl.DateTime) + '' + ', 102));

with ADOQ_RLTelp do begin

sql.Clear;

sql.Add(s);

```

ExecSQL;
Close;
Open;
end;
if ADOQ_RLTelp.RecordCount>0 then begin
for loop:=0 to ADOQ_RLTelp.RecordCount-1 do begin
ByaTelp:=ByaTelp + ADOQ_RLTelp.FieldValues[total_biaya_telp];
ADOQ_RLTelp.Next;
end;
tx_toduitTelp.Text:=CurrToStr(ByaTelp);
end
else begin
tx_toduitTelp.Text:='0';
end;
end;
end;
end;
end;
}
//=====
Form Entry Data Kamar
//=====
}

```

```

procedure Tfrm_entryDataKamar.bersih;

```

```

begin
tx_kdType.Text:='';
tx_nmType.Text:='';
tx_bodSize.Text:='';
tx_hrgKmr.Text:='';
end;

```

```

procedure Tfrm_entryDataKamar.FormCreate(Sender: TObject);

```

```

begin
ADOEntryKamar.Active:=True;
end;

```

```

procedure Tfrm_entryDataKamar.simpanClick(Sender: TObject);

```

```

var
cek:Boolean;
begin
if (tx_kdType.Text='') or (tx_nmType.Text='') or (tx_bodSize.Text='') or (tx_hrgKmr.Text='') then
begin
MessageDlg('Maaf Data Masih Ada yang Kosong. Harap Di isi Terlebih Dahulu.',mtConfirmation,[mbOK],0);
end
else

```

```

begin
if(edi=false) then begin
cek:=False;
if(not cek=cekID(txt_nmType.Text)) then begin
ShowMessage('Data kamar belum terisi penuh Harap di isi terlebih dulu');
Exit;
end;
ADOEntryKamar.Append;
end else
ADOEntryKamar.Edit;

with ADOEntryKamar do begin
FieldByName('kode_type').Value := Trim(txt_kdType.Text);
FieldByName('type_kamar').Value := Trim(txt_nmType.Text);
FieldByName('bod_size').Value := Trim(txt_bodSize.Text);
FieldByName('harga_kamar').Value := Trim(txt_hrgKmr.Text);

Post;
end;

Refresh;
ShowMessage('Data Berhasil tersimpan!!!');
bersih;
edi:=False;
end;
end;

```

```

procedure Tfrm_entryDataKamar.DBGridEntryKamarCellClick(Column: TColumn);

```

```

begin
edi:=True;
with ADOEntryKamar do begin
tx_kdType.Text := Trim(FieldValues['kode_type']);
tx_nmType.Text := Trim(FieldValues['type_kamar']);
tx_bodSize.Text := Trim(FieldValues['bod_size']);
tx_hrgKmr.Text := Trim(FieldValues['harga_kamar']);
end;
edi:=False;

```

```

end;

```

```

procedure Tfrm_entryDataKamar.keluarClick(Sender: TObject);

```

```

begin
Close;
end;

```

```

procedure Tfrm_entryDataKamar.tambahClick(Sender: TObject);

```

```

begin
bersih;

```

```

end;

procedure TForm_entryDataKamar.btn_entryNoKmrClick(Sender: TObject);
begin
    frm_tipeKamar.Show;
end;

function TForm_entryDataKamar.cekID(ID: string): boolean;
begin
    if(Not ADOentryKamar.Locate('tipe_kamar',ID,[loPartialKey,loCaseInsensitive])) then
        Result:=false
    else
        Result:=True;
    end;
end;

procedure TForm_entryDataKamar.btn_bapusClick(Sender: TObject);
begin
    ADOentryKamar.Locate('kode_tipe',txt_kdType.text,[loPartialKey,loCaseInsensitive]);
    if(Application.MessageBox('Anda Yakin menghapus record ini?','Peringatan',MH_YESNO or
    MB_ICONQUESTION)=IDYES)then begin
        ADOentryKamar.Delete;
        edt:=false;
        end;
        refresh;
        bersih;
    end;
end;

end;

{
//=====
Form Entry Data Karyawan
//=====
}

procedure TForm_entryDataKaryawan.bersih;
begin
    NIK.Text := '';
    nama.Text := '';
    tempat_lahir.Text := '';
    alamat.Text := '';
    kota.Text := '';
    no_telp.Text := '';
end;

procedure TForm_entryDataKaryawan.btn_simpanClick(Sender: TObject);
var
    cek:Boolean;
begin
    if (NIK.Text='') or (nama.Text='') or (tempat_lahir.Text='') or (alamat.Text='') or (kota.Text='') or
    (role_ID.Text='') then
        begin
            MessageDlg('Maaf! Data Masih Ada yang Kosong, Harap Di isi Terlebih
            Dahulu.',mfcConfirmation,[mbOK],0);
        end
    else
        begin
            if(cek=false) then begin
                cek:=false;
            if(Not cek=cekID(NIK.Text)) then begin
                ShowMessage('Nomor Induk Karyawan Sudah Ada');
                Exit;
            end;
            ADOdkrywn.Append;
        end
        end
    end
end;

end;

with ADOdkKrywn do begin
    FieldByName('nik').Value := Trim(NIK.Text);
    FieldByName('nama_karyawan').Value := Trim(nama.Text);
    FieldByName('tmp_lahir').Value := Trim(tempat_lahir.Text);
    FieldByName('tgl_lahir').Value := StrToDate(FormatDateTime('dd/mm/yyyy',tgl_lahir.Date));

    if RadioLaki.Checked = True then begin
        FieldByName('jenis_kelamin').Value := 'L';
    end
    else
        begin
            FieldByName('jenis_kelamin').Value := 'P';
        end;
end;

FieldByName('alamat').Value := Trim(alamat.Text);
FieldByName('kota').Value := Trim(kota.Text);
FieldByName('no_telp').Value := Trim(no_telp.Text);
FieldByName('role_ID').Value := Trim(role_ID.Text);

Post;
end;
Refresh;
ShowMessage('Data Karyawn Berhasil tersimpan');
bersih;
edt:=false;
end;
end;

procedure TForm_entryDataKaryawan.FormCreate(Sender: TObject);

```

```

begin
    ADOdtKrywn.Active := True;
end;

procedure TForm_entryDataKaryawan.btn_tambahClick(Sender: TObject);
begin
    bersih;
end;

procedure TForm_entryDataKaryawan.DBGkrywnCellClick(Column: TColumn);
var
    jnsKlmin : String;
begin
    edt := True;
    with ADOdtKrywn do begin
        NIK.Text := Trim(FieldValues['nik']);
        nama.Text := Trim(FieldValues['nama_karyawan']);
        tempat_lahir.Text := Trim(FieldValues['tmp_lahir']);
        tgl_lahir.Date := ADOdtKrywnagl_lahir.Value;
    end;
    jnsKlmin := FieldValues['jenis_kelamin'];
    if jnsKlmin = 'L' then
        Radio1.ski.Checked := True
    else
        Radio1Perempuan.Checked := True;
    end;
    alamat.Text := Trim(FieldValues['alamat']);
    kota.Text := Trim(FieldValues['kota']);
    no_telp.Text := Trim(FieldValues['no_telp']);
    role_ID.Text := Trim(FieldValues['role_ID']);
end;

end;

procedure TForm_entryDataKaryawan.btn_userIDClick(Sender: TObject);
begin
    frm_userID.Show;
end;

function TForm_entryDataKaryawan cekID(ID: string): boolean;
begin
    if Not ADOdtKrywn Locate('nik', ID, [loPartialKey, loCaseInsensitive]) then
        Result := False
    else
        Result := True;
    end;
end;

```

```

procedure TForm_entryDataKaryawan.btn_hapusClick(Sender: TObject);
begin
    ADOdtKrywn.Locate('nik', NIK.Text, [loPartialKey, loCaseInsensitive]);
    if (Application.MessageBox('Anda Yakin menghapus nama Karyawan ini?', 'Peringatan', MB_YESNO or MB_ICONQUESTION) = IDYES) then begin
        ADOdtKrywn.Delete;
        edt := False;
    end;
    refresh;
    bersih;
    tgl_lahir.Date := Now;
    edt := False;
end;

end;

```