

**RANCANG BANGUN *PROTOTYPE AUTOMATIC TERMINAL
INFORMATION SERVICE (P-ATIS)* BERBASIS *PERSONAL
COMPUTER (PC)* DI BANDARA RADIN INTEN II LAMPUNG**

(Skripsi)

Oleh

TAUFAN KURNIAWAN



**FAKULTAS TEKNIK
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2010**

ABSTRACT

DESIGN OF PROTOTYPE AUTOMATIC TERMINAL INFORMATION SERVICE (P-ATIS) THAT HAS PERSONAL COMPUTER (PC) BASE AT RADIN INTEN II AIRPORT LAMPUNG

By

TAUFAN KURNIAWAN

In flight field, there are many equipments which is used to support the operation of flight. One of them is Automatic Terminal Information Service (ATIS). ATIS is an equipment to support the flight smoothness that has a main function to inform recent weather. This facility result an output in audio form which informs the weather. Conventional ATIS has the form of big modules.

The construction plan of ATIS facilities which is applied in this research is engineer a personal base application that has a same function as conventional ATIS. Generally this application will be operated by BMG and BO, both unit will be connected with client/server base with server side of BMG unit. This application uses MySQL database, VB6.0 programming language and ODBC to access server database. This application is engineered by using waterfall method. The sound which is resulted by this application using additional library from VB, it is library text to speech.

These application tests include GUI test and client/server test. After passing the steps of those tests, this ATIS application that has personal computer base is ready to be implemented. This ATIS can be a choice to substitute the existing conventional ATIS.

Keyword : ATIS, *client/server, text to speech, Waterfall*

ABSTRAK

RANCANG BANGUN *PROTOTYPE AUTOMATIC TERMINAL INFORMATION SERVICE (P-ATIS)* BERBASIS *PERSONAL COMPUTER (PC)* DI BANDARA RADIN INTEN II LAMPUNG

Oleh

TAUFAN KURNIAWAN

Dalam dunia penerbangan terdapat banyak peralatan-peralatan yang digunakan untuk menunjang beroperasinya suatu penerbangan. Salah satu peralatan itu adalah *Automatic Terminal Information Service (ATIS)*. ATIS merupakan suatu alat penunjang kelancaran penerbangan yang mempunyai fungsi utama menginformasikan informasi cuaca terkini. Fasilitas ini menghasilkan keluaran berbentuk suara yang menginformasikan informasi cuaca. ATIS konvensional berbentuk modul-modul yang sangat besar.

Rancang bangun fasilitas ATIS yang telah dilakukan dalam penelitian ini adalah merencanakan sebuah aplikasi berbasis *personal computer* yang memiliki fungsi yang sama dengan ATIS konvensional. Aplikasi ini secara umum akan dioperasikan oleh unit BMG dan BO, kedua unit ini akan saling terkoneksi dengan basis *client/server*, dengan sisi *server* pada unit BMG. Aplikasi ini menggunakan *database* MySQL dan bahasa pemrograman VB 6.0, serta ODBC untuk mengakses *database server*. Aplikasi ini direkayasa menggunakan metode *waterfall*. Suara yang dihasilkan aplikasi ini menggunakan *library* tambahan dari VB yaitu *library text to speech*.

Pengujian aplikasi ini meliputi pengujian GUI dan pengujian *client/server*. Setelah melewati tahap pengujian tersebut, aplikasi ATIS berbasis *personal computer* ini siap untuk diimplementasikan. ATIS berbasis *personal computer* ini dapat menjadi pilihan untuk menggantikan ATIS konvensional yang sudah ada.

Kata Kunci : ATIS, *client/server*, *text to speech*, *Waterfall*

**RANCANG BANGUN *PROTOTYPE AUTOMATIC TERMINAL
INFORMATION SERVICE (P-ATIS)* BERBASIS *PERSONAL
COMPUTER (PC)* DI BANDARA RADIN INTEN II LAMPUNG**

Oleh

TAUFAN KURNIAWAN

Skripsi

Sebagai Salah Satu Syarat untuk Mencapai Gelar
SARJANA TEKNIK

Pada

**Jurusan Teknik Elektro
Fakultas Teknik Universitas Lampung**



**FAKULTAS TEKNIK
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG**

2010

Judul Skripsi : **Rancang Bangun *Prototype Automatic Terminal Information Service (P-ATIS)* Berbasis *Personal Computer (PC)* Di Bandara Radin Inten II Lampung**

Nama Mahasiswa : **Taufan Kurniawan**

No. Pokok Mahasiswa : 0415031088

Program Studi : Teknik Elektro

Fakultas : Teknik

MENYETUJUI

1. Komisi Pembimbing

Yulianto Raharjo, S.T., M.T.
NIP 196607131999031002

Mardiana., S.T., M.T.
NIP 197203161999032002

2. Ketua Jurusan Teknik Elektro

Ir . Abdul Haris, M.T.
NIP 196308011996031001

MENGESAHKAN

1. Tim Penguji

Ketua : **Mardiana, S.T., M.T.**

Sekretaris : **Yulianto Raharjo, S.T., M.T.**

Penguji
Bukan Pembimbing : **Wahyu Eko Sulistiono, S.T., M.Sc**

2. Dekan Fakultas Teknik Universitas Lampung

Dr. Ir. Lusmeilia Afriani, D.E.A
NIP 196505101993032008

Tanggal Lulus Ujian Skripsi : 26 Juli 2010

RIWAYAT HIDUP



Penulis adalah anak tunggal dari pasangan H. Nawawi (Alm) dan Titin Haryati. Penulis dilahirkan di Jakarta pada tanggal 21 bulan Juni tahun 1986.

Pendidikan formal penulis dimulai dari Sekolah Dasar Negeri (SDN) Karawaci Baru 1 Tangerang, yang diselesaikan pada tahun 1998. Sekolah Lanjutan Tingkat Pertama Negeri (SLTPN) 1 Tangerang, tamat pada tahun 2001. Sekolah Menengah Umum Negeri (SMUN) 2 Tangerang, lulus pada tahun 2004. Mulai tahun 2004 penulis terdaftar sebagai mahasiswa Jurusan Teknik Elektro Fakultas Teknik Universitas Lampung melalui Seleksi Penerimaan Mahasiswa Baru (SPMB).

Penulis memilih konsentrasi Teknik Komputer sebagai konsentrasi perkuliahan. Selama menjadi mahasiswa penulis pernah aktif di Himpunan Mahasiswa Teknik Elektro Universitas Lampung (Himatro Unila) sebagai anggota Divisi Olahraga (2004-2005) dan anggota Divisi Minat dan Kreatifitas (2005-2006). Semasa kuliah juga penulis juga sempat aktif menjadi asisten di Laboratorium Teknik Telekomunikasi. Pada tahun 2006, penulis melakukan kerja praktek di Bandara Radin Inten II Lampung pada Divisi Telekomunikasi dan Navigasi.

*Kupersembahkan karya ini untuk
Kedua Orang tua ku tercinta, Ayahanda (Alm) dan Ibunda
tercinta, yang dengan penuh kesabaran, keikhlasan dan
ketulusan senantiasa melindungiku, menyayangiku dan
mendidikku sejak aku lahir dan membuka mata untuk pertama
kali ke dunia ini tanpa mengenal lelah setitikpun.*

“Wahai golongan jin dan manusia, jika kamu sanggup menembus (melintasi) penjuru langit dan bumi, maka lintasilah, kamu tidak dapat menembusnya kecuali dengan kekuatan.”

(Ar-Rahman: 33)

“Niscaya Allah akan meninggikan orang-orang yang beriman di antaramu dan orang-orang yang diberi ilmu pengetahuan beberapa derajat.”

(Al Mujadilah: 11)

“Tinta bagi seorang penuntut ilmu lebih suci nilainya daripada darah seorang martir.”

(Nabi Muhammad SAW)

SANWACANA

Assalamualaikum Wr.Wb.

Puji syukur kehadirat Allah SWT, karena atas segala rahmat, hidayah, serta nikmat-Nya jualah, penulis akhirnya dapat menyelesaikan skripsi dengan judul **“Rancang Bangun *Prototype Automatic Terminal Information Service (P-ATIS)* Berbasis *Personal Computer (PC)* Di Bandara Radin Inten II Lampung”**.

shalawat dan salam selalu terlimpahkan kepada Baginda Rasulullah Muhammad SAW, kepada keluarga dan para sahabatnya serta pengikutnya yg menetapi petunjuk dan garis ketetapan yang sudah dibawa oleh beliau hingga akhir zaman, Amien.

Penulis menyadari bahwa karya ini masih jauh dari sempurna, tetapi, penulis dengan penuh kerendahan hati berharap semoga karya ini dapat menjadi tambahan ilmu yang bermanfaat bagi siapa saja yang menggunakannya, Amien. Penulis juga menyadari, tanpa adanya bantuan dan dukungan, baik moril maupun materil dari keluarga, sahabat dan pihak-pihak yang turut membantu, maka penulis tentu tidak akan dapat menyelesaikan karya ini dengan baik.

Dalam kesempatan ini penulis juga mengucapkan terima kasih kepada :

1. Bapak Prof. Dr. Ir. Sugeng P. Harianto, M.S. selaku Rektor Universitas Lampung
2. Ibu Dr. Ir. Lusmeilia Afriani, D.E.A. selaku Dekan Fakultas Teknik Universitas Lampung
3. Bapak Ir. Abd. Haris, M.T. selaku Ketua Jurusan Teknik Elektro Universitas Lampung.
4. Ibu Mardiana, S.T.,M.T., Selaku pembimbing utama skripsi yang telah memberikan arahan, saran, nasihat serta dorongan semangat dalam penyelesaian skripsi ini.
5. Bapak Yulianto Raharjo, S.T., M.T., Selaku pembimbing pendamping skripsi dan pembimbing akademik yang telah memberikan nasihat, arahan, saran, nasihat serta dorongan semangat dalam penyelesaian skripsi ini dan selama penulis menempuh kuliah di Jurusan Teknik Elektro Universitas Lampung.
6. Bapak Wahyu Eko Sulistiono, S.T., M.Sc., Selaku penguji skripsi yang di sela kesibukan beliau bersedia meluangkan waktunya untuk menjadi penguji dari skripsi ini.
7. Bapak Lukmanul Hakim, S.T., M.Sc., yang memberikan nasihat dan dorongan semangat serta sahabat diskusi.
8. Mbak Ning, atas bantuannya dalam mengurus masalah administrasi selama pennulis menjadi mahasiswa.
9. Seluruh staff pengajar di Jurusan Teknik Elektro Universitas Lampung yang telah memberikan bimbingan dan ilmu yang bermanfaat selama penulis menuntut ilmu.

10. Rekan-rekan di Mamank Entertainment 04 : Deddy “Mamank”, Deny “Djum”, Agus “Centenk”, Redi “Tile”, Galih “Wedhus”, Eric “Golox”, Abay “Badak”, Rio “Ijonx Bangbross”, Fisra “Ratu Lebay”.
11. *Someone Special* “ Wahyuni Surakhman,.A.Md., yang telah memberikan dorongan semangat, waktu, perhatian dan kesabaran yang amat dalam.
12. Para punggawa teknik komputer, Ammi, Boim, Eko, Lesa, Agoy, Bagince, Okta, Beny.
13. Akhwat-akhwat Teknik Elektro 2004, Etika, Eka, Dwi Ferdiana, Ratna.
14. Keluarga Firmansyah Miolo, S.E, Om Firman, Tante Tuti, Gabriel, Mikhael, untuk saran dan dorongan semangatnya.
15. Segenap karyawan dan pegawai yang membantu segala administrasi selama penulis menjadi mahasiswa di Jurusan Teknik Elektro.
16. Semua sahabat-sahabat di teknik elektro (tanpa mengenal angkatan) terima kasih untuk bantuan dan semangat kekeluargaannya
17. Semua pihak yang telah membantu dan memberikan dukungannya dalam penyelesaian laporan Tugas Akhir ini yang tidak dapat disebutkan satu persatu.

Semoga Allah SWT membalas semua amal baiknya. Penulis berharap tugas akhir ini berguna dan menjadi ilmu yang bermanfaat bagi kita semua, Amien.

Bandar Lampung, 26 Juli 2010

Penulis

Taufan Kurniawan

DAFTAR ISI

	Halaman
DAFTAR ISI	i
DAFTAR GAMBAR	ii
DAFTAR TABEL	iii
DAFTAR ISTILAH	iv
I. PENDAHULUAN.....	1
A. Latar Belakang	1
B. Tujuan Penelitian.....	3
C. Manfaat Penelitian	3
D. Batasan Masalah.....	4
E. Perumusan Masalah	4
F. Hipotesis	5
G. Sistematika Pembahasan	6
II. TINJAUAN PUSTAKA.....	7
A. <i>Automatic Terminal Information Service (ATIS)</i>	7
1. Frekuensi ATIS.....	8
2. Digital ATIS.....	9
3. Informasi Yang Dipancarkan ATIS.....	11
4. Pemancar VHF Yang Digunakan ATIS.....	12

a. Penempatan Pemancar VHF Di Luar Area Bandar Udara.....	13
b. Penempatan Pemancar VHF Di Luar Area Bandar Udara.....	14
B. Sistem Informasi.....	16
1. Konsep Dasar Sistem Informasi.....	16
a. Komponen Sistem Informasi.....	16
2. Tingkatan Sistem Informasi.....	18
3. Kategori Sistem Terotomasi.....	20
C. Konsep <i>Client-Server</i>	21
1. <i>Client</i>	21
2. <i>Server</i>	22
3. Jenis Koneksi Dan Interaksi.....	22
D. <i>Local Area Network (LAN)</i>	24
1. Pengertian <i>Local Area Network (LAN)</i>	24
2. Kategori Kecepatan Transmisi Data Dalam LAN.....	24
E. Topologi LAN.....	25
1. Macam-macam Bentuk Konfigurasi Topologi Jaringan.....	25
2. Kelebihan Dan Kelemahan Masing-Masing Topologi.....	27
F. Modulasi Yang Digunakan ATIS.....	28
G. Visual Basic 6.0.....	29
1. Antar Muka Visual Basic 6.0.....	30
2. Aturan Pemrograman Visual Basic 6.0.....	31
H. <i>Open Database Conectivity (ODBC)</i>	33
I. <i>Structured Query Language (SQL)</i>	33
1. <i>Structrured Query Language (SQL)</i> dengan <i>MySQL</i>	34

2. Keunggulan <i>MySQL</i>	34
3. Tipe Data pada <i>MySQL</i>	35
J. <i>SQLyog Enterprises</i>	36
K. Metode <i>Waterfall</i>	37
L. Elemen Model Analisis	39
1. <i>Entity Relationship Diagram</i> (ERD)	41
2. <i>Data Context Diagram</i> (DCD)	42
3. <i>Data Flow Diagram</i> (DFD)	43
4. <i>Specification Control</i> (CSPEC)	48
5. <i>Process Specification</i> (PSPEC)	48
6. <i>Data Dictionary</i>	49
M. Model Desain	49
1. Desain Data	50
2. Desain Arsitektur	50
3. Desain <i>Interface</i>	50
N. <i>Text To Speech</i>	51
III. METODE PENELITIAN	54
A. Waktu dan Tempat Penelitian	54
B. Alat dan Bahan	54
C. Prosedur Kerja	55
1. Studi Literatur	55
2. Pembuatan Sistem	55
3. Pengujian Sistem	55

D. Perancangan Perangkat Lunak P-ATIS	57
IV. HASIL DAN PEMBAHASAN	59
A. Perancangan.....	59
1. Tahap <i>System Engineering</i>	59
a. Ruang Lingkup (<i>Scope</i>) Perangkat Lunak P-ATIS	59
b. Dekomposisi Ruang Lingkup.....	60
2. Tahap <i>Requirement Analysis</i>	62
a. Kebutuhan Fungsional	62
b. Deskripsi Proses	63
3. Kebutuhan Antarmuka Eksternal	70
a. Antarmuka Pemakai	70
b. Antarmuka Perangkat Keras	70
c. Antarmuka Perangkat Lunak	70
d. Antarmuka Komunikasi	71
e. Kebutuhan Non-Fungsional	71
f. Batasan Perancangan	71
4. Pemodelan Proses	71
a. Data Context Diagram (DCD)	72
b. <i>Data Flow Diagram</i> (DFD)	73
c. <i>Proses Spesification</i> (P-SPEC)	78
d. <i>Entity Relationship Diagram</i> (ERD)	117
e. Kamus Data	118
3. Tahap <i>Design</i>	131
a. Desain <i>Database</i>	131

b. Desain <i>User Interface</i>	133
c. Desain Arsitektur Perangkat Lunak	136
4. Tahap <i>Coding</i>	138
a. Proses Otentifikasi	138
b. Proses Menampilkan Menu	140
c. Proses Keluar Dari Aplikasi	142
d. Kode Program Masing-Masing <i>User</i>	143
B. Pengujian	156
1. Otentifikasi <i>User</i>	156
2. Menu <i>User</i>	158
3. Olah Informasi	171
V. SIMPULAN DAN SARAN	209
A. Simpulan.....	209
B. Saran.....	210

DAFTAR PUSTAKA

LAMPIRAN

DAFTAR GAMBAR

Gambar	Halaman
1.1. ATIS	2
2.1. Pengendali VOR Di Pesawat	9
2.2. ATIS Digital	10
2.3. Luas Lahan Dan Lokasi Penempatan VOR	14
2.4. Syarat Batas Ketinggian Dan Benda Tumbuh Di Sekitar VOR	14
2.5. Penempatan Fasilitas VHF A/G dan sensor kecepatan angin	15
2.6. Blok Sistem Sistem Informasi yang berinteraksi	16
2.7. Topologi <i>Bus</i>	25
2.8. Topologi <i>Ring</i>	26
2.9. Topologi <i>Star</i>	26
2.10. Topologi <i>Mesh</i>	27
2.11. Sinyal Informasi Yang Telah Dimodulasi AM	28
2.12. Antar Muka Visual Basic 6.0	30

2.13. Komponen Standar Dalam <i>Toolbox</i>	31
2.14. Komponen Properti Dalam VB 6.0	32
2.15. Event Dalam VB 6	32
2.16. <i>Setting</i> ODBC Di Windows	33
2.17. Antar Muka SQLyog	37
2.18. Metode <i>Waterfall</i>	37
2.19. Struktur Model Analisis	40
2.20. ERD <i>One to one</i>	41
2.21. <i>One to many</i>	41
2.22. <i>Many to one</i>	41
2.23. <i>Many to many</i>	41
2.24. <i>Data Context Diagram</i>	43
2.25. Komponen DFD	43
2.26. Entitas luar	44
2.27. Proses	44
2.28. Proses <i>Write</i>	45
2.29. Proses <i>Read</i>	45

2.30. Proses <i>Read and Write</i>	46
2.31. <i>Packets of Data</i>	46
2.32. <i>Diverging Data Flow</i>	47
2.33. <i>Converging data flow</i>	47
2.34. Dari Proses Ke Bukan Proses	47
2.35. Dari Bukan Proses Ke Proses	48
2.36. Dari Proses Ke Proses	48
2.37. Model Desain	49
2.38. Blok Diagram <i>Text To Speech</i>	51
3.1. Diagram Aplikasi Secara Umum	56
3.2. Blok Diagram Prosedur Kerja Aplikasi	57
3.3. Diagram Alir Pembuatan Perangkat Lunak	58
4.1. <i>Data Context Diagram (DCD)</i>	72
4.2. DFD Level 1 P-ATIS	73
4.3. DFD Level 2 Proses 1 P-ATIS	74
4.4. DFD Level 2 Proses 2 P-ATIS	74
4.5. DFD Level 2 Proses 3 P-ATIS	75

4.6. DFD Level 3 Proses 3.2 P-ATIS (Aliran Proses Kepala Unit BMG)	76
4.7. DFD Level 3 Proses 3.3 P-ATIS (Aliran Proses Kepala Unit BO)	76
4.8. DFD Level 3 Proses 3.4 P-ATIS (Aliran Proses Pengguna Unit BMG)	77
4.9. DFD Level 3 Proses 3.5 P-ATIS (Aliran Proses Pengguna Unit BO)	77
4.10. DFD Level 3 Proses 3.6 P-ATIS (Aliran Proses <i>Supervisor</i>)	78
4.11. <i>Entity Relationship Diagram</i> (ERD) P-ATIS	117
4.12. Proses Input Data Pengguna BMG	118
4.13. Proses Edit Data Pengguna BMG	119
4.14. Proses Hapus Data Pengguna BMG	120
4.15. Proses Input Data Pengguna Unit BO	120
4.16. Proses Edit Data Pengguna BO	122
4.17. Proses Hapus Data Pengguna BO	122
4.18. Proses Input Data Global <i>Supervisor</i>	123
4.19. Proses Input Data Kepala Unit <i>Supervisor</i>	124
4.20. Proses Input Data Hak Akses <i>Supervisor</i>	126
4.21. Proses Olah Data Cuaca Pengguna Unit BMG	127
4.22. Proses Simpan Informasi METAR Pengguna Unit BO	129

4.23. <i>Database Relationship</i> P-ATIS	133
4.24. <i>Desain User Interface</i> Kelola Kepala Unit	134
4.25. <i>Desain User Interface</i> Kelola Pengguna Unit BMG	134
4.26. <i>Desain User Interface</i> Kelola Pengguna Unit BO	135
4.27. <i>Desain User Interface</i> Kelola Data Cuaca	135
4.28. <i>Desain User Interface</i> Kelola Data METAR	136
4.29. <i>Desain User Interface</i> Aplikasi <i>Chatting</i>	136
4.30. Struktur Hirarki Validasi Eksekutif	136
4.31. Struktur Hirarki Menampilkan Menu Eksekutif	137
4.32. Struktur Hirarki Olah Informasi Eksekutif	137
4.33. Cuplikan Kode Program Proses Otentifikasi <i>User</i>	138
4.34. Cuplikan Kode Program Kesalahan <i>User ID</i> Atau <i>Password</i>	139
4.35. Cuplikan Kode Program <i>User</i> Di Non-Aktifkan	139
4.36. Cuplikan Kode Program <i>Update</i> Informasi Status	140
4.37. Cuplikan Kode Program Mencari Form Menu <i>User</i>	141
4.38. Cuplikan Kode Program Tampil Menu <i>User</i>	142
4.39. Cuplikan Kode Program Keluar Dari Aplikasi	142

4.40. Cuplikan Kode Program Masukan Data Global Setup	143
4.41. Cuplikan Kode Validasi Data Global Setup	143
4.42. Cuplikan Kode Program Pembatasan Karakter Global Setup	143
4.43. Cuplikan Kode Program Simpan Dan Edit Data Kepala Unit	144
4.44. Cuplikan Kode Program Hapus Data Kepala Unit.....	144
4.45. Cuplikan Kode Program Validasi Data Kepala Unit.....	144
4.46. Cuplikan Kode Program Simpan Data Hak Akses.....	145
4.47. Cuplikan Kode Program <i>Reload</i> Data Hak Akses.....	145
4.48. Cuplikan Kode Program Realisasi Kamus Data Proses 3.2.2	146
4.49. Cuplikan Kode Validasi Data Pengguna Unit BMG	146
4.50. Cuplikan Kode Simpan Dan Edit Data Pengguna Unit BMG.....	146
4.51. Cuplikan Kode Program Hapus Data Pengguna Unit BMG.....	147
4.52. Cuplikan Kode Program Realisasi Kamus Data Proses 3.5.2.....	147
4.53. Cuplikan Kode Validasi Data Pengguna Unit BO.....	147
4.54. Cuplikan Kode Simpan Dan Edit Data Pengguna Unit BO.....	148
4.55. Cuplikan Kode Program Hapus Data Pengguna Unit BO.....	148
4.56. Cuplikan Kode Program Pembatasan Karakter Data Cuaca.....	148

4.57. Cuplikan Kode Validasi Data Cuaca.....	149
4.58. Cuplikan Kode Program Simpan Data Cuaca.....	149
4.59. Cuplikan Kode Program Update Atribut Active Data Cuaca.....	149
4.60. Cuplikan Kode Program Kirim Pesan <i>Chat</i> Pengguna Unit BM.....	150
4.61. Cuplikan Kode Program Terima Pesan <i>Chat</i> Pengguna Unit BMG.....	150
4.62. Cuplikan Kode Program Membuat Laporan Harian Data Cuaca.....	151
4.63. Cuplikan Kode Membuat Laporan Periodik Data Cuaca.....	151
4.64. Cuplikan Kode Program Pembatasan Karakter Data METAR.....	152
4.65. Cuplikan Kode Validasi Data Cuaca.....	152
4.66. Cuplikan Kode Program Simpan Data METAR.....	152
4.67. Cuplikan Kode Program Kirim Pesan <i>Chat</i> Pengguna Unit BO.....	153
4.68. Cuplikan Kode Program Terima Pesan <i>Chat</i> Pengguna Unit BO.....	153
4.69. Cuplikan Kode Program <i>ATIS Engine</i>	153
4.70. Cuplikan Kode Program Membuat Laporan Harian Data METAR.....	154
4.71. Cuplikan Kode Membuat Laporan Periodik Data METAR.....	154
4.72. Cuplikan Kode Program Perekaman Suara.....	155
4.73. Tampilan Form Otentifikasi <i>User</i>	156

4.74. Tampilan Pesan Kesalahan <i>User ID</i> Atau <i>Password</i>	157
4.75. Tampilan Pesan <i>User</i> Di Non-Aktifkan.....	158
4.76. Tampilan Menu Admin.....	158
4.77. Tampilan Sub-Menu File Admin.....	159
4.78. Tampilan Sub-Menu Utama Supervisor Admin.....	160
4.79. Tampilan Menu Kepala Unit BMG.....	160
4.80. Tampilan Sub-Menu File Kepala Unit BMG.....	161
4.81. Tampilan Sub-Menu Utama Master Data KepalaUnit BMG.....	162
4.82. Tampilan Sub-Menu Utama User Online KepalaUnit BMG.....	162
4.83. Tampilan Menu Kepala Unit BO.....	163
4.84. Tampilan Sub-Menu File Kepala Unit BO.....	164
4.85. Tampilan Sub-Menu Utama Master Data Kepala Unit BO.....	164
4.86. Tampilan Sub-Menu Utama User Online Kepala Unit BO.....	165
4.87. Tampilan Menu Pengguna Unit BMG.....	165
4.88. Tampilan Sub-Menu Utama File Pengguna Unit BMG.....	166
4.89. Tampilan Sub-Menu Utama Master Data Pengguna Unit BMG.....	167
4.90. Tampilan Sub-Menu Utama Laporan Pengguna Unit BMG.....	167

4.91. Tampilan Sub-Menu Utama User Online Pengguna Unit BMG.....	168
4.92. Tampilan Menu Pengguna Unit BO.....	168
4.93. Tampilan Sub-Menu Utama File Pengguna Unit BO.....	169
4.94. Tampilan Sub-Menu Utama Master Data Pengguna Unit BO.....	170
4.95. Tampilan Sub-Menu Utama Laporan Pengguna Unit BO	170
4.6 Tampilan Sub-Menu Utama Record Pengguna Unit BO	171
4.97. Tampilan Sub-Menu Utama User Online Pengguna Unit BO	171
4.98. Tampilan Sub-Menu Global Setup.....	172
4.9 Data Global Bandara Yang Dimasukan Tidak Valid.....	172
4.100. Tampilan Sub-Menu Set Kepala Unit.....	173
4.101. Data Kepala Unit Yang Dimasukan Tidak Valid.....	174
4.102. Konfirmasi Penyimpanan Data Kepala Unit.....	174
4.103. Tampilan Form Set <i>User</i> Menu.....	175
4.104. Tampilan Sub-Menu Set User BMG.....	176
4.105. Data Pengguna Unit BMG Yang Dimasukan Tidak Valid.....	177
4.106. Konfirmasi Penyimpanan Data Pengguna Unit BMG.....	177
4.107. Tampilan Form Edit Dan Hapus Data Pengguna Unit BMG.....	178

4.108. Tampilan Form Edit Data Pengguna Unit BMG.....	178
4.109. Tampilan Form Hapus Data Pengguna Unit BMG.....	179
4.110. Tampilan Sub-Menu Data Cuaca.....	179
4.111. Tampilan Sub-Menu Pengguna Unit BMG Online.....	180
4.112. Tampilan Sub-Menu Set User BO.....	180
4.113. Data Pengguna Unit BO Yang Dimasukan Tidak Valid.....	181
4.114. Konfirmasi Penyimpanan Data Pengguna Unit BO.....	181
4.115. Tampilan Form Edit Dan Hapus Data Pengguna Unit BO.....	182
4.116. Tampilan Form Edit Data Pengguna Unit BO.....	183
4.117. Tampilan Form Hapus Data Pengguna Unit BO.....	183
4.118. Tampilan Sub-Menu Data METAR.....	183
4.119. Tampilan Sub-Menu Pengguna Unit BO Online.....	184
4.120. Tampilan Sub-Menu Set Data Cuaca.....	184
4.121. Tampilan <i>Field</i> Set Data Cuaca.....	185
4.122. Data Cuaca Yang Dimasukan Tidak Valid.....	186
4.123. Konfirmasi Penyimpanan Data Cuaca.....	186
4.124. <i>Message Box</i> Pembatalan Penyimpanan Data Cuaca.....	187

4.125. Tampilan <i>Record</i> Data Cuaca Sub-Menu Set Data Cuaca.....	187
4.126. Tampilan Fasilitas <i>Chatting</i> Pengguna Unit BMG.....	188
4.127. Informasi Status Koneksi Pengguna Unit BMG Tidak Terkoneksi.....	188
4.128. Informasi Status Koneksi Pengguna Unit BMG Mencari Koneksi.....	189
4.129. Informasi Status Koneksi Pengguna Unit BMG Telah Terkoneksi.....	189
4.130. Aplikasi Penghitung Durasi Koneksi Pengguna Unit BMG.....	189
4.131. Pesan Komunikasi Pengguna Unit BMG.....	189
4.132. Informasi Komunikasi Dengan Pengguna Unit BO Terputus.....	190
4.133. Tampilan Sub-Menu Data Cuaca Pengguna Unit BMG.....	190
4.134. Tampilan Sub-Menu Laporan Harian Data Cuaca.....	191
4.135. Tampilan Memilih Tanggal Data Cuaca.....	191
4.136. Bentuk Laporan Harian Data Cuaca.....	191
4.137. Pesan Kesalahan Harian Data Cuaca Tidak Ditemukan.....	192
4.138. Tampilan Sub-Menu Laporan Periodik Data Cuaca.....	192
4.139. Bentuk Laporan Periodik Data Cuaca.....	193
4.140. Pesan Kesalahan Periode Data Cuaca Tidak Ditemukan.....	193
4.141. Tampilan Sub-Menu Pengguna Unit BO Online.....	194

4.142. Tampilan Sub-Menu Set Data METAR.....	194
4.143. Tampilan Tampilan <i>Field</i> Set Data METAR.....	195
4.144. Data METAR Yang Dimasukan Tidak Valid.....	196
4.145. Konfirmasi Penyimpanan Data METAR.....	196
4.146. <i>Message Box</i> Pembatalan Penyimpanan Data METAR.....	196
4.147. Tampilan <i>Record</i> Data Cuaca Sub-Menu Set Data METAR.....	197
4.148. Tampilan Fasilitas <i>Chatting</i> Pengguna Unit BO.....	197
4.149. Informasi Status Koneksi Pengguna Unit BO Tidak Terkoneksi.....	198
4.150. Informasi Status Koneksi Pengguna Unit BO Telah Terkoneksi.....	198
4.151. Aplikasi Penghitung Durasi Koneksi Pengguna Unit BO.....	199
4.152. Pesan Komunikasi Pengguna Unit BO.....	199
4.153. Informasi Komunikasi Dengan Pengguna Unit BMG Terputus.....	199
4.154. Informasi Komunikasi Gagal Terkoneksi.....	200
4.155. Fasilitas ATIS.....	200
4.156. ATIS Siap Membaca Teks Data Cuaca.....	201
4.157. Fasilitas ATIS Dihentikan.....	201
4.158. Tampilan Sub-Menu Data Cuaca Pengguna Unit BO.....	202

4.159. Tampilan Sub-Menu Data METAR.....	202
4.160. Tampilan Sub-Menu Laporan Harian Data METAR.....	202
4.161. Tampilan Memilih Tanggal Data METAR.....	203
4.162. Bentuk Laporan Harian Data METAR.....	203
4.163. Pesan Kesalahan Harian Data METAR Tidak Ditemukan.....	204
4.164. Tampilan Sub-Menu Laporan Periodik Data METAR.....	204
4.165. Bentuk Laporan Periodik Data METAR.....	204
4.166. Pesan Kesalahan Periode Data METAR Tidak Ditemukan.....	205
4.167. Tampilan Sub-Menu Rekam Suara.....	205
4.168. Penyimpanan Rekaman Suara.....	207
4.169. Mengambil Rekaman Suara.....	207
4.170. Tampilan Sub-Menu Pengguna Unit BMG Online.....	208

DAFTAR TABEL

Tabel	Halaman
2.1. Klasifikasi <i>Height Of Clouds</i>	11
2.2. Spektrum Elektromagnetik	12
2.3. Syarat Penempatan Pemancar Di Luar Area Banda	13
2.4. Syarat Penempatan Pemancar Di Dalam Area Bandara.....	15
2.5. Kelebihan Dan Kelemahan Topologi Jaringan	27
4.1. Kategori Pengguna	62
4.2. Kebutuhan Non-Fungsional	71
4.3. Tabel USER	131
4.4. Tabel GLOBAL	131
4.5. Tabel MENU	131
4.6. Tabel METAR	132
4.7. Tabel CUACA	132

DAFTAR ISTILAH

ATIS : Merupakan akronim dari *Automatic Terminal Information Service*, yaitu fasilitas penunjang penerbangan yang menginformasikan cuaca terkini secara terus-menerus.

BMG : Merupakan akronim dari Badan Meteorologi dan Geofisika, yaitu instansi yang bertugas mengumpulkan data-data cuaca dan fenomena alam yang terjadi.

BO : Merupakan akronim dari *Briefing Office*, yaitu suatu unit kerja dalam suatu bandara yang bertugas menjadi pusat pengumpulan data-data penerbangan.

ATC : Unit kerja di bandara yang bertugas mengatur lalu-lintas penerbangan.

VOR : Pemancar frekuensi di bandara

METAR : Menunjukkan kapan informasi ATIS diperbaharui. Diinisialkan dalam kode isyarat.

Visibility : Jarak pandang terbaik untuk melakukan *take-off* dan *landing*.

DEW Point : Titik pengembunan

QNH : Ketinggian daratan dari permukaan laut

Runway : Jalur yang digunakan pesawat terbang untuk melakukan *take-off* dan *landing*.

Wind Direction : Arah angin..

Wind Speed : Kecepatan angin.

Height Of Clouds : Ketinggian awan di sekitar lapangan terbang.

I. PENDAHULUAN

A. Latar Belakang Masalah

Transportasi udara adalah salah satu jenis transportasi yang sangat efektif bagi konsumen, karena dapat melakukan perjalanan yang jauh hanya dalam waktu yang relatif singkat jika dibandingkan dengan jenis transportasi lain. Oleh karena itu keselamatan penumpang adalah hal yang tidak boleh diabaikan. Untuk menjamin keselamatan penerbangan pemerintah menetapkan Undang-Undang RI No. 15 tahun 1992, pasal 3 yang berbunyi “Tujuan penerbangan adalah untuk mewujudkan penyelenggaraan penerbangan yang selamat, aman, cepat, lancar, tertib dan teratur, serta nyaman dan berdaya guna, dengan biaya yang terjangkau oleh daya beli masyarakat dengan mengutamakan dan melindungi penerbangan nasional untuk menunjang pemerataan, pertumbuhan, dan stabilitas, sebagai pendorong penggerak dan penunjang pembangunan nasional serta pemeratakan hubungan antar bangsa”. Untuk mencapai tujuan tersebut dalam pasal 20 dijelaskan bahwa setiap fasilitas ataupun peralatan penunjang penerbangan wajib memenuhi persyaratan keamanan dan keselamatan penerbangan.

Keselamatan dan keamanan penerbangan adalah semua kegiatan yang berkaitan dengan keselamatan dan keamanan operasi pesawat udara, baik di darat maupun di udara termasuk penumpang sehingga fasilitas elektronika dan listrik

penerbangan merupakan hal yang sangat perlu diperhatikan. Salah Satu fasilitas yang digunakan untuk menunjang kelancaran penerbangan adalah *Automatic Terminal Information Service* (ATIS). Perangkat ini memancarkan informasi secara otomatis tanpa henti. Informasi yang diberikan biasanya adalah informasi landasan yang digunakan, cuaca terkini, biasanya 1 jam atau 30 menit terakhir kecuali jika terjadi perubahan cuaca yang sangat tajam. Informasi lain yang penting juga sering diberikan, misalnya adanya *windshear*, penutupan jalur *taxyway* dan lain sebagainya. Sebelumnya atau pada bandara yang tidak terlalu sibuk, semua informasi itu diberikan oleh *Air Traffic Control* (ATC), dengan menggunakan komunikasi suara secara langsung. Tetapi seiring bertambahnya volume penerbangan dan keinginan dari pihak bandara untuk memperbaharui dan menambahkan fasilitas penunjang kelancaran penerbangan, maka ATIS sangat membantu untuk mengurangi kesibukan komunikasi di radio penerbangan dan dapat menjadi penunjang kelancaran penerbangan.



Gambar 1.1 ATIS

ATIS dipancarkan menggunakan pemancar VHF. Rentang frekuensi ATIS di Indonesia berada pada kisaran 108 – 138 MHz. Pemancar VHF yang digunakan haruslah memenuhi standar yang telah ditetapkan oleh Dinas Standarisasi dan Kelaikan Udara (DSKU). Modulasi yang digunakan ATIS dalam pengiriman informasi ke pesawat terbang adalah modulasi amplitudo. Ketika suatu bandara sudah menggunakan ATIS, maka unit yang bertanggung jawab dalam mengoperasikan ATIS adalah *Briefing Office* (BO), yaitu suatu unit yang bertanggung jawab dalam pengolahan data.

B. Tujuan Penelitian

Tujuan dari penulisan Tugas Akhir ini diantaranya adalah :

1. Merancang dan membangun *Prototype-ATIS* (P-ATIS) menggunakan bahasa pemrograman Visual Basic dan *database* MySQL dengan berbasis PC.
2. Membantu pihak Bandara Radin Inten II Lampung untuk menyediakan perangkat penunjang kelancaran penerbangan yang dalam hal ini adalah P-ATIS.

C. Manfaat Penelitian

Manfaat yang diharapkan pada penelitian dan penulisan Tugas Akhir ini di antaranya adalah :

1. Dapat membuat suatu perangkat penunjang kelancaran penerbangan dalam hal ini perangkat tersebut adalah ATIS dengan berbasis PC.

2. Dapat membantu meringankan tugas dari pihak Bandara Radin Inten II Lampung dalam mengatur kelancaran penerbangan.
3. Perangkat P-ATIS ini dapat terus di kembangkan oleh rekan-rekan lainnya guna dapat terciptanya suatu perangkat yang lebih baik dari versi awal ini.

D. Batasan Masalah

Dalam tugas akhir ini, penulis akan memberi batasan masalah hanya pada hal – hal berikut :

1. Aplikasi yang dibuat dalam bentuk *prototype*, aplikasi ini digunakan oleh dua unit, yaitu, Badan Meteorologi dan Geofisika (BMG) dan *Briefing Office* (BO).
2. Laporan hanya menjelaskan analisis, metode, tahap-tahap perancangan, dan implementasi perangkat lunak.
3. Tidak menjelaskan konsep pemrograman dengan menggunakan Visual Basic 6.0.
4. *Library* suara *text to speech* yang dipakai merupakan *library* suara yang telah tersedia untuk mesin Windows.

E. Perumusan Masalah

Pada penulisan tugas akhir ini penulis coba merancang suatu perangkat ATIS yang dapat beroperasi dalam suatu PC. Sebelumnya penginformasian data cuaca dilakukan oleh ATC saat pilot memintanya, namun, dikarenakan semakin bertambahnya maskapai penerbangan yang melayani rute penerbangan ke Bandara Radin Inten II ini maka permintaan data cuaca terkini akan semakin meningkat.

Kelemahan dalam menginformasikan data cuaca dengan cara manual seperti ini adalah :

1. Membutuhkan waktu yang relatif lama, karena petugas ATC bergiliran memberikan informasi cuaca terkini secara bergiliran kepada tiap maskapai penerbangan yang tengah mengudara.
2. Kemungkinan terjadinya kesalahan relatif besar, ini dikarenakan faktor konsentrasi dari petugas ATC yang kemungkinan dapat menurun.
3. Tidak efisien, karena diinformasikan secara bergiliran, maka ketika terjadi perubahan cuaca yang drastis, tiap maskapai yang tengah mengudara tidak mendapatkan kondisi cuaca terkini tersebut dengan cepat.

F. Hipotesis

Ada beberapa perkiraan atau hipotesis awal tentang hasil dari penelitian yang akan dilakukan oleh penulis, di antaranya adalah :

1. Bagaimana perangkat penunjang kelancaran penerbangan yang akan dibangun yaitu P-ATIS dapat dioperasikan pada suatu PC .
2. Bagaimana pengiriman data yang dilakukan dari BMG ke BO dilakukan dengan LAN.

G. Sistematika Pembahasan

Sistematika penulisan dan pembahasan Tugas Akhir ini adalah sebagai berikut :

1. BAB 1 PENDAHULUAN

Memuat latar belakang dan identifikasi masalah, ruang lingkup, pembahasan, hipotesis, tujuan dan manfaat penelitian, serta sistematika pembahasan Tugas Akhir.

2. BAB 2 TINJAUAN PUSTAKA

Pada bab ini dijelaskan dan dipaparkan secara garis besar tentang dasar teori yang digunakan dan yang berkaitan dengan sistem yang akan dirancang.

3. BAB 3 METODE PENELITIAN

Bagian ini akan menjelaskan metode penelitian yang digunakan dalam proses merancang P-ATIS.

4. BAB 4 HASIL DAN PEMBAHASAN

Bagian ini akan membahas mengenai hasil yang dihasilkan oleh sistem yang telah dirancang.

5. BAB 5 SIMPULAN DAN SARAN

Bab ini akan menyimpulkan semua kegiatan dan hasil-hasil yang diperoleh selama proses perancangan dan desain implementasi sistem.

II. TINJAUAN PUSTAKA

A. *Automatic Terminal Information Service (ATIS)*

ATIS adalah frekuensi radio yang memancarkan informasi secara otomatis tanpa henti. Informasi yang diberikan biasanya adalah informasi landasan yang digunakan, cuaca terkini, biasanya 1 jam atau 30 menit terakhir kecuali jika terjadi perubahan cuaca yang sangat tajam. Informasi lain yang penting juga sering diberikan, misalnya adanya *windshear*, penutupan jalur *taxyway* dan lain sebagainya informasi yang dibutuhkan pilot ¹⁾.

Sebelumnya atau juga pada bandar udara yang tidak sibuk, semua informasi itu diberikan oleh ATC, dengan komunikasi suara secara langsung. Tapi dengan bertambahnya *volume* penerbangan, maka ATIS sangat membantu untuk mengurangi kesibukan komunikasi di radio penerbangan.

ATIS merupakan suatu perangkat penunjang penerbangan yang tersusun dari beberapa modul perangkat elektronik, yaitu :

- a) Satu unit kabinet untuk menyimpan modul.
- b) VHF *transmitter*.
- c) *Power supply* dan baterai cadangan.
- d) Antena VHF.
- e) Perekam suara dan penginformasi ke pesawat.

Keterangan :

¹⁾ Sumber : www.ilmuterbang.com. 12 Desember 2008. Diakses pukul 12:41.

1. Frekuensi ATIS

Frekuensi yang digunakan untuk ATIS adalah satu frekuensi tersendiri yang memancarkan semua informasi. Di bandar udara besar dengan banyak landasan pacu, kadang-kadang disediakan lebih dari satu frekuensi ATIS, untuk memisahkan beberapa informasi. Dalam hal ini biasanya ATIS untuk keberangkatan (*departure*) dan ATIS untuk kedatangan (*arrival*) dipancarkan pada frekuensi yang berbeda.

Untuk menambah jarak pancar, ATIS juga bisa dipancarkan dari frekuensi radio navigasi VOR (VHF *Omni Range*). Dengan teknik tertentu pancaran ATIS dibedakan dengan pancaran identifikasi VOR. Bagi penerbang cara untuk membedakannya adalah dengan menekan tombol *Voice* pada pengendali frekuensi VOR. Dengan menekan tombol ini, maka pancaran morse identifikasi nama VOR akan digantikan dengan pancaran informasi dari ATIS.

Semua frekuensi yang digunakan oleh ATIS, baik frekuensi sendiri maupun menggunakan frekuensi VOR, masih menggunakan frekuensi VHF (*Very High Frequency*) yang mempunyai keterbatasan jarak dan pantulan dengan *obstacle* (benda yang menghalangi pancaran radio). Tanpa adanya *obstacle*, dengan ketinggian terbang sekitar 37000 kaki, maksimum pancaran ATIS akan didapat dalam jarak 150-200 *nautical mile*, tergantung pada kekuatan pemancarnya. Jarak ini akan berkurang dengan adanya *obstacle* (penghalang) di sekitar bandar udara ¹⁾.



Gambar 2.1 Pengendali VOR Di Pesawat

Frekuensi ATIS berada di jangkauan 108 – 138 MHz. Untuk menempatkan dan mengoperasikan perangkat ATIS, terlebih dahulu pihak bandara harus melapor ke Direktorat Sertifikasi Kelaikan Penerbangan (DSKU), DSKU akan memberikan suatu frekuensi ATIS tersendiri pada bandara yang akan mengoperasikan perangkat ATIS tersebut. Dalam peraturan *International Civil Aviation Organization* (ICAO), organisasi penerbangan internasional, ATIS diatur dalam ANNEX 11.

2. Digital ATIS

Dengan perkembangan komunikasi penerbangan, saat ini dikenal cara komunikasi melalui jaringan data (*data link*) melalui frekuensi VHF, atau SATCOM (*Satellite Communication*), pada waktu hubungan dengan frekuensi VHF tidak tersedia. Sebuah perusahaan penerbangan harus berlangganan untuk mendapatkan layanan sistem jaringan data ini.

Sistem yang disebut ACARS (*Aircraft Communication Addressing and Reporting System*) ini memungkinkan pesawat untuk berkomunikasi dengan

perusahaan, dengan ATC atau layanan penerbangan seperti permintaan laporan cuaca dan ATIS.

Dengan data link ACARS ini, permintaan akan laporan cuaca dari ATIS, tidak terbatas dengan jangkauan pemancar radio ATIS. Dimanapun posisi pesawat, penerbang dapat mendapatkan laporan terakhir keadaan cuaca dan landasan yang digunakan, termasuk semua informasi penting lainnya.

Sayangnya tidak semua bandar udara terhubung dengan D-ATIS (Digital ATIS) ini ¹⁾.

Indonesia sendiri merupakan negara yang belum menggunakan ATIS Digital ini, Indonesia masih menggunakan frekuensi VHF untuk layanan ATIS yang digunakan.



Gambar 2.2 ATIS Digital

3. Informasi Yang Dipancarkan ATIS ²⁾

- a. *Airport identity* : Menunjukkan asal bandara yang menyiarkan ATIS dan kode ATIS.
- b. *METAR Weather sequence* : Menunjukkan kapan informasi ATIS diperbaharui.
- c. *Wind direction* : Arah angin digunakan juga pesawat untuk membantu *take-off* dan *landing*. Untuk lepas-landas sangat baik jika mengikuti arah angin, sedangkan pendaratan sangat baik jika melawan arah angin.
- d. *Wind speed* : Kecepatan angin digunakan untuk status pesawat apakah diijinkan atau tidak untuk *take-off* atau *landing*. Pendaratan tidak akan diijinkan apabila kecepatan angin mencapai 45 *knot*.
- e. *Visibility* : Jarak pandang terbaik untuk melakukan *take-off* dan *landing*. Harus dibawah 1500 meter.
- f. *Present weather* : Keadaan cuaca saat ini di bandara. Ada beberapa keadaan, yaitu RAIN (hujan), FOG (berkabut), NIL (normal), HAZE(kabur).
- g. *Height of clouds* : Ketinggian awan di lapangan terbang yang dapat mempengaruhi jarak pandang penerbang ke landasan.

Klasifikasi *Height of clouds* :

Tabel 2.1 Klasifikasi *Height Of Clouds*

Phrase	Meaning	Cloud Coverage
SKC	sky clear	no cloud
FEW	Few	1 to 2 OKTAS
SCT	scattered	3 to 4 OKTAS
BKN	broken	5 to 7 OKTAS
OVC	overcast	8 OKTAS

Keterangan :

²⁾ Sumber : www.MihirEnterprises.com. 14 Desember 2008. Diakses Pukul 12:22.

- h. *Temperature* : Temperatur sangat mempengaruhi performa dari mesin pesawat terbang, terutama ketika lepas-landas. Ketika suhu tinggi, maka kekuatan daya dorong mesin pesawat dapat berkurang.
- i. *DEW point* : Titik pengembunan
- j. *QNH* : Adalah sebuah kode “Q” yang dipakai dalam dunia penerbangan yaitu sebuah seting tekanan udara berdasarkan pengukuran barometer, yang membaca ketinggian dari permukaan laut.
- k. *Runway* : Adalah jalur perkerasan yang dipergunakan oleh pesawat terbang untuk mendarat (*Landing*) dan lepas-landas (*Take-Off*).

4. Pemancar VHF yang digunakan ATIS

Kriteria dan standar pemancar yang digunakan oleh bandara – bandara di Indonesia mengacu pada ICAO (*ANNEX 10 Aeronautical telecommunication*)³⁾. Standar ini melingkupi lokasi penempatan pemancar dan kondisi lingkungan di sekitar pemancar. VHF merupakan suatu frekuensi yang digunakan untuk mentransmisikan data. VHF merupakan suatu gelombang radio, jadi data yang ditransmisikan disebar ke segala arah.

Tabel 2.2 Spektrum Elektromagnetik

Frequency Band	Name
3 – 10 KHz	ELF Extremely Low Frequency
10 – 30 KHz	VLF Very Low Frequency
30 – 300 KHz	LF Low Frequency
300 – 3000 KHz	MF Medium Frequency
3 – 30 MHz	HF High Frequency / Short wave
30 – 300 MHz	VHF Very High Frequency
300 – 3000 MHz	UHF Ultra High Frequency / Microwave
3 – 30 GHz	SHF Super High Frequency

Keterangan :

³⁾ Sumber : www.geocities.com/periyadi2000/tugas/standar_nasional.pdf. 8 Januari 2009. Diakses Pukul 11:08.

Ketentuan penempatan pemancar VHF di bandar udara terbagi dua, yaitu di dalam bandar dan di luar bandara.

a. Penempatan Pemancar VHF Di Luar Area Bandar Udara

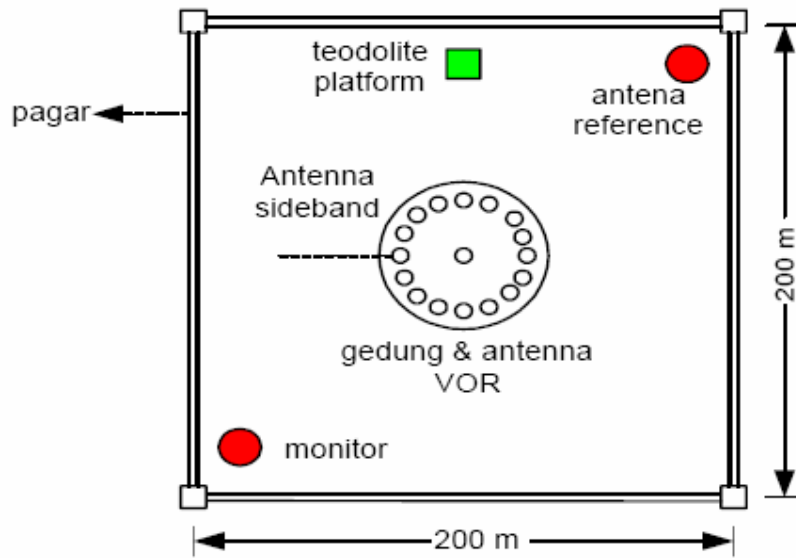
Ada beberapa syarat yang harus dipenuhi dalam penempatan pemancar VHF di luar area bandara ⁴⁾, yaitu :

Tabel 2.3 Syarat Penempatan Pemancar Di Luar Area Bandara

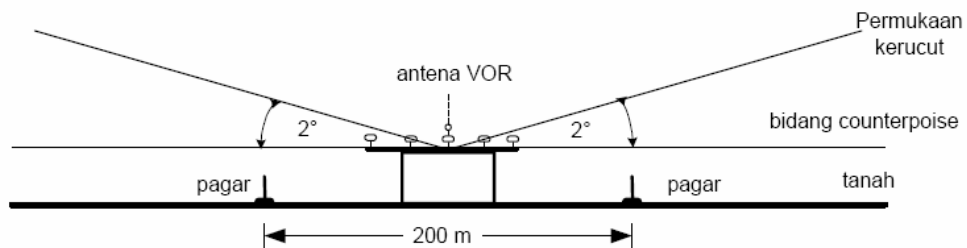
KETENTUAN	SYARAT
Lokasi pemancar	Di luar area bandara
Penempatan pemancar	360 meter sampai 12.971 meter dari ambang landasan pacu
Luas lahan bangunan pemancar	200 meter X 200 meter
<i>Obstacle</i> (Penghalang)	Tidak ada penghalang dalam radius 900 meter dari antena
	Di letakan di tempat tertinggi
	Dalam radius 60 meter dari pemancar harus berupa tanah rata
	Tidak diperkenankan terdapat saluran udara tegangan tinggi sampai jarak 600 meter dari titik pusat antena
	Dalam radius 100 meter dari pusat antena harus bebas benda tumbuh dan bangunan, kecuali bangunan pemancar
	Ketinggian benda tumbuh dan bangunan yang berada di luar radius 200 meter tidak melebihi permukaan kerucut 2 ⁰

Keterangan :

⁴⁾ Sumber : www.geocities.com/BSN/SNI_ref02.pdf, 13 Januari 2009. Diakses Pukul 11:08.



Gambar 2.3 Luas Lahan Dan Lokasi Penempatan VOR



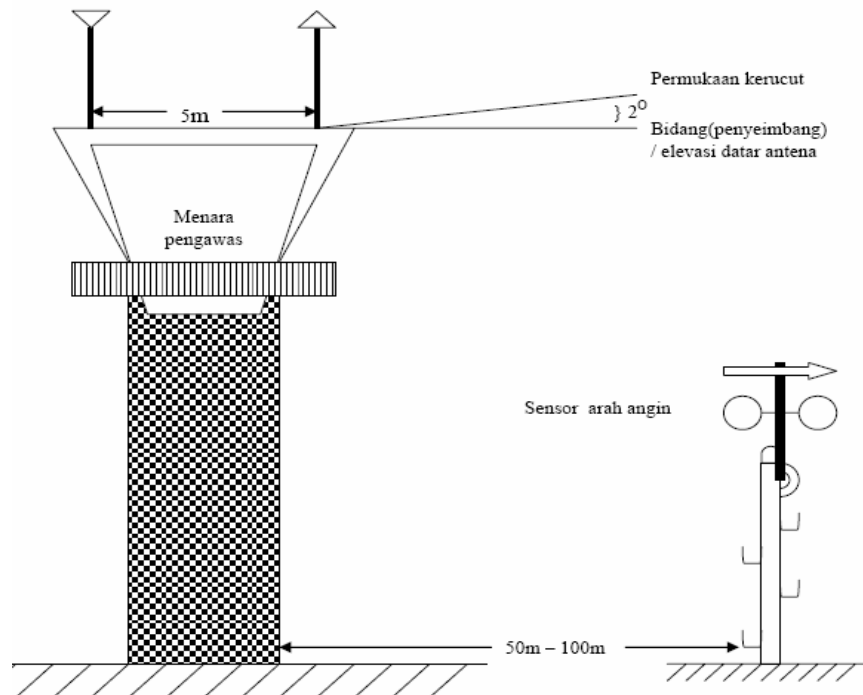
Gambar 2.4 Syarat Batas Ketinggian Dan Benda Tumbuh Di Sekitar VOR

b. Penempatan Pemancar VHF Di Dalam Area Bandar Udara

Seperti penempatan pemancar VHF di luar area bandara, penempatan pemancar VHF di dalam area bandar udara juga harus memenuhi persyaratan yang berlaku ⁴⁾.

Tabel 2.4 Syarat Penempatan Pemancar Di Dalam Area Bandara

KETENTUAN	SYARAT
Lokasi pemancar	Di dalam area bandara
Penempatan pemancar	Di atas menara pengawas
Jarak antara antenna pemancar	Minimal lima meter
Jarak anantara antenna dengan RADAR	Minimal 25 meter
<i>Obstacle</i> (Penghalang)	Ketinggian bangunan di sekitar pemancar sejauh 500 meter tidak melebihi ketinggian elevasi dasar pancaran antenna
	Ketinggian bangunan yang berjarak lebih dari 500 meter dari menara tidak boleh melebihi permukaan kerucut 2°
	Jarak tiang alat sensor arah angin dengan menara pengawas adalah 50 m sampai dengan 100 m



Gambar 2.5 Penempatan Fasilitas VHF A/G dan sensor kecepatan angin

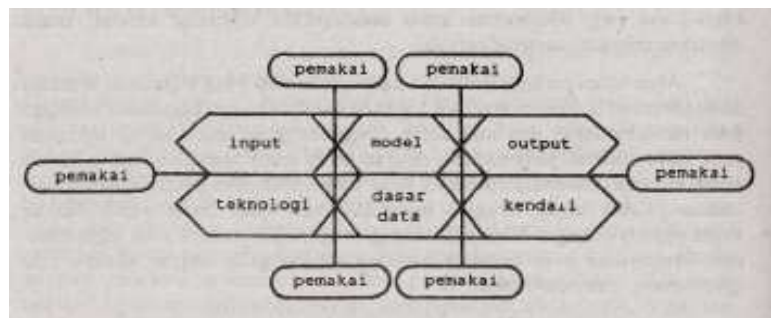
B. Sistem Informasi

1. Konsep Dasar Sistem Informasi

Sistem informasi dapat didefinisikan sebagai suatu sistem di dalam suatu organisasi yang merupakan kombinasi dari orang-orang, fasilitas, teknologi, media, prosedur-prosedur dan pengendalian yang ditujukan untuk mendapatkan jalur komunikasi penting, memproses tipe transaksi rutin tertentu, memberi sinyal kepada manajemen dan yang lainnya terhadap kejadian-kejadian internal dan eksternal yang penting dan menyediakan suatu dasar informasi untuk pengambilan keputusan yang cerdas.

a. Komponen Sistem Informasi

Sistem informasi terdiri dari komponen-komponen yang disebut dengan istilah blok bangunan (*building block*), yaitu blok masukan (*input block*), blok model (*model block*), blok keluaran (*output block*), blok teknologi (*technology block*), blok dasar data (*database block*) dan blok kendali (*control block*). Sebagai suatu sistem, keenam blok tersebut masing-masing saling berinteraksi satu dengan yang lainnya membentuk satu kesatuan untuk mencapai sasarnya⁵⁾.



Gambar 2.6 Blok Sistem Sistem Informasi yang berinteraksi

Keterangan :

⁵⁾ Sumber : Hartono, Jogiyanto. 1999. *Pengenalan Komputer*. Penerbit ANDI. Yogyakarta.

1. Blok Masukan

Input mewakili data yang masuk ke dalam sistem informasi. *Input* di sini termasuk metode-metode dan media untuk menangkap data yang akan di masukkan, yang dapat berupa dokumen-dokumen dasar.

2. Blok Model

Blok ini terdiri dari kombinasi prosedur, logika dan model matematik yang akan memanipulasi data input dan data yang tersimpan di dasar data dengan cara yang sudah tertentu untuk menghasilkan keluaran yang diinginkan.

3. Blok Keluaran

Produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

4. Blok Teknologi

Teknologi merupakan “kotak alat” dari pekerjaan sistem informasi. Teknologi digunakan untuk menerima masukan, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian dari sistem secara keseluruhan.

5. Blok Basis Data

Basis data merupakan kumpulan dari data yang saling berhubungan satu dengan lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya.

6. Blok Kendali

Blok ini utamanya diperlukan untuk memastikan sistem informasi berjalan sesuai dengan yang diinginkan, maka perlu diterapkan pengendalian-pengendalian di dalamnya.

2. Tingkatan Sistem Informasi

Beberapa jenis TI (teknologi informasi) yang dikembangkan berdasarkan lini manajerial, memiliki fungsi dan manfaat bagi tiap tingkatan manajerial⁶⁾. adapun tingkatan SI tersebut adalah :

a. Sistem Pemrosesan Transaksi (SPT)

Merupakan hasil perkembangan dari pembentukan kantor elektronik, dimana sebagian dari pekerjaan rutin diotomatisasi termasuk untuk pemrosesan transaksi. Pada SPT, data yang dimasukkan merupakan data-data transaksi yang terjadi.

b. Sistem Informasi Manajemen (SIM)

Adalah sebuah kelengkapan pengelolaan dari proses-proses yang menyediakan informasi untuk manajer guna mendukung operasi-operasi dan pembuatan keputusan dalam sebuah organisasi. Pada SIM, masukan yang diberikan berupa data transaksi yang telah diproses, beberapa data yang asli, model-model pengolahan data. Kemudian data-data tersebut akan diproses. Proses yang terjadi berupa pembuatan laporan-laporan yang ringkas, keputusan-keputusan yang rutin dan jawaban dari query yang diberikan.

Keterangan :

⁶⁾ Sumber : www.freewebs.com/siskapipin/konsepSI.doc. 6 Februari 2009. Diakses Pukul 16:23.

c. Sistem Pendukung Keputusan (SPK)

Merupakan peningkatan dari SIM dengan penyediaan prosedur-prosedur khusus dan pemodelan yang unik yang akan membantu manajer dalam memperoleh alternative keputusan.

d. Sistem Informasi *E-Business*

dibangun untuk menjawab tantangan pengintegrasian data dan informasi dari proses bisnis berbasis *internet*.

Lebih spesifik dikenal juga yang disebut dengan sistem terotomasi, yang merupakan bagian dari sistem buatan manusia dan berinteraksi dengan kontrol oleh satu atau lebih komputer sebagai bagian dari sistem yang digunakan dalam masyarakat modern. Sistem terotomasi memiliki sejumlah komponen, yaitu :

1. Perangkat keras (CPU, *disk printer, tape*).
2. Perangkat lunak (sistem operasi, sistem basis data, program pengontrol komunikasi, program aplikasi).
3. Personil (yang mengoperasikan sistem).
4. Data (yang harus tersimpan dalam sistem, dalam jangka waktu tertentu).
5. Prosedur (instruksi dan kebijakan untuk mengoperasikan sistem).

3. Kategori Sistem Terotomasi ⁶⁾

a. *On-Line System*

Sistem *on-line* adalah sistem yang menerima langsung *input* pada area dimana *input* tersebut direkam dan menghasilkan *output* yang dapat berupa hasil komputasi pada area dimana mereka dibutuhkan. Area sendiri dapat dipisah-pisah dalam skala, misalnya ratusan kilometer. Biasanya digunakan bagi reservasi angkutan udara, reservasi kereta api, perbankan dll.

b. *Real-Time System*

Adalah mekanisme pengontrolan, perekaman data, pemrosesan yang sangat cepat sehingga *output* yang dihasilkan dapat diterima dalam waktu yang relatif sama. Perbedaan dengan sistem *on-line* adalah satuan waktu yang digunakan *real-time* biasanya seperseratus atau seperseribu detik sedangkan *on-line* masih dalam skala detik atau bahkan kadang beberapa menit. Perbedaan lainnya, *on-line* biasanya hanya berinteraksi dengan pemakai, sedangkan *real-time* berinteraksi langsung dengan pemakai dan lingkungan yang dipetakan.

c. *Decision support system* dan *Strategic planning system*

Sistem yang memproses transaksi organisasi secara harian dan membantu para manajer mengambil keputusan, mengevaluasi dan menganalisa tujuan organisasi. Digunakan untuk sistem penggajian, sistem pemesanan, sistem akuntansi dan sistem produksi. Biasanya berbentuk paket statistik, paket pemasaran dll. Sistem ini tidak hanya merekam dan menampilkan data tetapi juga fungsi-fungsi matematik,

data analisa statistik dan menampilkan informasi dalam bentuk grafik (tabel, chart) sebagaimana laporan konvensional.

d. *Knowledge-based system*

Program komputer yang dibuat mendekati kemampuan dan pengetahuan seorang pakar. Umumnya menggunakan perangkat keras dan perangkat lunak khusus seperti LISP dan PROLOG.

C. Konsep *Client-Server*

Sesuai dengan namanya, *Client-Server* berarti adanya pembagian kerja pengolahan data antara *client* dan *server*. Saat ini, sebagian besar jaringan menggunakan model *client/server*⁵⁾. Secara singkat, jaringan *client/server* adalah jaringan dimana komputer *client* bertugas melakukan permintaan data dan *server* bertugas melayani permintaan tersebut.

1. *Client*

- a. Memberikan antarmuka bagi pengguna untuk melakukan pekerjaan
- b. Format *request* data ke bentuk yang dapat dimengerti oleh *server*
- c. Menampilkan hasil yang diminta ke dalam layar.

Komputer *client* menerima instruksi dari *user* melalui *interface* yang disediakan, merubah format instruksi ke bentuk yang dapat dimengerti oleh *database server*, dan mengirimkannya melalui jaringan ke *server* yang dituju. *Server* kemudian akan mengolah *request*, memilih informasi yang sesuai, dan mengirimkan kembali data hasil pengolahan ke *client*. *Client* kemudian mengolah data yang diterima untuk ditampilkan sebagai informasi yang berguna melalui *interface* yang tersedia.

2. *Server*

Pada jaringan *client/server*, *server* khusus digunakan untuk pemrosesan, penyimpanan dan manajemen data. *Server* bertugas menerima *request* dari *client*, mengolahnya, dan mengirimkan kembali hasilnya ke *client*. Untuk itu, *server* membutuhkan komputer khusus dengan spesifikasi *hardware* yang jauh lebih baik dan bertenaga dibandingkan *hardware* untuk *client* karena komputer harus mampu melayani :

- a. *Request* secara simultan dalam jumlah besar.
- b. Aktivitas manajemen jaringan.
- c. Menjamin keamanan pada *resource* jaringan.

3. Jenis Koneksi Dan Interaksi⁷⁾

a. *Host-Dumb Client*

Pada interaksi ini *client* adalah suatu *dumb* terminal yang tidak melakukan proses komputasi. Sifatnya hanya benar-benar sebagai suatu perangkat I/O. Masukan keyboard akan dikirimkan melalui jaringan (biasanya menggunakan RS-232), ke host. Host akan menangani seluruh proses, dari penanganan keyboard, penanganan *cursor* dan lain sebagainya. Ini dilakukan dengan mengirimkan perintah ke VDU di terminal.

Keterangan :

⁷⁾ Sumber : www.118.98.171.131/Dokumen/Oleh-oleh%201/jis/Client_Server.doc. 15 Februari 2009. Diakses Pukul 13:22.

b. *Host-Smart Client*

Dengan mekanisme ini, *client/workstation* telah menangani *cursor*, *keyboard* sendiri. Hanya proses utama yang masih dilakukan oleh *server*. Banyak dimanfaatkan untuk *terminal connection* yang menggunakan modem. Dengan cara ini data yang harus dikirim antara *host-client* lebih sedikit. Ini meminimalkan biaya komunikasi, sehingga kecepatan akan lebih tinggi.

c. *File-Server-Workstation*

Pada susunan ini *server* hanya bertugas sebagai penyimpanan data. Sehingga proses dijalankan di masing-masing *workstation*. *Workstation-workstation* tersebut akan menggunakan *file* secara bersamaan (*filesharing*). Untuk proses yang besar dibutuhkan *workstation* yang berkemampuan besar pula (RAM dan *processor*nya).

d. *Client-Server Homogen*

Pada sistem ini beban pekerjaan dibagi antara *client* dan *server*. Proses yang membutuhkan kemampuan *server* akan ditangani oleh *server* dan hasilnya akan dikirimkan ke *client*. Dengan cara ini meminimalkan kemampuan *client* yang besar. *Client* dan *Server* sendiri mengacu kepada entiti logis, bukan fisik. Jadi memungkinkan *client* dan *server* berada pada mesin yang sama.

e. *Client-Server Yang Heterogen*

Pada mekanisme ini, program *client* yang digunakan bersifat heterogen, demikian juga *server* yang digunakan juga bersifat

heterogen. Jadi bisa berjalan dengan *hardware* yang berbeda ataupun sistem operasi yang berbeda. Interaksi antara *client-server* benar-benar hanya menggunakan *message*. Sistem ini menunjang sistem terdistribusi yang heterogen.

D. Local Area Network (LAN)

1. Pengertian *Local Area Network* (LAN)

LAN merupakan suatu jaringan yang menghubungkan suatu komunitas *Data Terminal Equipment* (DTE) yang ditempatkan dalam suatu lokasi (gedung atau grup). Umumnya menggunakan media transmisi berupa kabel, cara ini biasa disebut *wired LAN*. Jadi LAN adalah suatu *network* yang terbatas dalam jarak atau area setempat (lokal) ⁵⁾.

2. Kategori Kecepatan Transmisi Data Dalam LAN

a. *High Speed Network*

Kapasitas transmisi data lebih besar dari 20 MBps. Biasanya diterapkan dalam LAN untuk *mainframe computer* yang besar.

b. *Medium Speed Network*

Kapasitas transmisi data sekitar 1 MBps – 20 MBps yang biasanya diterapkan untuk *mainframe computer* yang kecil atau *minicomputer*.

c. *Low Speed Network*

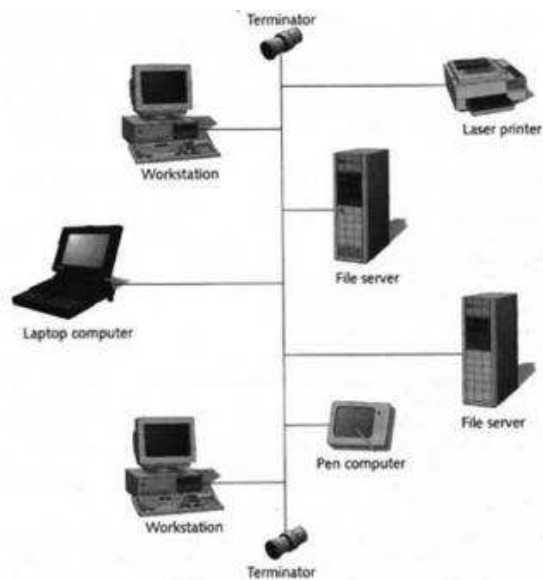
Kapasitas transmisi data lebih kecil dari 1 MBps, biasanya diterapkan untuk *personal computer*.

E. Topologi LAN

Topologi jaringan adalah konfigurasi tentang bagaimana menghubungkan komputer secara fisik sehingga membentuk sebuah jaringan ⁵⁾.

1. Macam-Macam Bentuk Konfigurasi Topologi Jaringan

a. Topologi *Bus*

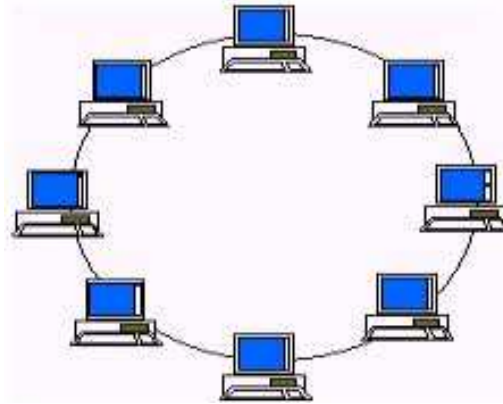


Gambar 2.7 Topologi *Bus*

Pada topologi ini komputer *server* dan *workstation* dihubungkan secara berantai melalui kabel tunggal. Topologi ini menghubungkan beberapa *node* dalam jalur data(*bus*).

Masing-masing *node* dapat melakukan tugas-tugas operasi yang berbeda-beda.

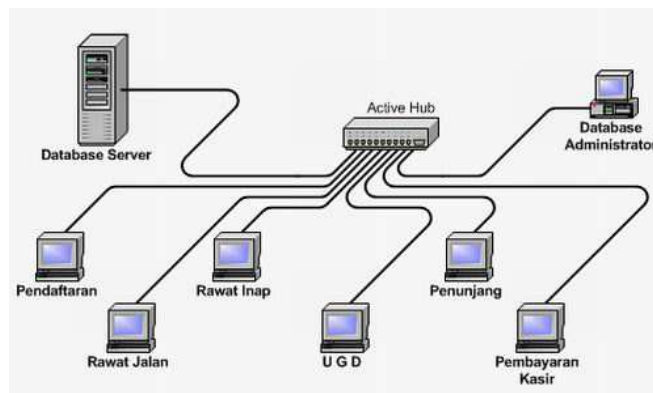
b. Topologi *Ring*



Gambar 2.8 Topologi *Ring*

Pada topologi ini data mengalir searah, artinya seluruh komputer dalam jaringan akan ikut ambil bagian dalam mengelola informasi yang lewat. Sehingga bila salah satu rusak, maka akan berpengaruh terhadap keseluruhan jaringan.

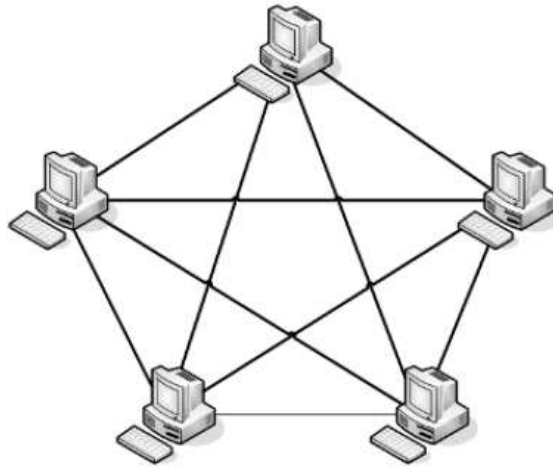
c. Topologi *Star*



Gambar 2.9 Topologi *Star*

Dalam topologi ini, setiap komputer atau workstation dihubungkan secara langsung melalui media perantara berupa *hub* atau *switch*.

d. Topologi *Mesh*



Gambar 2.10 Topologi *Mesh*

Topologi ini merupakan bentuk *network* yang masing-masing *node* dalam *network* dapat berhubungan dengan *node* lainnya melalui beberapa *link*.

2. Kelebihan Dan Kelemahan Masing-Masing Topologi

Tabel 2.5 Kelebihan Dan Kelemahan Topologi Jaringan

Bentuk Topologi	Kelebihan	Kelemahan
<i>Bus</i>	1. Bila sebuah <i>node</i> rusak, tidak akan mengganggu <i>node</i> lainnya.	1. Bila bus rusak, semua <i>node</i> tidak dapat berfungsi.
<i>Ring</i>	1. Waktu untuk mengakses data lebih optimal.	1. Bila sebuah komputer rusak, semua <i>node</i> tidak berfungsi.

Tabel Lanjutan 2.5 Kelebihan Dan Kelemahan Topologi Jaringan

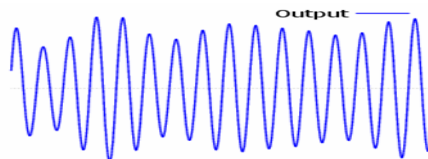
Bentuk Topologi	Kelebihan	Kelemahan
<i>Star</i>	1. Kontrol lebih mudah karena terpusat.	1. Jika central rusak, maka semua tidak dapat berfungsi.
<i>Mesh</i>	1. Bila sebuah Node rusak, Yang lain Masih dapat Berhubungan.	1. Terlalu banyak link, sehingga kontrol agak sulit dilakukan.

F. Modulasi Yang Digunakan ATIS

Modulasi Merupakan teknik-teknik yang dipakai untuk memasukkan informasi dalam suatu gelombang pembawa, biasanya berupa gelombang sinus. Alat yang digunakan untuk modulasi disebut Modulator, dan alat yang digunakan untuk melakukan demodulasi disebut Demodulator, sedangkan alat yang dapat melakukan keduanya adalah Modem ⁸⁾. Jenis modulasi yang digunakan dalam memancarkan ATIS adalah modulasi amplitudo.

1. Modulasi Amplitudo

Modulasi amplitudo termasuk jenis modulasi analog. Modulasi amplitudo adalah suatu bentuk modulasi di mana amplitudo sinyal pembawa divariasikan secara proporsional berdasarkan sinyal pemodulasi (sinyal informasi). Frekuensi sinyal pembawa tetap konstan.



Gambar 2.11 Sinyal Informasi Yang Telah Dimodulasi AM

Keterangan :

⁸⁾ Sumber : Coolen, John. 1984. *Komunikasi elektronika*. Erlangga. Jakarta.

Dalam modulasi AM, amplitudo dari suatu sinyal *carrier*, dengan frekuensi dan phase tetap, divariasikan oleh suatu sinyal lain (sinyal informasi). Modulasi inilah yang digunakan ATIS dalam memancarkan informasi. Karena ATIS menggunakan pemancar VHF, maka ATIS membutuhkan pemancar dengan *Line Of Sight* (LOS) atau garis pandang yang lebar, dengan menggunakan modulasi AM, pancaran gelombang radio dapat mencakup area yang lebih luas.

G. Visual Basic 6.0

Visual basic 6.0 merupakan salah satu aplikasi pemrograman visual yang dibuat Microsoft. Dalam Visual Basic 6.0 menyediakan berbagai perangkat yang dapat digunakan untuk membuat program aplikasi baik aplikasi kecil dan sederhana untuk keperluan sendiri, hingga aplikasi untuk *system enterprise* yang besar dan rumit, atau bahkan aplikasi yang dijalankan melalui Internet ⁹⁾.

Sebab utama semakin banyaknya pengguna Visual Basic dewasa ini dikarenakan kemampuan Visual Basic untuk dapat berinteraksi dengan aplikasi lain di dalam sistem operasi Windows dengan komponen ActiveX Control. Dengan komponen ini memungkinkan pengguna untuk memanggil dan menggunakan semua model data yang ada di dalam sistem operasi windows. Hal ini juga ditunjang dengan teknik pemrograman di dalam Visual Basic yang jenis pemrograman visual.

Visual Basic 6.0 sebetulnya perkembangan dari versi sebelumnya dengan beberapa penambahan komponen yang sedang tren saat ini, seperti kemampuan pemrograman internet dengan DHTML (*Dynamic HyperText Mark Language*), dan beberapa penambahan fitur database dan multimedia yang

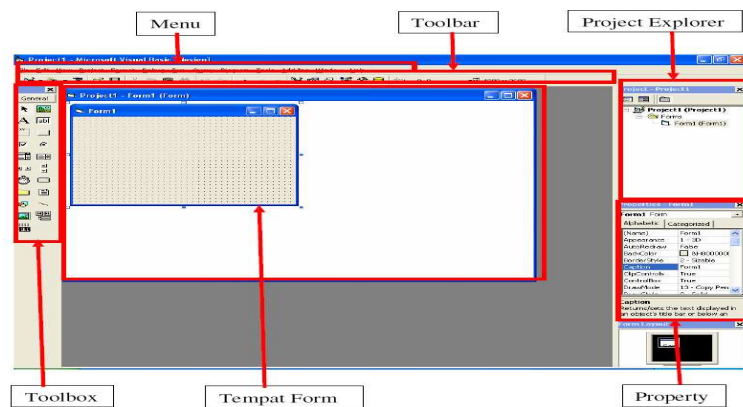
Keterangan :

⁹⁾ Sumber : Nugroho, Bunafit. 2002. *Belajar Visual Basic 6.0*. Elex Media Komputindo.Jakarta

semakin baik. Sampai saat buku ini ditulis bisa dikatakan bahwa Visual Basic 6.0 masih merupakan pilih pertama di dalam membuat program aplikasi yang ada di pasar perangkat lunak nasional. Hal ini disebabkan oleh kemudahan dalam melakukan proses *development* dari aplikasi yang dibuat.

1. Antar Muka Visual Basic 6.0

Interface antar muka Visual Basic 6.0, berisi menu, toolbar, toolbox, form, project explorer dan property. Berikut ini merupakan gambar dari antar muka perangkat lunak Visual Basic seri 6.0.

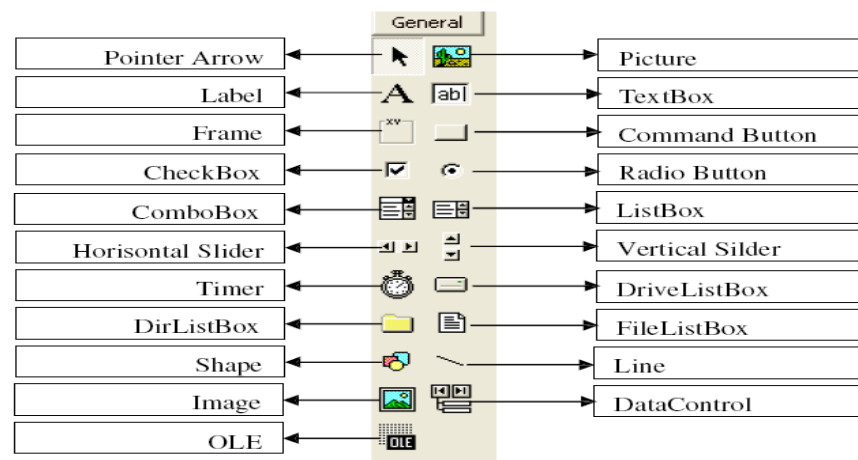


Gambar 2.12 Antar Muka Visual Basic 6.0

Pembuatan program aplikasi menggunakan Visual Basic dilakukan dengan membuat tampilan aplikasi pada *form*, kemudian diberi *script* program di dalam komponen-komponen yang diperlukan. *Form* disusun oleh komponen-komponen yang berada di [*Toolbox*], dan setiap komponen yang dipakai harus diatur propertinya lewat jendela [*Property*].

Menu pada dasarnya adalah operasional standar di dalam sistem operasi windows, seperti membuat *form* baru, membuat *project* baru, membuka *project* dan menyimpan *project*. Di samping itu terdapat fasilitas-fasilitas pemakaian visual basic pada menu. Untuk lebih jelasnya Visual Basic menyediakan bantuan yang sangat lengkap dan detail dalam MSDN.

Toolbox berisi komponen-komponen yang bisa digunakan oleh suatu *project* aktif, artinya isi komponen dalam *toolbox* sangat tergantung pada jenis *project* yang dibangun.



Gambar 2.13 Komponen Standar Dalam *Toolbox*

2. Aturan Pemrograman Visual Basic 6.0

1. *Property*

Setiap komponen di dalam pemrograman Visual Basic dapat diatur propertinya sesuai dengan kebutuhan aplikasi. *Property* yang tidak boleh dilupakan pada setiap komponen adalah "*Name*", yang berarti nama variabel.

(komponen) yang akan digunakan dalam scripting. Properti “*Name*” ini hanya bisa diatur melalui jendela *Property*, sedangkan nilai peroperti yang lain bisa diatur melalui *script* seperti

```
Command1.Caption="Play"
Text1.Text="Visual Basic"
Label1.Visible=False
Timer1.Enable=True
```

Gambar 2.14 Komponen Properti Dalam VB 6.0

2. Method

Metode inilah tempat untuk mengekspresikan logika pemrograman dari pembuatan suatu program aplikasi.

3. Event

Setiap komponen dapat beraksi melalui *event*, seperti *event click* pada *command button* yang tertulis dalam layar *script Command1_Click*, atau *event Mouse Down* pada picture yang tertulis dengan *Picture1_MouseDown*. Pengaturan *event* dalam setiap komponen yang akan menjalankan semua metode yang dibuat.

Struktur dasar pemrograman Visual Basic 6.0 adalah sebagai berikut:

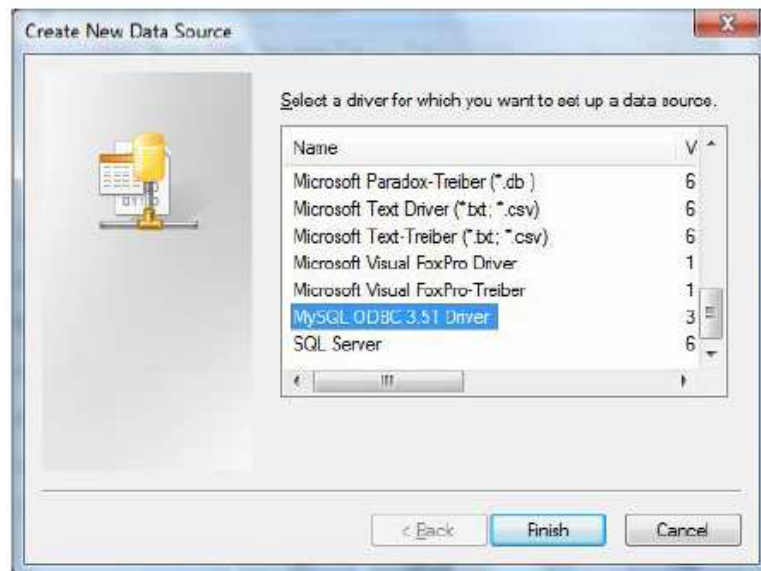
```
Private Sub Command1_Click()
Label2.Caption = Text1.Text
End Sub

Private Sub Command2_Click()
End
End Sub
```

Gambar 2.15 Event Dalam VB 6

H. Open Database Conectivity (ODBC)

ODBC adalah standar yang digunakan untuk mengakses *database*. Di dalamnya memiliki sekumpulan fungsi yang memungkinkan kita terkoneksi ke *local* atau *remote data source*. Data yang dapat di akses bermacam-macam dari dBase, Microsoft FoxPro, Microsoft Access, Microsoft SQL Sever, Oracle, MySQL, dan sebagainya ⁷⁾.



Gambar 2.16 *Setting ODBC Di Windows*

I. *Structured Query Language (SQL)*

Structured Query Language (SQL) adalah bahasa yang digunakan untuk berinteraksi dengan *database*. *Database* adalah kumpulan data yang disimpan dengan berbagai cara pengorganisasian. Cara termudah untuk memahaminya ialah dengan membayangkan *database* sebagai sebuah lemari. Lemari adalah bentuk lokasi fisik yang sederhana untuk penyimpanan data, tanpa memperhatikan apa jenis datanya dan bagaimana data itu diatur.

SQL khusus diciptakan untuk berkomunikasi dengan *database*. Dirancang dengan sedikit kata agar efisien dalam membaca dan menulis data kedalam sebuah *database*. Hampir semua database besar mendukung SQL sehingga, dengan mempelajari SQL kita dapat berinteraksi dengan hampir semua *software database* yang beredar ¹⁰⁾.

1. *Structured Query Language (SQL) dengan MySQL*

Software database Structure Query Language yang sering dipakai dan dapat diintegrasikan dengan *Apache* adalah *MySQL*. Dengan *software* ini, anda akan mempunyai suatu sistem *database* yang dapat diakses dari *web*. Jika anda memiliki *database* berbasis *web* seperti *MySQL-Apache*, maka anda dapat dengan mudah mendapatkan data itu dari mana saja di dunia ini asal terhubung ke jaringan. *MySQL* didistribusikan secara gratis dibawah lisensi GPL (*General Public License*). Dimana setiap orang bebas menggunakan *MySQL*, namun tidak boleh dijadikan produk turunan yang bersifat komersil.

2. Keunggulan *MySQL*

Sebagai server basis data yang memiliki konsep basis data modern, *MySQL* memiliki banyak keistimewaan. Berikut beberapa keistimewaan yang dimiliki oleh *MySQL*.

1. Dapat beroperasi pada banyak platform yang berbeda, antara lain *Windows, Linux, FreeBSD, Mac OS X Server, Solaris, NetBSD, HP-UX*.
2. *MySQL* didistribusikan secara gratis, dibawah lisensi GPL. Sehingga kita dapat menggunakannya secara cuma-cuma.

Keterangan :

¹⁰⁾ Sumber : Hutabarat, Bernaridho. 2005. *Pemrograman MySQL*. Dian Rakyat. Jakarta

3. Mempunyai banyak tipe kolom, seperti *unsign/sign integer* 1,2,3,4 dan 8 bit panjangnya, *float*, *double*, *char*, *varchar*, *text*, *blob*, *date*, *datetime*, *set*, *year* dan *enum*.
4. Kecepatan yang tinggi dalam hal menangani *query* sederhana, dengan kata lain dapat memproses lebih banyak SQL per satuan waktu.
5. Mendukung fungsi lengkap pada bagian *SELECT* dan *WHERE* dalam *query*.
6. Metode enkripsi *password* yang baik.
7. Mampu menangani *database* yang besar (lebih dari 50.000.000 record)
8. *MySQL* dapat digunakan oleh beberapa *user* dalam waktu yang bersamaan tanpa mengalami masalah. Hal ini memungkinkan sebuah *database server MySQL* dapat diakses *client* secara bersamaan.
9. Antarmuka program-program aplikasinya banyak, antara lain bahasa *C*, *C++*, *JAVA*, *Pearl*, *Phyton*, dan *TCL*.
10. Mendukung fungsi *GROUP BY* dan *ORDER BY*.

3. Tipe Data pada MySQL

1. Tipe Data Numerik

Tipe data numerik dibedakan dalam dua kelompok, yaitu: *integer* dan *floating point*. *Integer* digunakan untuk data bilangan bulat, sedangkan *floating point* digunakan untuk bilangan desimal.

2. Tipe data *Char()* dan *Varchar()*

Tipe data *char()* dan *varchar()* pada prinsipnya sama. Bedanya hanya terletak pada jumlah memori yang dibutuhkan untuk penyimpanannya. Memori yang dibutuhkan untuk tipe data *char()* bersifat statis, besarnya bergantung pada jumlah karakter yang ditetapkan pada saat *field* tersebut dideklarasikan. Pada tipe data *varchar()*, besarnya memori penyimpanan tergantung pada jumlah karakter ditambah 1 *byte*.

3. Tipe Data Tanggal

Untuk tanggal dan jam tersedia tipe-tipe data *field* berupa *DATETIME*, *DATE*, *TIME*, dan *YEAR*. Masing-masing tipe mempunyai kisaran nilai tertentu. Mulai dari *MySQL* 5.0.2, *MySQL* memberikan peringatan kesalahan (*error*) apabila tanggal atau waktu yang dimasukkan salah.

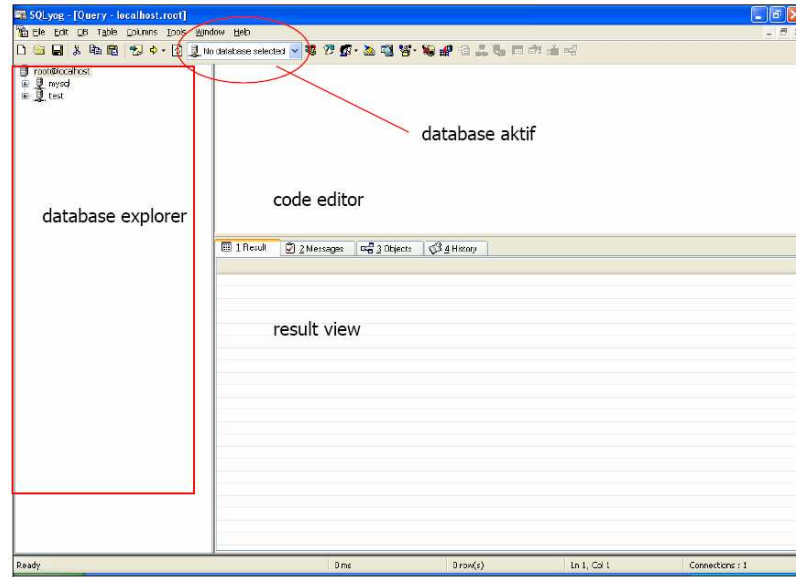
J. SQLyog Enterprises

SQLyog merupakan suatu perangkat lunak yang digunakan untuk mempermudah dalam merancang, mengatur dan memanipulasi basis data. SQLyog dirancang berbasis *Graphical User Interface* (GUI). Perangkat lunak ini dirancang oleh perusahaan perangkat lunak Webyog, yang bertempat di Bangalore, India ¹¹⁾.

SQLyog bekerja di lingkungan sistem operasi Windows, perangkat lunak ini juga dapat bekerja di lingkungan sistem operasi UNIX, tetapi dengan menggunakan perangkat lunak tambahan, yaitu Wine.

Keterangan :

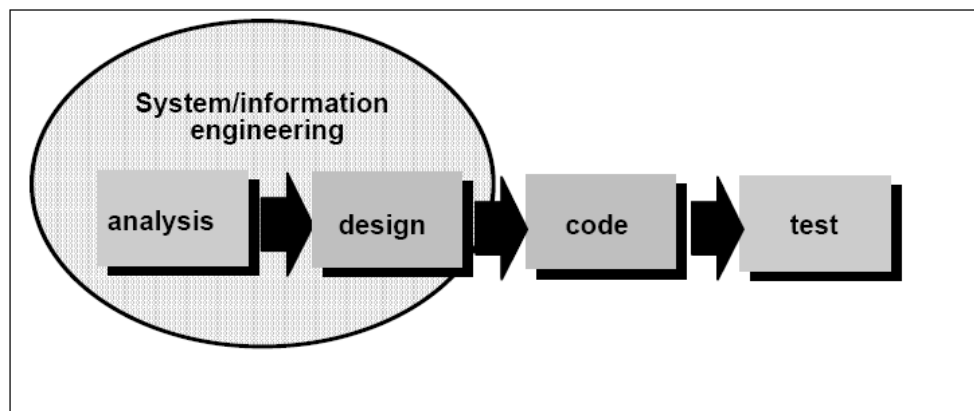
¹¹⁾ Sumber : www.evodevelopment.com/SQLyog.pdf. 27 Januari 2009. Diakses Pukul 20:00.



Gambar 2.17 Antar Muka SQLyog

K. Metode *Waterfall*

Dalam pengembangan sistem informasi perlu digunakan metodologi sebagai pedoman bagaimana dan apa yang harus dilakukan selama melaksanakan pengembangan sistem¹²⁾. Adapun pengembangan sistem yang digunakan adalah paradigma *waterfall (classical life cycle)* tahapan-tahapannya adalah sebagai berikut :



Gambar 2.18 Metode *Waterfall*

Keterangan :

¹²⁾ Sumber : Pressman, Roger S, Ph.D. 1997. *Rekayasa Perangkat Lunak: Pendekatan Praktisi*. Penerbit ANDI.

Keterangan :

a. *System Engineering*

Karena perangkat lunak adalah bagian terbesar dari sistem, maka pekerjaan dimulai dengan cara menetapkan kebutuhan semua elemen sistem dan mengalokasikan sebagian kebutuhan tersebut ke perangkat lunak. Pandangan terhadap sistem adalah penting, terutama pada saat perangkat lunak berhubungan dengan elemen lain, seperti perangkat keras, perangkat lunak lain, *database*.

b. *Analysis*

Suatu proses pengumpulan kebutuhan perangkat lunak untuk mengerti sifat-sifat program yang dibentuk *software engineering*, atau analisis harus mengerti fungsi perangkat lunak yang diinginkan, *performance* dan *interface* terhadap elemen lainnya. Hasil dari analisis ini didokumentasikan dan dibahas atau ditinjau bersama-sama *customer*.

c. *Design*

Proses desain menterjemahkan kebutuhan ke dalam representasi perangkat lunak yang dapat diukur kualitasnya sebelum memulai *coding*. Hasil dari desain ini didokumentasikan dan menjadi bagian dari konfigurasi perangkat lunak.

d. *Coding*

Bentuk rancangan diubah menjadi suatu bahasa pemrograman yang dapat dimengerti oleh mesin komputer.

e. *Testing*

Segera sesudah objek program dihasilkan, pengetesan program dimulai. Proses *testing* difokuskan pada logika internal perangkat lunak. Jaminan bahwa semua pernyataan atau *statements* sudah dites dan lingkungan *external* menjamin bahwa definisi *input* akan menghasilkan *output* yang diinginkan.

f. *Maintenance*

Perangkat lunak yang sudah dikirim ke *customer* berubah karena :

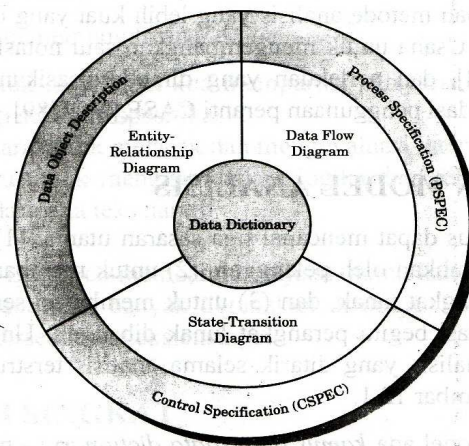
1. Perangkat lunak mengalami *error*.
2. Perangkat lunak harus diadaptasi untuk menyesuaikan dengan lingkungan eksternal, misalnya adanya sistem operasi baru atau peripheral baru.
3. Perangkat lunak yang lebih disempurnakan karena permintaan *customer*.

Model ini adalah model klasik yang bersifat sistematis, berurutan dalam membangun perangkat lunak. Model ini juga merupakan model utama dan model dasar dari banyak model perancangan perangkat lunak.

L. Elemen Model Analisis

Model analisis harus dapat mencapai tiga sasaran , yaitu : untuk menggambarkan apa yang dibutuhkan pelanggan, untuk membangun dasar bagi pembuatan desain perangkat lunak, dan membatasi serangkaian persyaratan yang dapat divalidasi begitu perangkat lunak dibangun. ¹²⁾ .

Pada inti model ada kamus data (*data dictionary*)- penyimpanan yang berisi deskripsi dari semua objek data yang dikonsumsi atau diproduksi perangkat lunak. Kemudian terdapat tiga diagram yang mengelilingi inti. *Entity-relationship diagram* (ERD) menggambarkan hubungan antara objek dan data. Atribut dari masing-masing objek data yang ditulis pada ERD dapat digambarkan dengan menggunakan deskripsi objek data. *Data flow diagram* (DFD) melayani dua tujuan : untuk memberikan indikasi mengenai bagaimana data ditransformasi pada saat data bergerak melalui sistem, dan untuk menggambarkan fungsi-fungsi dan sub-fungsi) yang mentransformasi aliran data. Deskripsi setiap fungsi yang disajikan DFD diisikan dalam sebuah *process specification* (PSPEC). *State transition diagram* (STD) menunjukkan bagaimana sistem bertingkah laku sebagai akibat dari kejadian eksternal. Informasi tambahan mengenai aspek kontrol dari perangkat lunak didisikan dalam *control specification* (CSPEC).



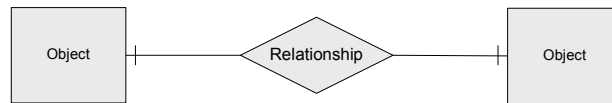
Gambar 2.19 Struktur Model Analisis

1. Entity Relationship Diagram (ERD)

ERD merepresentasikan pemodelan data dimana pemodelan ini diperlukan untuk meminimalkan kebergantungan objek data terhadap proses, memfokuskan pada pengekplorasi domain data, membuat suatu model yang memudahkan *customer* memahaminya dan mengindikasikan keterhubungan antara suatu objek data dengan objek data lainnya.

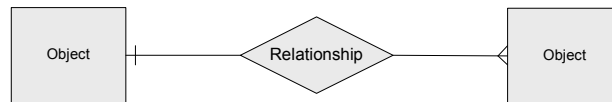
Notasi ERD digambarkan sebagai berikut:

a. One to one



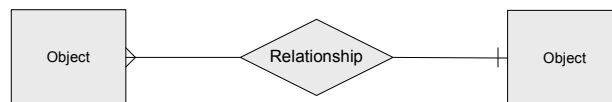
Gambar 2.20 ERD *One to one*

b. One to many



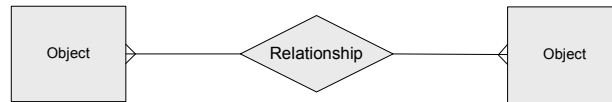
Gambar 2.21 ERD *One to many*

c. Many to one



Gambar 2.22 ERD *Many to one*

d. *Many to many*



Gambar 2.23 ERD *Many to many*

Untuk kasus dimana data yang terlibat dalam sistem tidak memerlukan penyimpanan (basis data) atau jika objek-objek data tersebut tidak memiliki keterhubungan satu sama lain, maka ERD tidak perlu digambarkan

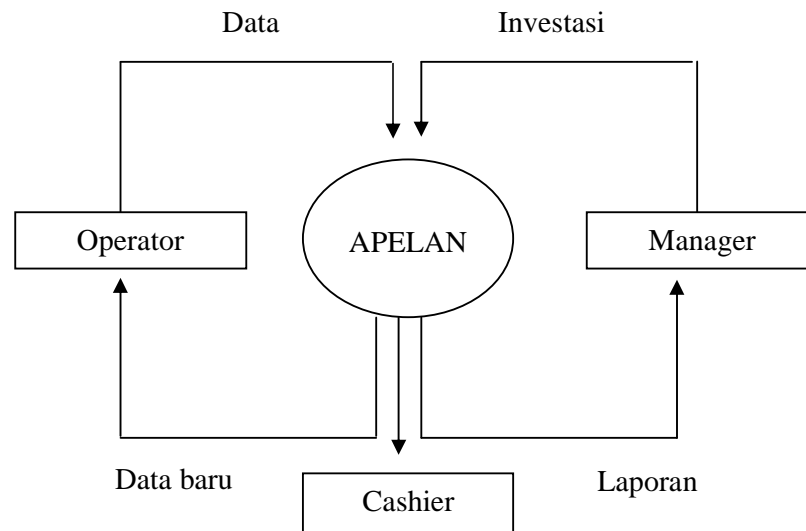
2. *Data Context Diagram (DCD)*

Diagram konteks merupakan diagram level pertama yang memperlihatkan sistem sebagai suatu proses yang berinteraksi dengan lingkungannya. Ada pihak luar yang memasukkan informasi ke dalam sistem dan ada yang menerima informasi dari sistem. Pihak luar bisa berupa sistem lain, perangkat keras, orang atau organisasi.

Pembuatan diagram konteks dilakukan dengan tahapan sebagai berikut :

- a. Menentukan entitas eksternal.
- b. Menentukan informasi yang mengalir dari entitas luar ke sistem dan sebaliknya.
- c. Menggambarkan diagram konteks.

Gambar di bawah memperlihatkan contoh diagram konteks

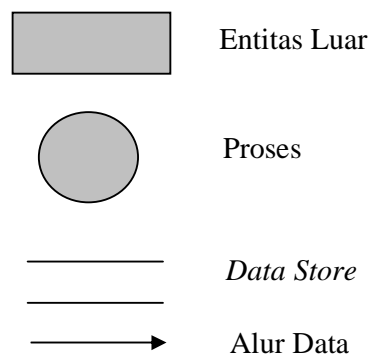


Gambar 2.24 Data Context Diagram

3. Data Flow Diagram (DFD)

DFD merepresentasikan pemodelan fungsi dimana pemodelan ini diperlukan untuk memperlihatkan proses-proses yang dimiliki aplikasi dan bagaimana proses tersebut mentransformasikan data menjadi informasi. DFD digunakan untuk menggambarkan aliran data yang mengalir dalam sistem atau perangkat lunak tersebut.

Notasi DFD dapat dilihat dari gambar di bawah :



Gambar 2.25 Komponen DFD

Entitas luar adalah entitas diluar sistem yang berkomunikasi atau berhubungan langsung dengan sistem. Entitas luar dapat berupa orang, sekelompok orang, organisasi, perusahaan atau departemen yang berada di luar sistem yang akan dibuat, diberi nama yang berhubungan dengan sistem tersebut dan biasanya menggunakan kata benda.

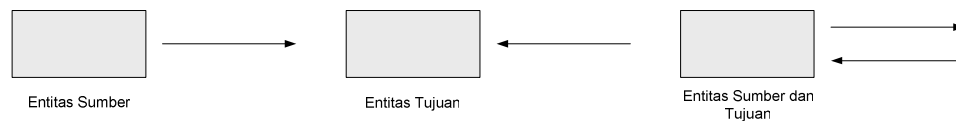
Terdapat dua jenis entitas luar:

1. Entitas luar sumber

Merupakan entitas luar yang menjadi sumber.

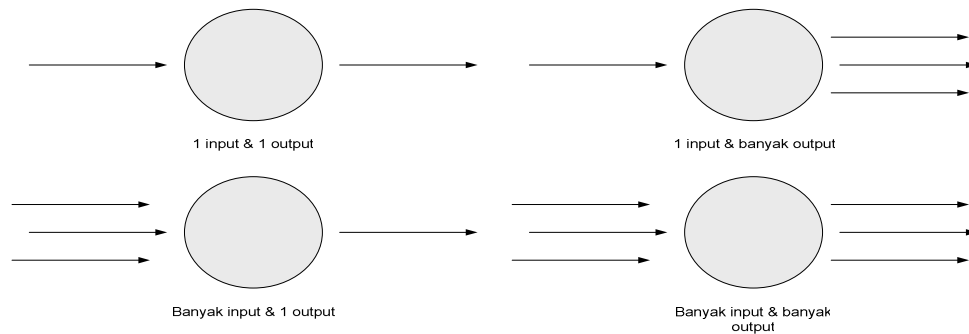
2. Entitas luar tujuan

Merupakan entitas luar yang menjadi tujuan data atau informasi sistem.



Gambar 2.26 Entitas luar

Proses menggambarkan transformasi *input* menjadi *output*. Penamaan proses disesuaikan dengan proses atau kegiatan yang sedang dilakukan. Ada empat kemungkinan yang dapat terjadi dalam proses sehubungan dengan *input* dan *output*:

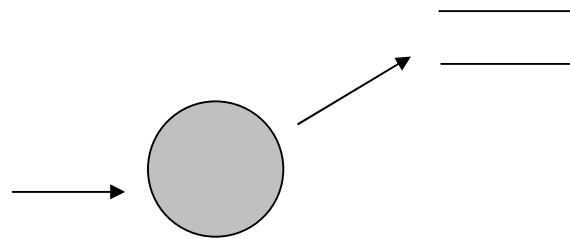


Gambar 2.27 Proses

Data store digunakan untuk membuat model sekumpulan paket data dan diberi nama dengan kata benda bersifat jamak. *Data store* dapat berupa *file* atau *database* yang tersimpan dalam disket, *hardisk* atau bersifat manual seperti buku alamat, *file folder*.

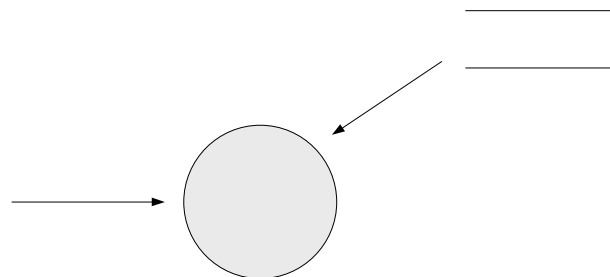
Yang perlu diperhatikan tentang *data store*:

1. Alur data dari proses menuju *data store*, hal ini berarti *data store* berfungsi sebagai tujuan atau tempat penyimpanan dari suatu proses (proses *write*).



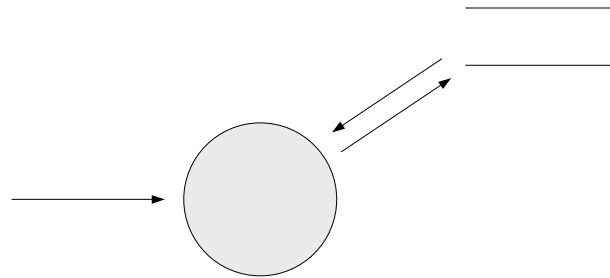
Gambar 2.28 Proses *Write*

2. Alur data dari *store* ke proses, hal ini berarti *data store* berfungsi sebagai sumber atau proses memerlukan data (proses *read*)



Gambar 2.29 Proses *Read*

3. Alur data dari proses menuju *data store* dan sebaliknya berarti berfungsi sebagai sumber dan tujuan (proses *read and write*).



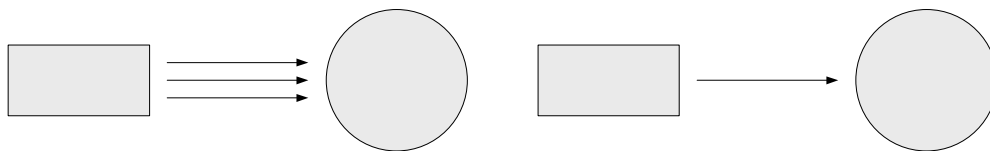
Gambar 2.30 Proses *Read and Write*

Alur data digunakan untuk menerangkan perpindahan data atau paket data dari satu bagian ke bagian lainnya. Alur data dapat berupa kata, pesan, formulir atau informasi.

Ada empat konsep tentang alur data:

1. *Packets of Data*

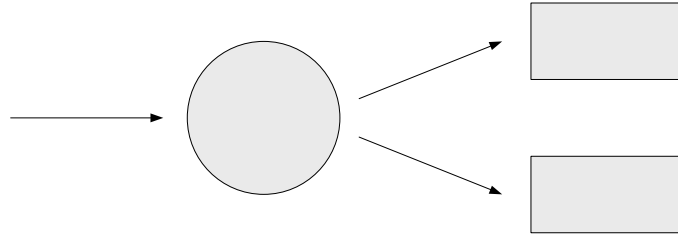
Apabila ada dua data atau lebih yang mengalir dari satu sumber yang sama menuju pada tujuan yang sama dan mempunyai hubungan digambarkan dengan satu alur data.



Gambar 2.31 *Packets of Data*

2. *Diverging Data Flow*

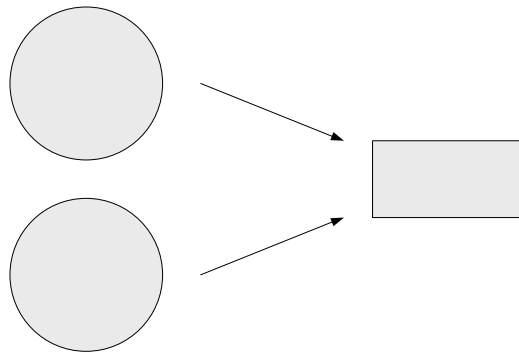
Apabila ada sejumlah paket data yang berasal dari sumber yang sama menuju pada tujuan yang berbeda atau paket data yang kompleks dibagi menjadi beberapa elemen data yang dikirim ke tujuan yang berbeda.



Gambar 2.32 *Diverging Data Flow*

3. *Converging Data Flow*

Apabila ada beberapa alur data yang berbeda sumber menuju ke tujuan yang sama.

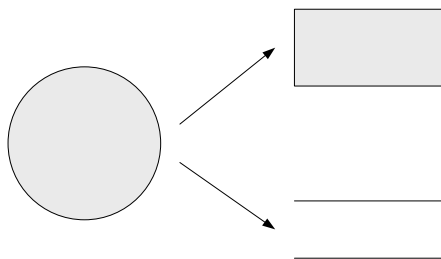


Gambar 2.33 *Converging data flow*

4. Sumber dan tujuan

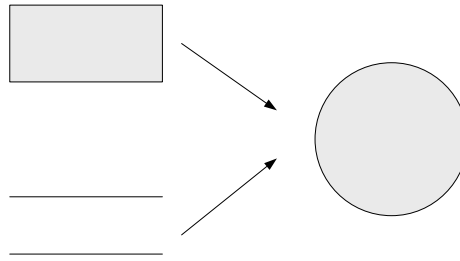
Arus data harus dihubungkan pada proses, baik dari maupun yang menuju proses.

Dari proses ke bukan proses



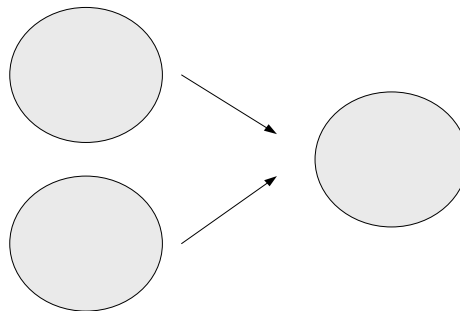
Gambar 2.34 *Dari Proses Ke Bukan Proses*

Dari bukan proses menuju proses



Gambar 2.35 Dari Bukan Proses Ke Proses

Dari proses ke proses



Gambar 2.36 Dari Proses Ke Proses

4. *Specification Control (CSPEC)*

CSPEC merepresentasikan tingkah laku sistem (pada tingkat di mana dia direferensi) di dalam dua cara yang berbeda. CSPEC berisi sebuah diagram transisi keadaan (STD) yang merupakan suatu spesifikasi sekuensial dari tingkah laku. Dia juga dapat berisi suatu tabel aktivasi proses (PAT)- sebuah spesifikasi kombinatorial dari tingkah laku.

5. *Process Specification (PSPEC)*

Spesifikasi proses (PSPEC) digunakan untuk menggambarkan semua proses model aliran yang nampak pada tingkat akhir penyaringan. Kandungan dari

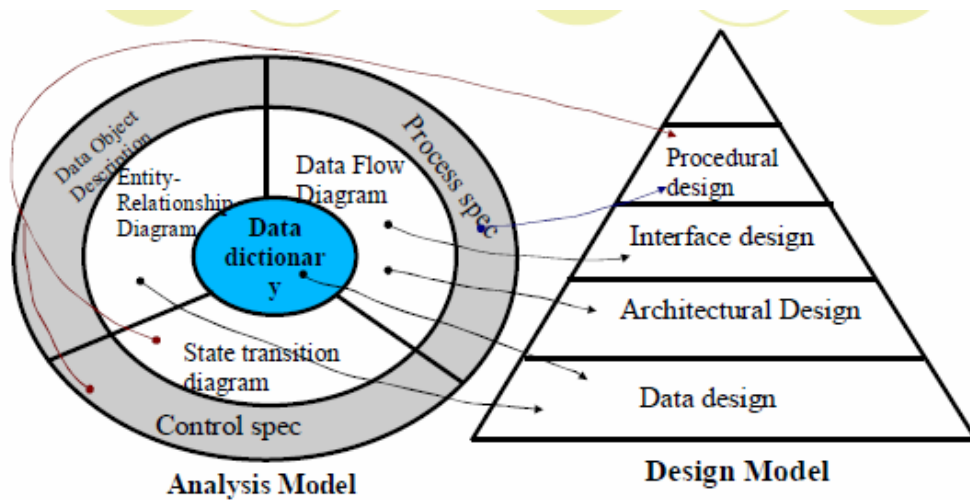
spesifikasi proses dapat termasuk teks naratif, gambaran bahasa desain program dari algoritma proses, persamaan matematika, tabel, diagram atau bagan.

6. Data Dictionary

Data dictionary adalah sebuah tata bahasa untuk menggambarkan kandungan dari objek yang didefinisikan selama analisis terstruktur. Jadi, kamus data merupakan sebuah daftar yang terorganisasi dari elemen data yang berhubungan dengan sistem, dengan definisi yang tegas dan teliti sehingga pemakai dan analis sistem akan memiliki pemahaman yang umum mengenai *input*, *output*, komponen penyimpan, dan bahkan kalkulasi inter-mediate.

M. Model Desain

Desain digambarkan sebagai proses multi-langkah di mana representasi struktur data, struktur program, karakteristik *interface*, dan detail prosedur, disintesis dari persyaratan informasi.



Gambar 2.37 Model Desain

1. Desain Data

Desain data adalah aktifitas pertama (dan beberapa sering mengatakan yang terpenting) dari aktifitas-aktifitas lain yang dilakukan selama rekayasa perangkat lunak. Pengaruh struktur data pada struktur program dan kompleksitas prosedural menyebabkan desain data berpengaruh penting terhadap kualitas perangkat lunak.

2. Desain Arsitektur

Sasaran utama desain arsitektur adalah untuk mengembangkan struktur program modular dan merepresentasikan hubungan kontrol antar modul. Desain arsitektur juga membentuk struktur program dan struktur data dengan menentukan interface yang memungkinkan data mengalir melalui program. Dalam bentuknya yang paling sederhana, arsitektur merupakan struktur hirarki dari komponen program (modul) ¹²⁾.

3. Desain *Interface*

Desain interface memfokuskan diri pada tiga area perhatian : (1) desain *interface* antara modul-modul perangkat lunak; (2) desain *interface* antara perangkat lunak dan prosedur dan konsumen informasi bukan manusia lainnya (yakni entitas eksternal lainnya); dan (3) desain *interface* antara seorang manusia (seperti pemakai) dan komputer.

N. *Text To Speech*

Sistem *Text to speech* sebenarnya terdiri atas dua sub sistem ¹³⁾, yaitu :

1. Bagian konverter teks ke fonem
2. Bagian konverter fonem ke ucapan



Gambar 2.38 Blok Diagram *Text To Speech*

Bagian konverter teks ke fonem berfungsi untuk mengubah kalimat masukan dalam suatu bahasa tertentu yang berbentuk teks menjadi rangkaian kode-kode bunyi yang biasanya direpresentasikan dengan kode fonem, durasi serta *pitch*-nya. Bagian ini bersifat sangat *language dependant*. Untuk suatu bahasa baru, bagian ini harus dikembangkan secara lengkap khusus untuk bahasa tersebut.

Bagian konverter fonem ke ucapan akan menerima masukan berupa kode-kode fonem serta *pitch* dan durasi yang dihasilkan oleh bagian sebelumnya. Berdasarkan kode-kode tersebut, bagian Konverter Fonem ke Ucapan akan menghasilkan bunyi atau sinyal ucapan yang sesuai dengan kalimat yang ingin diucapkan. Ada beberapa alternatif teknik yang dapat digunakan untuk implementasi bagian ini. Dua teknik yang banyak digunakan adalah *formant synthesizer*, serta *diphone concatenation*.

Formant synthesizer bekerja berdasarkan suatu model matematis yang akan melakukan komputasi untuk menghasilkan sinyal ucapan yang diinginkan. Synthesizer jenis ini telah lama digunakan pada berbagai aplikasi. Walaupun dapat menghasilkan ucapan dengan tingkat kemudahan interpretasi yang baik,

Keterangan :

¹³⁾ Sumber : (<http://oc.itb.ac.id/TTS.pdf>, 16 Januari 2009, Diakses Pukul 16:30).

synthesizer ini tidak dapat menghasilkan ucapan dengan tingkat kealamian yang tinggi.

Synthesizer yang menggunakan teknik *diphone concatenation* bekerja dengan cara menggabung-gabungkan segmen-segmen bunyi yang telah direkam sebelumnya. Setiap segmen berupa *diphone* (gabungan dua buah fonem). Synthesizer jenis ini dapat menghasilkan bunyi ucapan dengan tingkat kealamian (*naturalness*) yang tinggi. Struktur sistem seperti di atas pada prinsipnya merupakan konfigurasi tipikal yang digunakan pada berbagai sistem *text to Speech* berbagai bahasa. Namun demikian, pada setiap sub-sistem terdapat sifat-sifat serta proses-proses yang sangat spesifik dan sangat tergantung dari bahasanya.

Konversi dari teks ke fonem sangat dipengaruhi oleh aturan-aturan yang berlaku dalam suatu bahasa. Pada prinsipnya proses ini melakukan konversi dari simbol-simbol tekstual menjadi simbol-simbol fonetik yang merepresentasikan unit bunyi terkecil dalam suatu bahasa. Setiap bahasa memiliki aturan cara pembacaan dan cara pengucapan teks yang sangat spesifik. Hal ini menyebabkan implementasi unit konverter teks ke fonem menjadi sangat spesifik terhadap suatu bahasa.

Untuk mendapatkan ucapan yang lebih alami, ucapan yang dihasilkan harus memiliki intonasi (*prosody*). Secara kuantisasi, prosodi adalah perubahan nilai *pitch* (frekuensi dasar) selama pengucapan kalimat dilakukan atau *pitch* sebagai fungsi waktu. Pada prakteknya, informasi pembentuk prosodi berupa data-data *pitch* serta durasi pengucapannya untuk setiap fonem yang dibangkitkan. Nilai-nilai yang dihasilkan diperoleh dari suatu model prosodi. Prosodi bersifat sangat

spesifik untuk setiap bahasa, sehingga model yang diperlukan untuk membangkitkan data-data prosodi menjadi sangat spesifik juga untuk suatu bahasa. Beberapa model umum prosodi pernah dikembangkan, tetapi untuk digunakan pada suatu bahasa masih perlu banyak penyesuaian yang harus dilakukan.

Konverter fonem ke ucapan berfungsi untuk membangkitkan sinyal ucapan berdasarkan kode-kode fonem yang dihasilkan dari proses sebelumnya. Sub sistem ini harus memiliki pustaka setiap unit ucapan dari suatu bahasa. Pada sistem yang menggunakan teknik *diphone concatenation*, sistem harus didukung oleh suatu *diphone database* yang berisi rekaman segmen-segmen ucapan yang berupa diphone. Ucapan dalam suatu bahasa dibentuk dari satu set bunyi yang mungkin berbeda untuk setiap bahasa, oleh karena itu setiap bahasa harus dilengkapi dengan diphone database yang berbeda.

III. METODE PENELITIAN

A. Waktu dan Tempat Penelitian

Penelitian ini dilaksanakan dari bulan Desember 2008 sampai dengan bulan Mei 2009. Proses perancangan, pembuatan dan pengujian dilakukan di Laboratorium Teknik Digital Jurusan Teknik Elektro Fakultas Teknik Universitas Lampung.

B. Alat dan Bahan

Alat yang digunakan dalam pengerjaan tugas akhir ini adalah:

1. Komputer Intel Pentium IV 3.0 GHz RAM 2 GB
2. HDD 80 GB untuk media penyimpanan data.
3. Perangkat lunak Microsoft Visual Basic 6.0 sebagai *tools* dalam pembuatan sistem.
4. Perangkat lunak MySQL 4.0.18 sebagai *database*.
5. Perangkat lunak MyODBC-3.51.03 sebagai driver MySQL ke Visual Basic 6.0.
6. SQLyog Enterprises sebagai perangkat lunak yang digunakan untuk memanipulasi *database*.
7. *Switch* sebagai penghubung antar PC.

C. Prosedur Kerja

Dalam penyelesaian tugas akhir ini ada beberapa langkah kerja yang dilakukan untuk mencapai hasil akhir yang diinginkan, di antaranya :

1. Studi Literatur

Studi literatur dimaksudkan untuk mempelajari berbagai sumber referensi (buku dan internet) yang berkaitan dengan perancangan dan pembuatan sistem. Literatur yang dipelajari adalah literatur yang berkaitan dengan :

- a. Cara kerja ATIS.
- b. Bahasa pemrograman Visual basic 6.0.
- c. *Client – server* dalam Visual Basic 6.0.
- d. Konsep sistem informasi.

2. Pembuatan Sistem

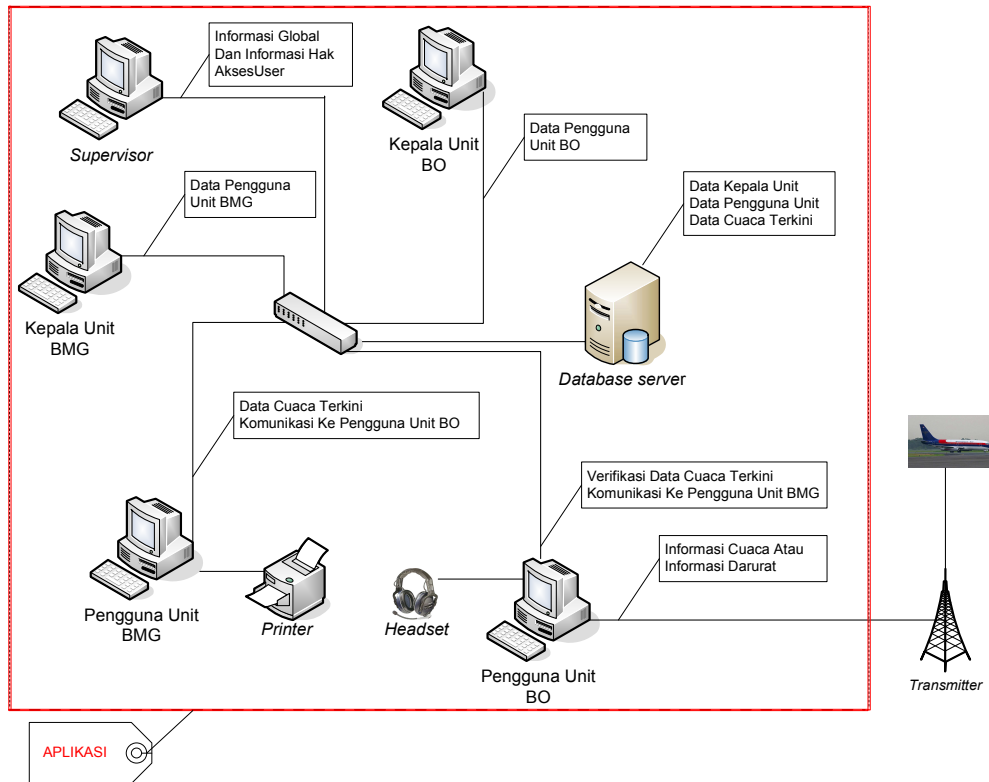
Dalam tahap ini dilakukan realisasi dari perancangan yang telah dibuat. Dimana pembuatan sistem meliputi proses pemrograman perangkat lunak.

3. Pengujian Sistem

Pada tahap ini dilakukan pengujian terhadap program yang telah dibuat dengan cara memutar data cuaca yang telah dimasukan menjadi keluaran berupa suara.

Gambar di bawah ini akan menggambarkan blok diagram secara umum aplikasi

P-ATIS yang akan dibuat.

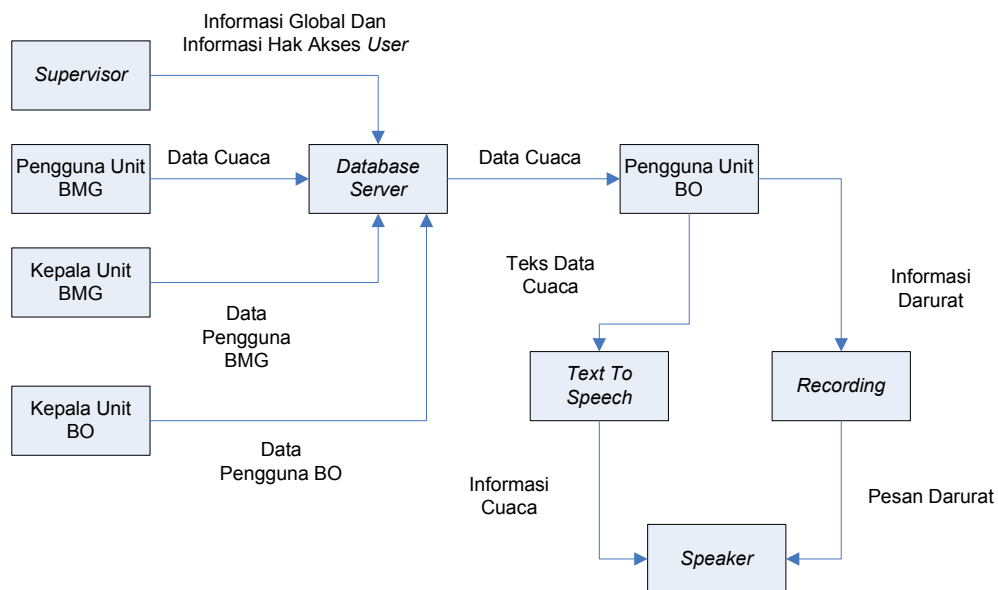


Gambar 3.1. Diagram Aplikasi Secara Umum

Dari gambar dapat dijelaskan bahwa Kepala Unit BMG akan memasukkan data Pengguna Unit BMG ke dalam aplikasi, Pengguna Unit BMG inilah yang mempunyai otoritas untuk memasukkan data-data cuaca terkini ke dalam aplikasi. Pengguna Unit BMG juga dapat membuat laporan data cuaca. Sedangkan Kepala Unit BO mempunyai tugas utama untuk memasukkan data Pengguna Unit BO ke dalam aplikasi. Pengguna Unit BO hanya bertugas mengirim informasi cuaca terkini ke pesawat terbang. Selain itu Pengguna Unit BO juga dapat mengirim pesan darurat mengenai hal yang berkaitan dengan bandara dan penerbangan. *Supervisor* adalah *user* yang mempunyai hak penuh untuk mengatur dan

membatasi hak akses kepala unit dan pengguna unit dalam menggunakan perangkat lunak. *Supervisor* juga bertugas memasukan informasi global bandara. Sistem yang dirancang menggunakan konsep sistem informasi yang terhubung melalui suatu jaringan LAN. Dalam perancangan perangkat lunak ini, P-ATIS akan menggunakan sistem informasi dengan kategori *on-line system*, menggunakan jaringan LAN. P-ATIS juga berbasis *client / server*, dengan sisi *server* pada unit BMG.

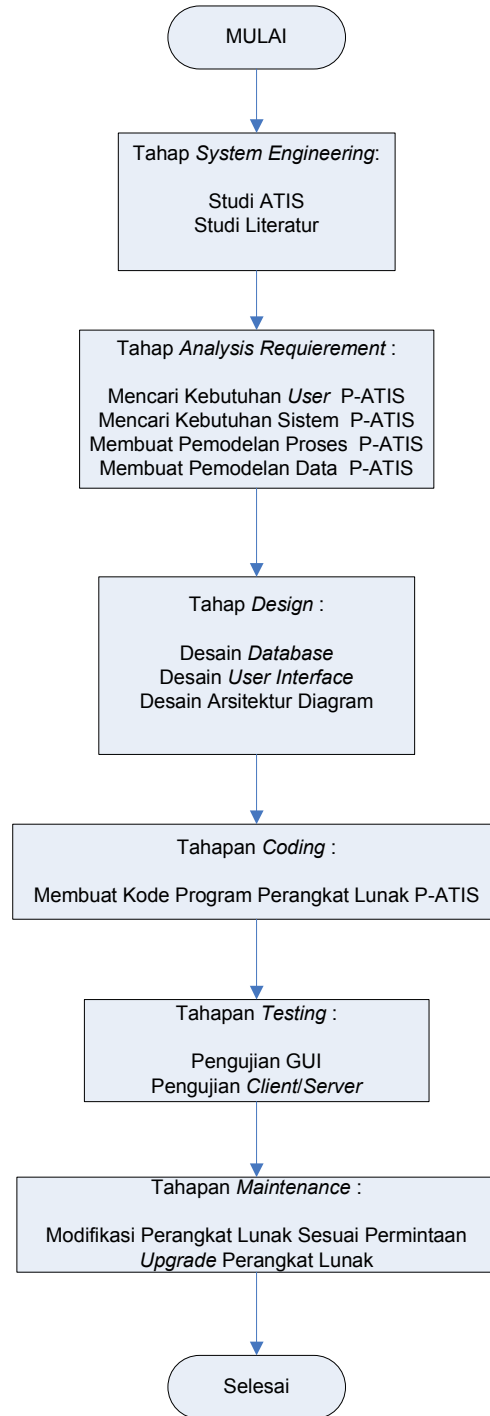
Gambar dibawah adalah blok diagram prosedur kerja aplikasi P-ATIS.



Gambar 3.2. Blok Diagram Prosedur Kerja Aplikasi

D. Perancangan Perangkat Lunak P-ATIS

Perangkat lunak P-ATIS ini dibuat menggunakan model perancangan perangkat lunak *waterfall*. Proses perancangan perangkat lunak melalui metode ini melewati beberapa tahap, yaitu, tahap *system engineering*, tahap *requirement analysis*, tahap *design*, tahap *coding*, tahap *testing*, dan tahap *maintenance*. Perangkat lunak ini berkonsep sistem informasi yang dikelola oleh beberapa pengguna.



Gambar 3.3. Diagram Alir Pembuatan Perangkat Lunak

IV. HASIL DAN PEMBAHASAN

A. Perancangan

1. Tahap *System Engineering*

a. Ruang Lingkup (*Scope*) Perangkat Lunak P-ATIS

P-ATIS memproduksi suara dari data-data cuaca terkini yang berupa teks sebagai informasi cuaca terbaru. Data-data yang berupa teks tersebut dirubah ke bentuk ucapan oleh P-ATIS. Data-data yang digunakan diambil dari *database*. Data-data cuaca ini dimasukkan oleh Pengguna Unit BMG, sedangkan yang mempunyai akses untuk menjalankan fasilitas pengaktifan suara adalah Pengguna Unit BO. P-ATIS memberikan hak akses yang berbeda untuk kedua unit tersebut melalui *supervisor*. *Supervisor* memberikan hak akses penuh berupa Kepala Unit yang mengelola pengguna Unit dari kedua unit tersebut. *Supervisor* juga yang mengatur hak akses dari Pengguna Unit dari kedua unit tersebut dalam menggunakan perangkat lunak P-ATIS. Selain itu, tugas utama *supervisor* adalah memasukan informasi global bandara pengguna perangkat lunak. *Supervisor* juga dapat mengelola secara global user dari kedua unit tersebut. Data-data cuaca terkini yang telah dimasukkan oleh Pengguna Unit BMG selanjutnya akan

diolah oleh Pengguna Unit BO sebagai sebuah informasi cuaca terkini. Pengguna Unit BMG akan memasukkan data cuaca terkini. Pengguna Unit BMG juga dapat membuat laporan berkala dari data-data cuaca yang telah disimpan dalam *database*.

Pengguna Unit BO menggunakan data cuaca yang paling terbaru yang telah dimasukan Pengguna Unit BMG. Selanjutnya data cuaca tersebut akan diformat oleh Pengguna Unit BO sebagai informasi METAR. Pengguna Unit BO juga dapat membuat laporan berkala data METAR. Informasi cuaca terus diinformasikan berulang-ulang selama kurang-lebih 10 menit atau ketika cuaca ekstrem mengalami perubahan. Pengguna Unit BMG menggunakan fasilitas *chatting* untuk memberitahukan perihal tanggal dan waktu untuk data cuaca terkini kepada Pengguna Unit BO. Ketika terjadi suatu keadaan darurat, Pengguna unit BO dapat menggunakan fasilitas perekam yang ada di dalam P-ATIS untuk merekam informasi darurat tersebut, diluar informasi cuaca, dan P-ATIS akan menginformasikan secara berulang-ulang sampai Pengguna Unit BO menyatakan perubahan kondisi lainnya. P-ATIS juga dapat memberikan informasi siapa saja user yang sedang bertugas.

b. Dekomposisi Ruang Lingkup

1. Memproduksi suara. Suara yang diproduksi ini diperdengarkan secara berulang-ulang sebagai suatu informasi cuaca terkini atau sebagai informasi darurat (SRS-P-ATIS-01).

2. Pengolahan data-data cuaca. Data cuaca yang diolah merupakan data cuaca terbaru setiap saat (SRS-P-ATIS-02).
3. Otentifikasi Pengguna Unit. Membatasi fasilitas yang dapat digunakan oleh pengguna dalam unit yang berbeda (SRS-P-ATIS-03).
4. Otentifikasi Kepala Unit. Memberikan akses penuh untuk mengelola pengguna di tiap unit, tiap unit memiliki Kepala Unit yang berbeda (SRS-P-ATIS-04).
5. Pembuatan laporan data cuaca berkala. Data-data yang telah dimasukkan dapat dibuat sebagai suatu laporan (SRS-P-ATIS-05).
6. Merekam Suara. Suara yang direkam merupakan suatu informasi darurat (SRS-P-ATIS-06).
7. Komunikasi antar unit. Komunikasi dilakukan untuk memberitahukan tentang adanya informasi data cuaca terkini yang ada (SRS-P-ATIS-07).
8. Otentifikasi *Supervisor*. Mengatur, mengelola dan membatasi hak akses kepala unit dan pengguna unit dalam menggunakan perangkat lunak, memasukan informasi global bandara (SRS-P-ATIS-08).
9. Pengawasan. Untuk mengetahui siapa saja Pengguna Unit masing-masing unit yang sedang bertugas (SRS-P-ATIS-09).

10. Pembuatan laporan data METAR berkala. Data-data yang telah dimasukkan dapat dibuat sebagai suatu laporan (SRS-P-ATIS-10).

2. Tahap *Requirement Analysis*

a. Kebutuhan Fungsional

1. Karakteristik Pengguna

Tabel 4.1 Kategori Pengguna

Kategori Pengguna	Tugas	Hak Akses Ke Aplikasi
<i>Supervisor</i>	Mengatur, mengelola dan membatasi hak akses Kepala Unit dan Pengguna Unit. Memasukan informasi global bandara.	SRS-P-ATIS-08
Kepala Unit BMG	Mengelola pengguna Unit BMG, pengawasan.	SRS-P-ATIS-04 SRS-P-ATIS-09
Kepala Unit BO	Mengelola pengguna Unit BO, pengawasan.	SRS-P-ATIS-04 SRS-P-ATIS-09
Pengguna Unit BMG	Mengolah data cuaca terkini, membuat laporan berkala data cuaca, berkomunikasi dengan pengguna Unit BO, pengawasan.	SRS-P-ATIS-02 SRS-P-ATIS-03 SRS-P-ATIS-05 SRS-P-ATIS-07 SRS-P-ATIS-09
Pengguna Unit BO	Mengaktifkan produksi suara dari aplikasi, merekam informasi darurat, berkomunikasi dengan pengguna Unit BMG, pengawasan, membuat laporan berkala data METAR.	SRS-P-ATIS-01 SRS-P-ATIS-03 SRS-P-ATIS-06 SRS-P-ATIS-07 SRS-P-ATIS-09 SRS-P-ATIS-10

b. Deskripsi Proses

1. Pengguna

a. *Supervisor*

1. Login ke aplikasi

Untuk dapat menjalankan tugasnya, *Supervisor* harus *login* terlebih dahulu ke dalam aplikasi untuk membuka *form* kerjanya.

2. Mengelola hak akses dan informasi global bandara

Supervisor bertugas mengatur dan membatasi hak akses apa saja yang dapat dilakukan oleh Kepala Unit dan Pengguna Unit. Memberikan *user ID default* dan *password default* untuk Kepala Unit. Memasukan informasi global bandara.

3. Melihat informasi *user* dan *weather*

Supervisor dapat melihat seluruh data Kepala Unit, Pengguna Unit, dan data cuaca.

4. *Logout* dari aplikasi

Untuk menghentikan semua proses penggunaan aplikasi, *Supervisor* harus *logout* dari aplikasi.

b. Kepala Unit

1. Kepala Unit BMG

a. *Login* ke aplikasi

Untuk dapat menjalankan tugasnya, Kepala Unit BMG harus *login* terlebih dahulu ke dalam aplikasi untuk membuka *form* kerjanya.

b. Mengelola pengguna Unit BMG

Kepala Unit BMG memasukan data pengguna Unit BMG, data yang dimasukan adalah nama lengkap, *user ID default* dan *password default*. Untuk memberikan pengguna Unit BMG akses ke aplikasi.

c. Melihat informasi Pengguna Unit BMG dan cuaca

Kepala Unit BMG dapat melihat seluruh data cuaca dan melihat Pengguna Unit BMG yang sedang menggunakan P-ATIS

d. *Logout* dari aplikasi

Untuk menghentikan semua proses penggunaan aplikasi, Kepala Unit BMG harus *logout* dari aplikasi.

2. Kepala Unit BO

a. *Login* ke aplikasi

Untuk dapat menjalankan tugasnya, Kepala Unit BO harus *login* terlebih dahulu ke dalam aplikasi untuk membuka *form* kerjanya.

b. Mengelola pengguna Unit BO

Kepala Unit BO memasukan data pengguna Unit BO, data yang dimasukan adalah nama lengkap, *user ID default* dan *password default*. Untuk memberikan pengguna Unit BO akses ke aplikasi.

c. Melihat informasi Pengguna Unit BO dan METAR

Kepala Unit BO dapat melihat seluruh data METAR dan melihat Pengguna Unit BO yang sedang menggunakan P-ATIS

d. *Logout* dari aplikasi

Untuk menghentikan semua proses penggunaan aplikasi, Kepala Unit BO harus *logout* dari aplikasi.

c. Pengguna Unit

1 . Pengguna Unit BMG

a. *Login* ke aplikasi

Untuk dapat menjalankan tugasnya sebagai pengguna, pengguna Unit BMG harus *login* terlebih dahulu ke dalam aplikasi untuk membuka *form* penggunanya.

b. Mengelola data cuaca terkini

Pengguna Unit BMG bertugas mengelola data cuaca terkini, yaitu memasukan data cuaca terkini ke dalam *database* dan melihat data cuaca yang telah dimasukan.

c. Membuat laporan berkala

Pengguna Unit BMG juga bertugas membuat laporan berkala data cuaca, secara rutin setiap harinya untuk dilaporkan.

d. Berkomunikasi dengan Pengguna Unit BO

Setelah mengelola data cuaca terbaru, pengguna Unit BMG harus bergegas menginformasikan kepada Pengguna Unit BO.

Informasi yang disampaikan meliputi tanggal dan waktu data cuaca terbaru. Komunikasi dilakukan melalui fasilitas *chatting* dari aplikasi.

e. Melihat informasi cuaca

Pengguna Unit BMG dapat melihat seluruh data cuaca yang tersimpan dalam *database*.

f. Melihat Pengguna Unit Aktif

Pengguna Unit BMG dapat memperoleh informasi siapa Pengguna Unit BO yang sedang bertugas

g. *Logout* dari aplikasi

Untuk menghentikan semua proses penggunaan aplikasi, pengguna Unit BMG harus *logout* dari aplikasi.

2. Pengguna Unit BO

a. *Login* ke aplikasi

Untuk dapat menjalankan tugasnya sebagai pengguna, Pengguna Unit BO harus *login* terlebih dahulu ke dalam aplikasi untuk membuka *form* penggunanya.

b. Mengaktifkan Suara

Pengguna Unit BO bertanggung jawab mengaktifkan aplikasi suara dari P-ATIS. Setelah menerima informasi tanggal dan waktu terkini data cuaca, Pengguna Unit BO dapat mengambil data cuaca terkini dari *database*. Data berupa teks itu kemudian dirubah menjadi suara oleh aplikasi.

c. Merekam suara

Pengguna Unit BO juga dapat merekam suaranya sendiri sebagai pesan atau informasi darurat jika dibutuhkan, aplikasi akan memproduksi rekamannya secara berulang-ulang.

d. Berkomunikasi dengan Pengguna Unit BMG

Pengguna Unit BO harus selalu siaga memperhatikan informasi cuaca terbaru dari Pengguna Unit BMG. Komunikasi dilakukan melalui fasilitas *chatting* dari aplikasi.

e. Melihat informasi cuaca dan METAR

Pengguna Unit BO dapat melihat seluruh data cuaca yang tersimpan dalam *database* dan data METAR yang tersimpan.

f. Melihat Pengguna Unit Aktif

Pengguna Unit BO dapat memperoleh informasi siapa Pengguna Unit BMG yang sedang bertugas

g. *Logout* dari aplikasi

Untuk menghentikan semua proses penggunaan aplikasi, Pengguna Unit BO harus *logout* dari aplikasi.

2. Perangkat Lunak

a. Aplikasi dapat memproduksi suara

1. Pengguna Unit BO dapat mengaktifkan aplikasi suara dari aplikasi.
2. Pengguna Unit BO dapat meng-*update* informasi cuaca bila terjadi perubahan data cuaca secara ekstrem.
3. Pengguna Unit BO dapat memproduksi suara dari pesan atau informasi darurat.

b. Aplikasi dapat menangani pengolahan data cuaca

1. Pengguna Unit BMG dapat memasukan data cuaca terkini yaitu, *wind speed, wind direction, visibility, temperature, QNH, height of clouds, feet, present weather, DEW point, Date, Time.*

2. Pengguna Unit BMG dapat menampilkan data cuaca pada waktu tertentu.
 3. Pengguna Unit BO dapat melakukan pencarian dan menampilkan data cuaca pada waktu tertentu. Pengguna Unit BO menggunakan tanggal dan waktu data cuaca terkini sebagai kunci untuk melakukan pencarian.
 4. Pengguna Unit BO dapat memasukan data METAR yaitu, *Date, Time, METAR.*
- c. Aplikasi dapat membantu membuat laporan berkala data cuaca
1. Pengguna Unit BMG dapat membuat laporan berkala harian data data cuaca secara rutin.
- d. Aplikasi menyediakan fasilitas perekaman suara
1. Pengguna Unit BO dapat melakukan perekaman suaranya sebagai informasi darurat.
- e. Aplikasi menyediakan fasilitas komunikasi
1. Pengguna Unit BO dapat berkomunikasi dengan pengguna Unit BMG.
 2. Pengguna Unit BMG dapat berkomunikasi dengan pengguna Unit BO.

3. Kebutuhan Antarmuka Eksternal

a. Antarmuka Pemakai

Antarmuka pemakai akan dikembangkan dengan modus grafik.

P-ATIS menerima masukan dari pemakai melalui perintah yang diklik pada *mouse* atau yang diketikan pada *keyboard*. Keluaran

dapat dilihat pemakai dengan menggunakan monitor.

Laporan dapat dicetak melalui *printer*.

b. Antarmuka Perangkat Keras

1. *Personal Computer (PC)*.

2. *Keyboard*.

3. *Mouse*.

4. *Printer*.

5. *Headset*.

c. Antarmuka Perangkat Lunak

Library yang dibutuhkan adalah MyODBC, sebagai *driver* untuk

Database MySQL, *database MySQL* sebagai perangkat lunak *service*

Database, *ActiveX controls agent SAPI* sebagai *library text to*

speech pada visual basic, SDK 5.1 sebagai *voice library* pada

sistem operasi Windows.

d. Antarmuka Komunikasi

Yang dibutuhkan adalah sebuah komputer sebagai *server* dan sebuah komputer sebagai *client*. Menggunakan protokol TCP/IP

e. Kebutuhan Non-Fungsional

Tabel 4.2 Kebutuhan Non-Fungsional

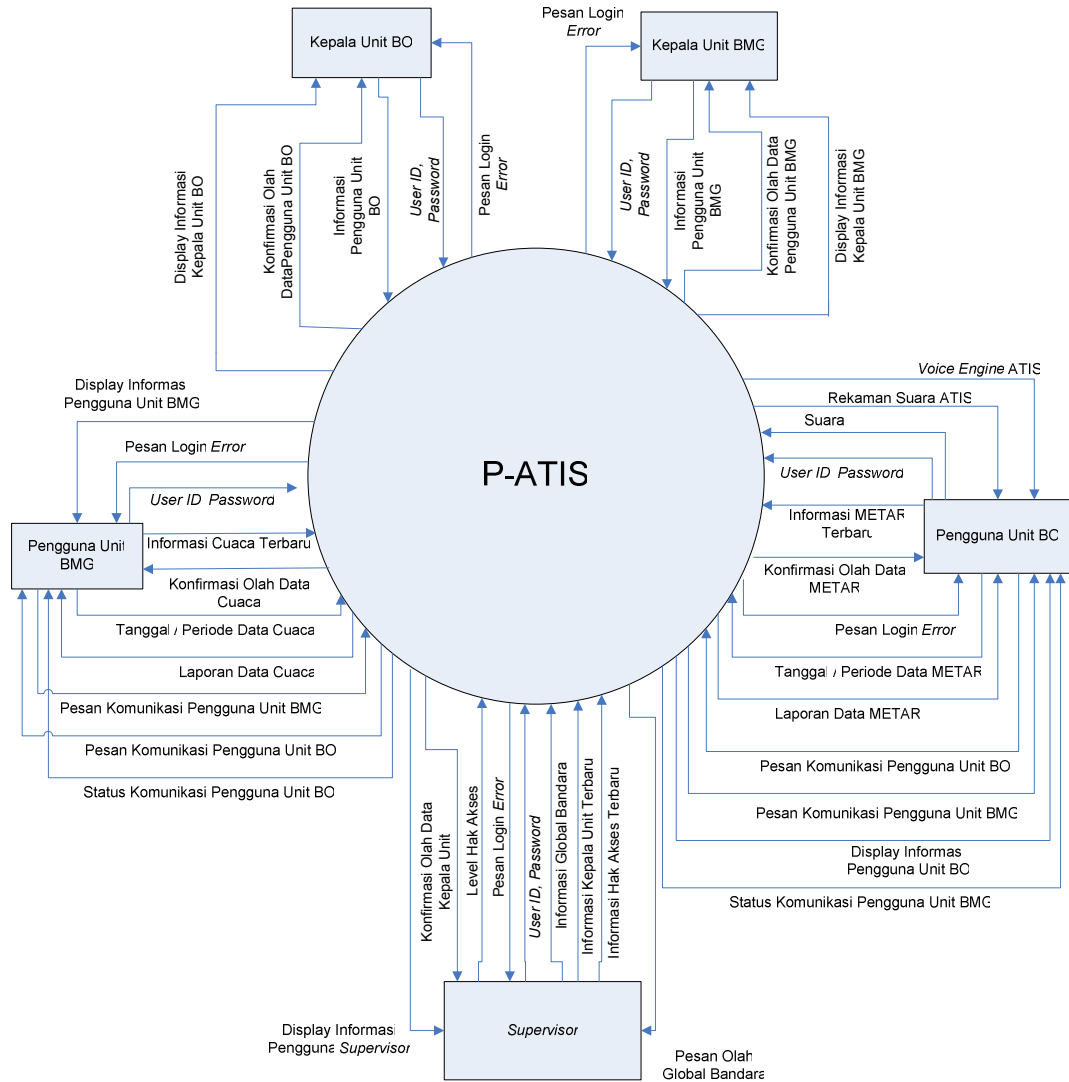
SRS_ID	Parameter	Requirement
SRS-P-ATIS-08	Availability	24 jam sehari, 7 hari seminggu
SRS-P-ATIS-09	Portability	Mudah diadopsi pada lingkungan sistem operasi windows dan DBMS MySql
SRS-P-ATIS-10	Komunikasi	Bahasa indonesia

f. Batasan Perancangan

P-ATIS hanya dapat dijalankan pada sistem operasi windows, dan DBMS MySql. Modifikasi pengguna BO dan pengguna BMG hanya dapat dilakukan administrator masing-masing unit.

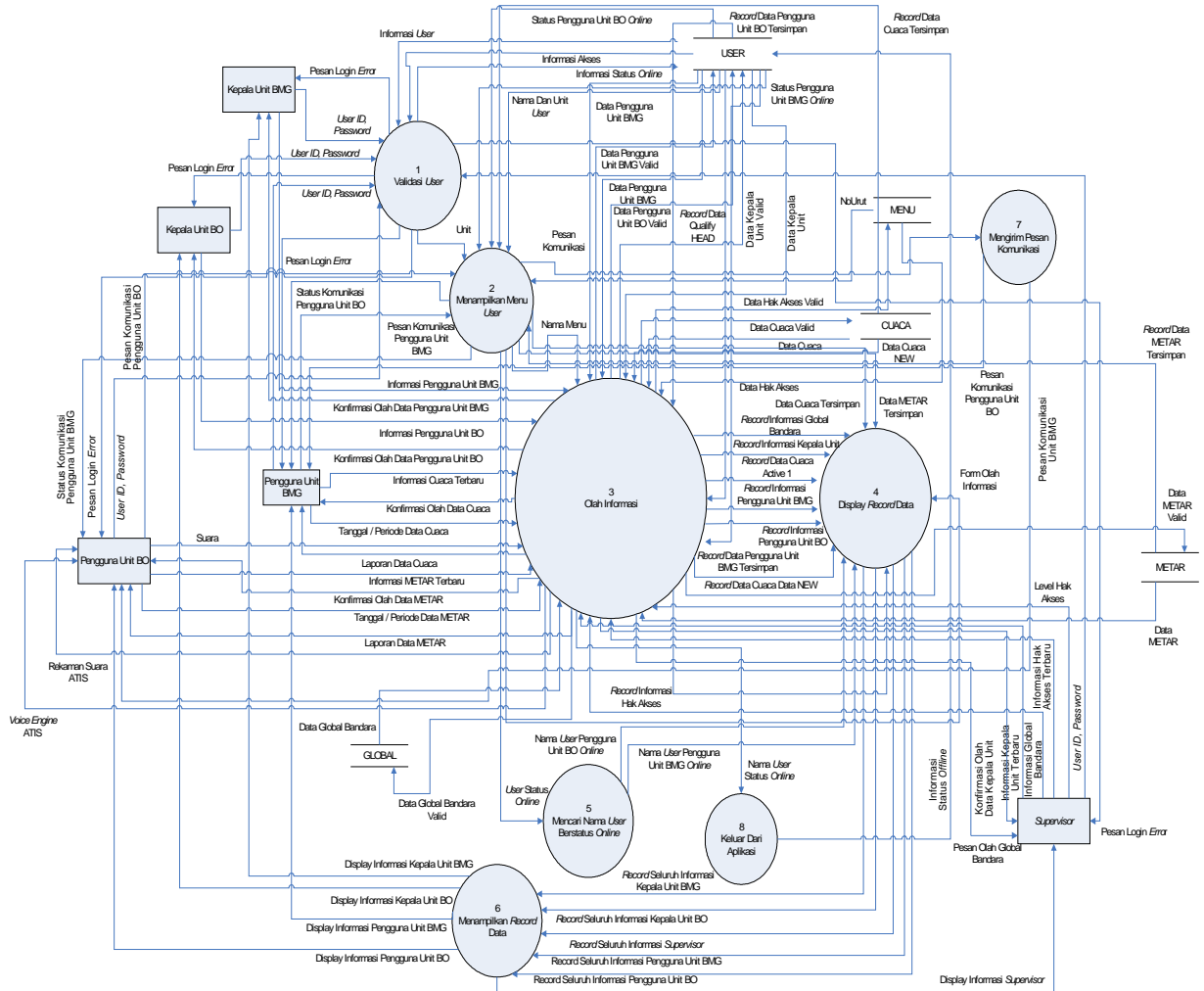
4. Pemodelan Proses

a. Data Context Diagram (DCD)

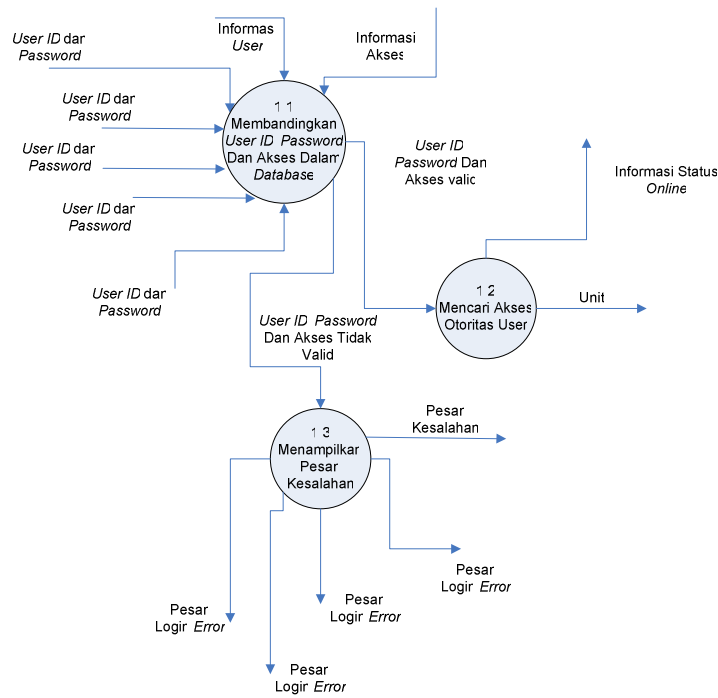


Gambar 4.1 DCD P-ATIS

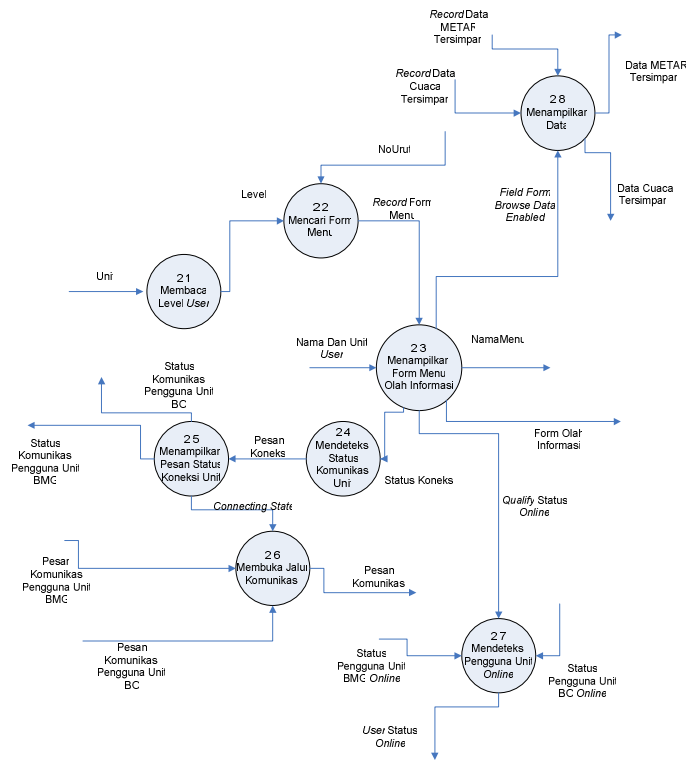
b. Data Flow Diagram (DFD)



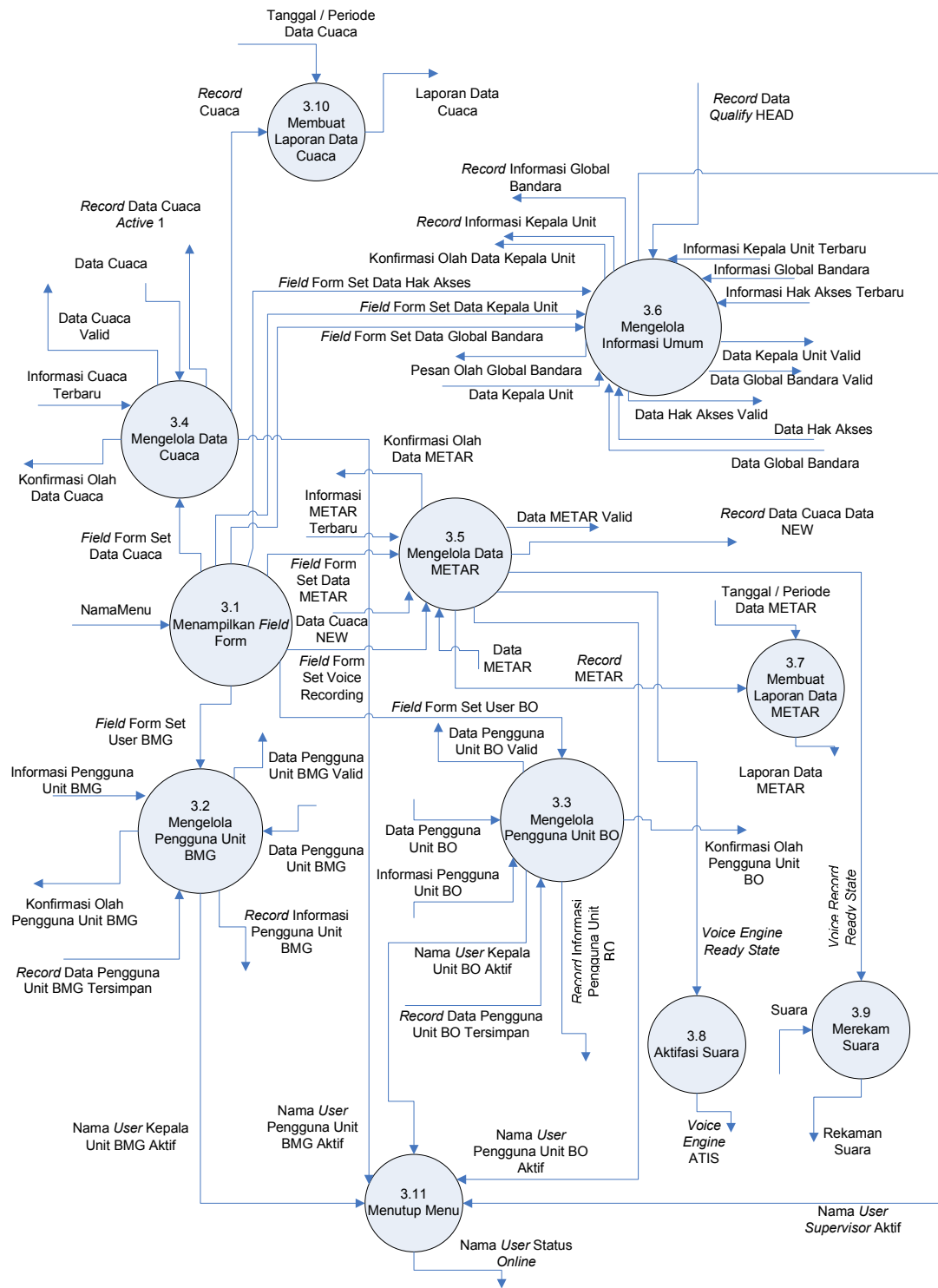
Gambar 4.2 DFD Level 1 P-ATIS



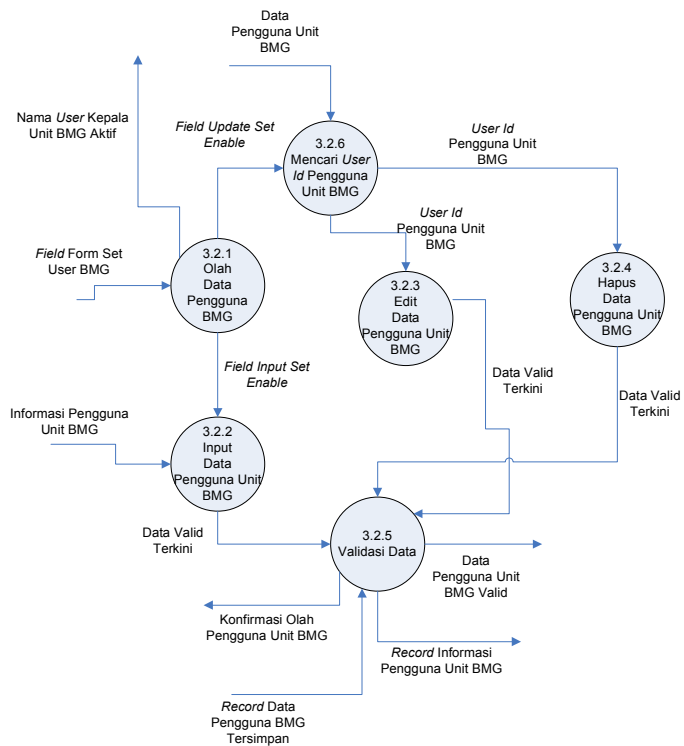
Gambar 4.3 DFD Level 2 Proses 1 P-ATIS



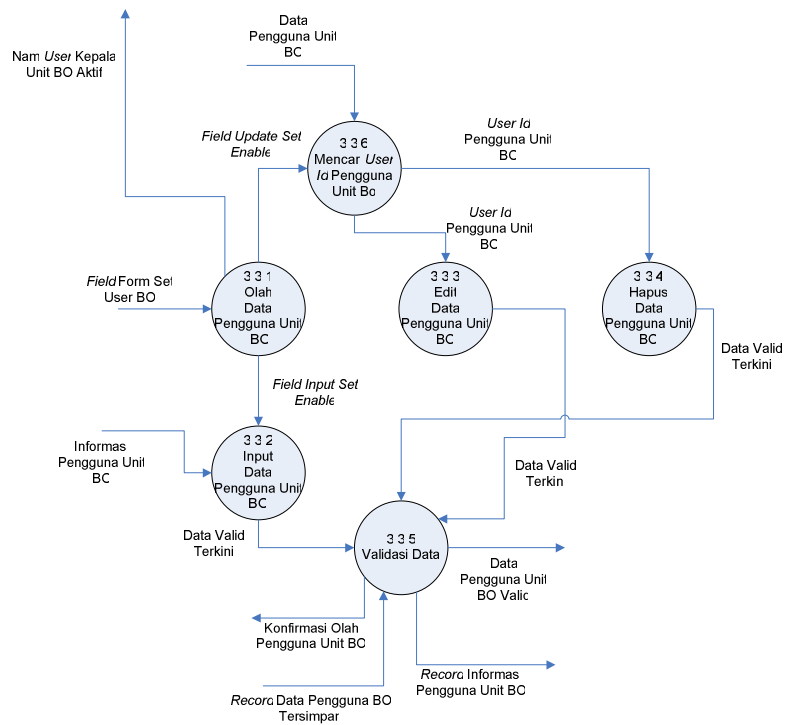
Gambar 4.4 DFD Level 2 Proses 2 P-ATIS



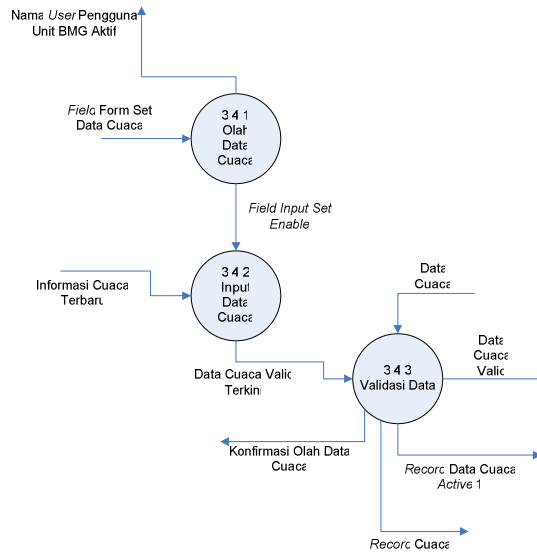
Gambar 4.5 DFD Level 2 Proses 3 P-ATIS



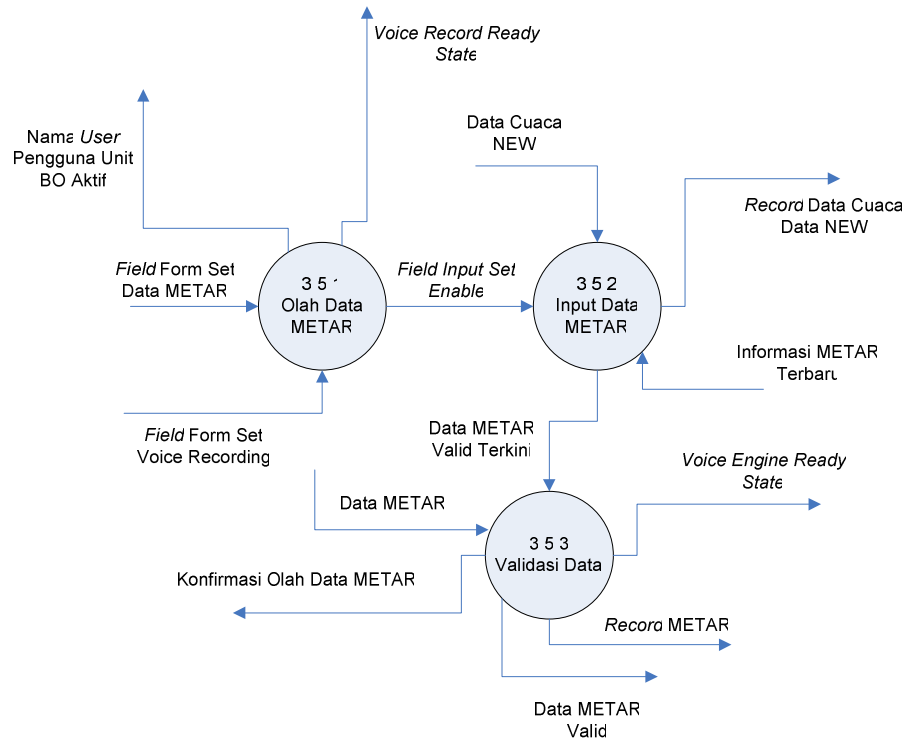
Gambar 4.6 DFD Level 3 Proses 3.2 P-ATIS (Aliran Proses Kepala Unit BMG)



Gambar 4.7 DFD Level 3 Proses 3.3 P-ATIS (Aliran Proses Kepala Unit BO)



Gambar 4.8 DFD Level 3 Proses 3.4 P-ATIS (Aliran Proses Pengguna Unit BMG)



Gambar 4.9 DFD Level 3 Proses 3.5 P-ATIS (Aliran Proses Supervisor)

serta akses yang dikenakan oleh *user* tersebut dari data di dalam *database*. *User ID*, *Password*, Dan Akses sebagai kunci unik *user* untuk menggunakan aplikasi. Proses membandingkan *User ID* dan *Password* dengan informasi *User* yang tersimpan dalam *database*. Kemudian proses akan membandingkan akses *user* dengan informasi akses yang tersimpan, apakah *user* diizinkan untuk menggunakan aplikasi. Jika nilai ketiga kunci ini sesuai dengan informasi *user* dan informasi akses yang tersimpan, maka *User ID*, *Password* dan Akses *user* tersebut valid, dan *user* tersebut dapat menggunakan aplikasi. Jika ketiga kunci ini tidak sesuai dengan informasi *user* dan informasi akses yang tersimpan, maka *User ID*, *Password* dan Akses *user* tersebut tidak valid, dan *user* tersebut tidak dapat menggunakan aplikasi.

2. P-SPEC 1.3

Masukan : *User ID, Password* Dan Akses Valid

Keluaran : Unit

Informasi Status *Online*

Algoritma :

Proses Mencari Akses Otoritas *user* menerima *User ID, Password* Dan Akses Valid. Proses selanjutnya secara otomatis memperbaharui *database*, yaitu merubah informasi status *user* yang bersangkutan tersebut menjadi *Online*. Proses selanjutnya menandakan *user* yang bersangkutan tersebut bertugas dalam unit apa, unit BMG, unit BO, supervisor, head of BMG, atau head of BO dalam menu masing-masing *user* tersebut.

3. P-SPEC 1.4

Masukan : *User ID, Password* Dan Akses Tidak Valid

Keluaran : Pesan Login *Error*

Algoritma :

Proses Menampilkan Pesan Kesalahan menerima *User ID, Password* Dan Akses Tidak Valid. Proses ini selanjutnya menampilkan Pesan Login *Error*.

4. P-SPEC 2.1

Masukan : Unit

Keluaran : Level

Algoritma :

Proses Membaca Level *User* ini menerima unit, proses selanjutnya akan membaca level *user* yang bersangkutan berdasarkan unit. Jika kepala unit BMG, maka level *user* tersebut adalah 3, jika kepala unit BO, maka level *user* tersebut adalah 4, Jika pengguna unit BMG, maka level *user* tersebut adalah 5, jika pengguna unit BO, maka level *user* tersebut adalah 6, jika *supervisor*, maka level *user* tersebut adalah 2.

5. P-SPEC 2.2

Masukan : Level

NoUrut

Keluaran : *Record* Form Menu

Algoritma :

Proses Mencari Form Menu menerima Level dari *user* yang bersangkutan, selanjutnya proses akan mencari ke dalam *database* NoUrut dari masing-masing form kerja yang diizinkan diakses oleh *user* berdasarkan level dari *user*. Kumpulan dari NoUrut form kerja yang unik

tersebut merupakan *Record Form Menu* yang merupakan area kerja dari *user* yang bersangkutan.

6. P-SPEC 2.3

Masukan : *Record Form Menu*

Nama Dan Unit *User*

Keluaran : NamaMenu

Status Koneksi

Qualify Status Online

Form Olah Informasi

Field Form Browse Data Enabled

Algoritma :

Proses Menampilkan Form Menu Olah Informasi menerima *Record Form Menu*, selanjutnya proses akan menampilkan NamaMenu sebagai field form kerja pada menu masing-masing *user*. Secara otomatis proses akan mencari dalam *database* yaitu Nama Dan Unit *User* yang bertugas untuk ditampilkan dalam menu *user* tersebut. proses juga secara otomatis mengaktifkan status koneksi *user* tersebut, pengguna unit BMG atau pengguna unit BO. Proses juga secara otomatis menampilkan *Field Form Browse Data Enabled* yang berisi data cuaca untuk kepala unit BMG, pengguna unit BMG dan pengguna unit BO, serta data METAR untuk kepala unit BO dan

pengguna unit BO. Proses juga secara otomatis melakukan query *Qualify Status Online* untuk mencari pengguna unit BO atau pengguna unit BMG yang sedang menggunakan aplikasi. Proses juga secara otomatis memilah informasi yang dikelola masing-masing *user* melalui Form Olah Informasi.

7. P-SPEC 2.4

Masukan : Status Koneksi

Keluaran : Pesan Koneksi

Algoritma :

Proses Mendeteksi Status Komunikasi Unit menerima Status Koneksi dari pengguna unit BMG dan pengguna unit BO. Proses selanjutnya mendeteksi port komunikasi antara keduanya apakah terkoneksi atau tidak hasil pendeteksian itu akan menjadi parameter Pesan Koneksi.

8. P-SPEC 2.5

Masukan : Pesan Koneksi

Keluaran : Status Komunikasi Pengguna Unit BMG

Status Komunikasi Pengguna Unit BO

Connecting State

Algoritma :

Proses Menampilkan Pesan Status Koneksi Unit menerima Pesan Komunikasi sebagai parameter koneksi. Proses selanjutnya menampilkan status komunikasi pengguna unit BO kepada pengguna unit BMG dan status komunikasi pengguna unit BMG kepada pengguna unit BO, jika status “Connected To Host”, berarti kedua pengguna unit terkoneksi. Jika status “Tidak Terkoneksi”, berarti kedua unit tidak terkoneksi. Proses selanjutnya akan menyediakan suatu sesi komunikasi berupa *connecting state* yang artinya kedua unit dapat saling berkomunikasi.

9. P-SPEC 2.6

Masukan : *Connecting State*

Pesan Komunikasi Pengguna Unit BMG

Pesan Komunikasi Pengguna Unit BO

Keluaran : Pesan Komunikasi

Algoritma :

Proses Membuka Jalur Komunikasi menerima *Connecting State* sebagai sesi untuk saling berkomunikasi. Proses selanjutnya dapat menerima pesan komunikasi pengguna unit BMG dan pengguna

unit BO. Kedua pesan tersebut akan dikirim dalam suatu sesi sebagai pesan komunikasi.

10. P-SPEC 2.7

Masukan : *Qualify Status Online*

Status Pengguna Unit BMG *Online*

Status Pengguna Unit BO *Online*

Keluaran : *User Status Online*

Algoritma :

Proses mendeteksi pengguna unit *online* menerima *query* *Qualify Status Online*. Proses selanjutnya akan mencari dalam *database* Status Pengguna Unit BMG *Online* dan Status Pengguna Unit BO *Online* kedua hasil proses pencarian itu akan menjadi parameter *User Status Online*, yaitu pengguna unit yang sedang menggunakan aplikasi.

11. P-SPEC 2.8

Masukan : *Field Form Browse Data Enabled*

Record Data METAR Tersimpan

Record Data Cuaca Tersimpan

Keluaran : Data METAR Tersimpan

Data Cuaca Tersimpan

Algoritma :

Proses menampilkan data menerima *Field Form Browse Data Enabled* yang secara otomatis aktif dalam menu. Proses selanjutnya melakukan pencarian dalam database *Record Data METAR Tersimpan* dan *Record Data Cuaca Tersimpan*. Hasil dari pencarian itu adalah Data Cuaca Tersimpan yang akan otomatis muncul dalam menu pengguna unit BMG, pengguna unit BO dan kepala unit BMG. Kemudian Data METAR Tersimpan yang akan otomatis muncul dalam menu pengguna unit BO dan kepala unit BO.

12. P-SPEC 3.1

Masukan : NamaMenu

Keluaran : *Field Form Set User BMG*

Field Form Set User BO

Field Form Set Data METAR

Field Form Set Data Cuaca

Field Form Set Data Kepala Unit

Field Form Set Data Global Bandara

Field Form Set Data Hak Akses

Field Form Set Voice Recording

Algoritma :

Proses menampilkan *field* form menerima NamaMenu sebagai parameter form olah yang akan ditampilkan dalam menu masing-masing *user*. Jika *user* adalah *supervisor*, maka *Field* Form Set Data Kepala Unit, *Field* Form Set Data Global Bandara, *Field* Form Set Data Hak Akses akan tampil dalam menu *supervisor*. Jika *user* adalah Kepala Unit BMG, maka *Field* Form Set User BMG, akan tampil dalam menu Kepala Unit BMG. Jika *user* adalah Kepala Unit BO, maka *Field* Form Set User BO, akan tampil dalam menu Kepala Unit BO. Jika *user* adalah Pengguna Unit BMG, maka *Field* Form Set Data Cuaca, akan tampil dalam menu Pengguna Unit BMG. Jika *user* adalah Pengguna Unit BO, maka *Field* Form Set Data METAR, *Field* Form Set Voice Recording akan tampil dalam menu Pengguna Unit BO.

13. P-SPEC 3.2.1

Masukan : *Field* Form Set User BMG

Keluaran : *Field* Input Set Enable

Field Update Set Enable

Nama *User* Kepala Unit BMG Aktif

Algoritma :

Proses olah data pengguna unit BMG menerima *Field Form Set User BMG*. Jika kepala unit BMG akan memasukan data terbaru pengguna unit BMG, maka *Field Input Set Enable*, artinya kepala unit BMG dapat menambah data. Jika kepala unit BMG ingin mengedit informasi pengguna unit BMG, maka *Field Update Set Enable*, artinya artinya kepala unit BMG dapat merubah data pengguna unit BMG yang tersimpan. Nama *User Kepala Unit BMG Aktif* akan menjadi parameter selama kepala unit BMG ini menggunakan aplikasi.

14. P-SPEC 3.2.2

Masukan : *Field Input Set Enable*

Informasi Pengguna Unit BMG

Keluaran : Data Valid Terkini

Algoritma :

Proses input data pengguna unit BMG menerima *Field Input Set Enable*, maka kepala unit BMG dapat memasukan data pengguna unit terbaru. Kepala unit BMG memasukan rangkaian Informasi Pengguna Unit BMG ke dalam *field* rekam informasi yang tersedia. Rangkaian informasi ini akan menjadi calon Data Valid Terkini yang akan disimpan dalam *database*.

15. P-SPEC 3.2.3

Masukan : *User ID* Pengguna Unit BMG

Keluaran : Data Valid Terkini

Algoritma :

Proses edit data pengguna unit BMG menerima *User ID* Pengguna Unit BMG sebagai parameter pengguna unit BMG yang hendak diedit. Proses selanjutnya akan mengelola rangkaian informasi terbaru ini yang akan menjadi calon Data Valid Terkini yang akan disimpan dalam *database*.

16. P-SPEC 3.2.4

Masukan : *User ID* Pengguna Unit BMG

Keluaran : Data Valid Terkini

Algoritma :

Proses hapus data pengguna unit BMG menerima *User ID* Pengguna Unit BMG sebagai parameter pengguna unit BMG yang hendak dihapus. Proses selanjutnya akan menghapus rangkaian informasi ini dari dalam *database*.

17. P-SPEC 3.2.5

Masukan : Data Valid Terkini

Record Informasi Pengguna Unit BMG Tersimpan

Keluaran : Data Pengguna Unit BMG Valid

Konfirmasi Olah Pengguna Unit BMG

Record Informasi Pengguna Unit BMG

Algoritma :

Proses validasi data menerima data valid terkini yang merupakan informasi yang siap disimpan atau diedit dalam *database*. Proses selanjutnya akan memeriksa, apakah ada informasi yang salah atau terlewat. Jika informasi tidak sesuai, akan muncul pesan Konfirmasi Olah Pengguna Unit BMG yang memberitahu letak kesalahannya. Jika informasi sesuai, akan muncul pesan Konfirmasi Olah Pengguna Unit BMG yang bertanya pada kepala unit BMG apakah akan menyimpan data atau tidak. Jika kepala unit BMG memutuskan untuk menyimpan data, maka data akan tersimpan dalam *database*. Data yang tersimpan dalam *database* merupakan Data Pengguna Unit BMG Valid. Jika tidak, maka data tidak akan tersimpan. Proses juga secara otomatis mencari *Record* Informasi Pengguna Unit BMG

Tersimpan dan mengirim parameter *Record* Informasi Pengguna Unit BMG untuk siap ditampilkan.

18. P-SPEC 3.2.6

Masukan : Data Pengguna Unit BMG

Field Update Set Enable

Keluaran : *User ID* Pengguna Unit BMG

Algoritma :

Proses mencari *user id* pengguna unit BMG menerima *Field Update Set Enable* sebagai pemicu. Selanjutnya proses akan melakukan pencarian Data Pengguna Unit BMG yang tersimpan dalam *database*. Yang menjadi acuan dan parameter pencarian adalah *User ID* Pengguna Unit BMG yang diinginkan.

19. P-SPEC 3.3.1

Masukan : *Field Form Set User BO*

Keluaran : *Field Input Set Enable*

Field Update Set Enable

Nama *User Kepala* Unit BO Aktif

Algoritma :

Proses olah data pengguna unit BO menerima *Field Form Set User BO*. Jika kepala unit BO akan memasukan data terbaru pengguna unit BO, maka *Field Input Set Enable*, artinya kepala unit BO dapat menambah data. Jika kepala

unit BO ingin mengedit informasi pengguna unit BO, maka *Field Update Set Enable*, artinya artinya kepala unit BO dapat merubah data pengguna unit BO yang tersimpan. Nama *User* Kepala Unit BO Aktif akan menjadi parameter selama kepala unit BO ini menggunakan aplikasi.

20. P-SPEC 3.3.2

Masukan : *Field Input Set Enable*

Informasi Pengguna Unit BO

Keluaran : Data Valid Terkini

Algoritma :

Proses input data pengguna unit BO menerima *Field Input Set Enable*, maka kepala unit BO dapat memasukan data pengguna unit terbaru. Kepala unit BO memasukan rangkaian Informasi Pengguna Unit BO ke dalam *field* rekam informasi yang tersedia. Rangkaian informasi ini akan menjadi calon Data Valid Terkini yang akan disimpan dalam *database*.

21. P-SPEC 3.3.3

Masukan : *User ID* Pengguna Unit BO

Keluaran : Data Valid Terkini

Algoritma :

Proses edit data pengguna unit BO menerima *User ID* Pengguna Unit BO sebagai parameter pengguna unit BO yang hendak diedit. Proses selanjutnya akan mengelola rangkaian informasi terbaru ini yang akan menjadi calon Data Valid Terkini yang akan disimpan dalam *database*.

22. P-SPEC 3.3.4

Masukan : *User ID* Pengguna Unit BO

Keluaran : Data Valid Terkini

Algoritma :

Proses hapus data pengguna unit BO menerima *User ID* Pengguna Unit BO sebagai parameter pengguna unit BO yang hendak dihapus. Proses selanjutnya akan menghapus rangkaian informasi ini dari dalam *database*.

23. P-SPEC 3.3.5

Masukan : Data Valid Terkini

Record Informasi Pengguna Unit BO Tersimpan

Keluaran : Data Pengguna Unit BO Valid

Konfirmasi Olah Pengguna Unit BO

Record Informasi Pengguna Unit BO

Algoritma :

Proses validasi data menerima data valid terkini yang merupakan informasi yang siap disimpan atau diedit dalam *database*. Proses selanjutnya akan memeriksa, apakah ada informasi yang salah atau terlewat. Jika informasi tidak sesuai, akan muncul pesan Konfirmasi Olah Pengguna Unit BO yang memberitahu letak kesalahannya. Jika informasi sesuai, akan muncul pesan Konfirmasi Olah Pengguna Unit BO yang bertanya pada kepala unit BO apakah akan menyimpan data atau tidak. Jika kepala unit BO memutuskan untuk menyimpan data, maka data akan tersimpan dalam *database*. Data yang tersimpan dalam database merupakan Data Pengguna Unit BO Valid. Jika tidak, maka data tidak akan tersimpan. Proses juga secara otomatis mencari *Record* Informasi Pengguna Unit BO Tersimpan dan mengirim parameter *Record* Informasi Pengguna Unit BO untuk siap ditampilkan.

24. P-SPEC 3.3.6

Masukan : Data Pengguna Unit BO

Field Update Set Enable

Keluaran : *User ID* Pengguna Unit BO

Algoritma :

Proses mencari *user id* pengguna unit BO menerima *Field Update Set Enable* sebagai pemicu. Selanjutnya proses akan melakukan pencarian Data Pengguna Unit BO yang tersimpan dalam *database*. Yang menjadi acuan dan parameter pencarian adalah *User ID* Pengguna Unit BO yang diinginkan.

25. P-SPEC 3.4.1

Masukan : *Field Form Set Data Cuaca*

Keluaran : *Field Input Set Enable*

Nama *User* Pengguna Unit BMG Aktif

Algoritma :

Proses olah data cuaca menerima *Field Form Set Data Cuaca*. Jika pengguna unit BMG akan memasukan data cuaca terbaru, maka *Field Input Set Enable*, artinya pengguna unit BMG dapat menambah data cuaca baru.. Nama *User* Pengguna Unit BMG Aktif akan menjadi parameter selama pengguna unit BMG ini menggunakan aplikasi

26. P-SPEC 3.4.2

Masukan : *Field Input Set Enable*

Informasi Cuaca Terbaru

Keluaran : Data Cuaca Valid Terkini

Algoritma :

Proses input data cuaca menerima *Field Input Set Enable*, maka pengguna unit BMG dapat memasukan data cuaca terbaru. Pengguna unit BMG memasukan rangkaian Informasi Cuaca Terbaru ke dalam *field* rekam informasi yang tersedia. Rangkaian informasi ini akan menjadi calon Data Cuaca Valid Terkini yang akan disimpan dalam *database*.

27. P-SPEC 3.4.3

Masukan : Data Cuaca Valid Terkini

Data Cuaca

Keluaran : Data Cuaca Valid

Konfirmasi Olah Data Cuaca

Record Data Cuaca Active 1

Record Cuaca

Algoritma :

Proses validasi data menerima data cuaca valid terkini yang merupakan informasi yang siap disimpan dalam *database*. Proses selanjutnya akan memeriksa, apakah

ada informasi yang salah atau terlewat. Jika informasi tidak sesuai, akan muncul pesan Konfirmasi Olah Data Cuaca yang memberitahu letak kesalahannya. Jika informasi sesuai, akan muncul pesan Konfirmasi Olah Data Cuaca yang bertanya pada pengguna unit BMG apakah akan menyimpan data atau tidak. Jika pengguna unit BMG memutuskan untuk menyimpan data, maka data akan tersimpan dalam *database*. Data yang tersimpan dalam *database* merupakan Data Cuaca Valid. Jika tidak, maka data tidak akan tersimpan. Proses juga secara otomatis mencari Data Cuaca Tersimpan dan mengirim parameter *Record Data Cuaca Active 1*, yaitu informasi cuaca yang baru saja dimasukkan untuk siap ditampilkan. Juga mengaktifkan parameter *Record Cuaca* yang seluruhnya tersimpan dalam *database* yang menjadi parameter untuk pembuatan laporan data cuaca.

28. P-SPEC 3.5.1

Masukan : *Field Form Set Data METAR*

Field Form Set Voice Recording

Keluaran : *Field Input Set Enable*

Nama *User* Pengguna Unit BO Aktif

Voice Record Ready State

Algoritma :

Proses olah data METAR menerima *Field Form Set Data METAR*. Jika pengguna unit BO akan memasukan data METAR terbaru, maka *Field Input Set Enable*, artinya pengguna unit BMG dapat menambah data cuaca baru. Jika pengguna unit BO akan melakukan perekaman suara, maka *Field Form Set Voice Recording*, artinya pengguna unit BO dapat melakukan perekaman suara. Parameter *Voice Record Ready State* menandakan sesi perekaman sudah dapat dilakukan. Nama *User Pengguna Unit BO Aktif* akan menjadi parameter selama pengguna unit BO ini menggunakan aplikasi.

29. P-SPEC 3.5.2

Masukan : *Field Input Set Enable*

Data Cuaca NEW

Informasi METAR Terbaru

Keluaran : Data METAR Valid Terkini

Record Data Cuaca Data NEW

Algoritma :

Proses input data METAR menerima *Field Input Set Enable*, maka pengguna unit BO dapat memasukan data METAR terbaru. Proses selanjutnya akan mencari Data Cuaca NEW dari dalam database dan otomatis akan

membawa parameter *Record* Data Cuaca Data NEW sebagai parameter yang siap ditampilkan. Pengguna unit BO memasukan rangkaian Informasi METAR Terbaru ke dalam *field* rekam informasi yang tersedia. Rangkaian informasi ini akan menjadi calon Data METAR Valid Terkini yang akan disimpan dalam *database*.

30. P-SPEC 3.5.3

Masukan : Data METAR Valid Terkini

Data METAR

Keluaran : Konfirmasi Olah Data METAR

Data METAR Valid

Voice Engine Ready State

Record METAR

Algoritma :

Proses validasi data menerima data METAR valid terkini yang merupakan informasi yang siap disimpan dalam *database*. Proses selanjutnya akan memeriksa, apakah ada informasi yang salah atau terlewat. Jika informasi tidak sesuai, akan muncul pesan Konfirmasi Olah Data METAR yang memberitahu letak kesalahannya. Jika informasi sesuai, akan muncul pesan Konfirmasi Olah Data METAR yang bertanya pada pengguna unit BO apakah akan menyimpan data atau tidak. Jika pengguna

unit BO memutuskan untuk menyimpan data, maka data akan tersimpan dalam *database*. Data yang tersimpan dalam database merupakan Data METAR Valid. Proses selanjutnya akan mencari Data METAR terbaru dan mengaktifkan parameter *Voice Engine Ready State* untuk sesi aktifasi suara. Jika tidak, maka data tidak akan tersimpan. Juga mengaktifkan parameter *Record METAR* yang seluruhnya tersimpan dalam *database* yang menjadi parameter untuk pembuatan laporan data METAR.

31. P-SPEC 3.6.1

Masukan : *Field Form Set Data Global Bandara*

Field Form Set Data Kepala Unit

Field Form Set Data Hak Akses

Keluaran : *Field Input Global Enable*

Field Olah Kepala Unit Enable

Field Olah Hak Akses Enable

Nama User *Supervisor* Aktif

Algoritma :

Proses menampilkan *field form* pengelolaan informasi menerima *Field Form Set Data Global Bandara*, Jika *supervisor* akan memasukan data global bandara, maka *Field Input Global Enable*, artinya *supervisor* dapat menambah data global bandara. *Field Form Set Data*

Kepala Unit, Jika *supervisor* akan memasukan data kepala unit baru, maka *Field* Olah Kepala Unit Enable, artinya *supervisor* dapat menambah data kepala unit. *Field* Form Set Data Hak Akses, Jika *supervisor* akan memasukan data hak akses *user*, maka *Field* Olah Hak Akses Enable, artinya *supervisor* dapat menambah data hak akses *user*. Nama *User supervisor* Aktif akan menjadi parameter selama *supervisor* ini menggunakan aplikasi.

32. P-SPEC 3.6.2

Masukan : *Field Input Global Enable*

Informasi Global Bandara

Data Global Bandara

Keluaran : Data Valid Terkini

Algoritma :

Proses input data global bandara menerima *Field Input Global Enable*, maka *supervisor* dapat memasukan data global bandara. *supervisor* memasukan rangkaian Informasi Global Bandara ke dalam *field* rekam informasi yang tersedia. Rangkaian informasi ini akan menjadi calon Data Valid Terkini yang akan disimpan dalam *database*. Jika sebelumnya sudah ada data global

bandara, maka proses secara otomatis akan mencari Data Global Bandara yang tersimpan.

33. P-SPEC 3.6.3

Masukan : *Field Olah Kepala Unit Enable*

Keluaran : *Field Input Kepala Unit Enable*

Field Update Kepala Unit Enable

Algoritma :

Proses olah data kepala unit menerima *Field Olah Kepala Unit Enable*. Jika *supervisor* akan memasukan data terbaru kepala unit, maka *Field Input Kepala Unit Enable*, artinya *supervisor* dapat menambah data kepala unit. Jika *supervisor* ingin mengedit informasi kepala unit, maka *Field Update Kepala Unit Enable*, artinya *supervisor* dapat merubah data kepala unit yang tersimpan.

34. P-SPEC 3.6.4

Masukan : *Field Olah Hak Akses Enable*

Keluaran : *Field Input Hak Akses Enable*

Field Update Hak Akses Enable

Algoritma :

Proses olah data hak akses menerima *Field Olah Hak Akses Enable*. Jika *supervisor* akan memasukan data terbaru hak akses, maka *Field Input Hak Akses Enable*,

artinya *supervisor* dapat menambah data hak akses. Jika *supervisor* ingin mengedit informasi hak akses, maka *Field Update Hak Akses Enable*, artinya *supervisor* dapat merubah data hak akses yang tersimpan.

35. P-SPEC 3.6.5

Masukan : *Field Input Kepala Unit Enable*

Informasi Kepala Unit Terbaru

Keluaran : Data Valid Terkini

Algoritma :

Proses input data kepala unit menerima *Field Input Kepala Unit Enable*, *supervisor* dapat memasukan data kepala unit terbaru. *supervisor* memasukan rangkaian Informasi Informasi Kepala Unit Terbaru ke dalam *field* rekam informasi yang tersedia. Rangkaian informasi ini akan menjadi calon Data Valid Terkini yang akan disimpan dalam *database*.

36. P-SPEC 3.6.6

Masukan : *User ID Kepala Unit*

Keluaran : Data Valid Terkini

Algoritma :

Proses edit data kepala unit menerima *User ID* kepala unit sebagai parameter kepala unit yang hendak diedit. Proses selanjutnya akan mengelola rangkaian informasi

terbaru ini yang akan menjadi calon Data Valid Terkini yang akan disimpan dalam *database*.

37. P-SPEC 3.6.7

Masukan : *User ID* Kepala Unit

Keluaran : Data Valid Terkini

Algoritma :

Proses hapus data kepala unit menerima *User ID* kepala unit sebagai parameter kepala unit yang hendak dihapus. Proses selanjutnya akan menghapus rangkaian informasi ini dari dalam *database*.

38. P-SPEC 3.6.8

Masukan : *Field Update* Kepala Unit *Enable*

Data Kepala Unit

Keluaran : User ID Kepala Unit

Algoritma :

Proses mencari *user id* kepala unit menerima *Field Update* Kepala Unit *Enable* sebagai pemicu. Selanjutnya proses akan melakukan pencarian Data Kepala Unit yang tersimpan dalam *database*. Yang menjadi acuan dan parameter pencarian adalah *User ID* kepala unit yang diinginkan.

39. P-SPEC 3.6.9

Masukan : Data Valid Terkini

Record Data Qualify HEAD

Keluaran : Data Hak Akses Valid

Data Global Bandara Valid

Data Kepala Unit Valid

Pesan Olah Global Bandara

Konfirmasi Olah Data Kepala Unit

Record Informasi Global Bandara

Record Informasi Kepala Unit

Algoritma :

Proses validasi data menerima Data Valid Terkini, ini yang merupakan informasi yang siap disimpan dalam *database*. Proses selanjutnya akan memeriksa, apakah ada informasi yang salah atau terlewati. Saat *supervisor* mengelola data kepala unit, Jika informasi tidak sesuai, akan muncul pesan Konfirmasi Olah Data Kepala Unit yang memberitahu letak kesalahannya. Jika informasi sesuai, akan muncul pesan Konfirmasi Olah Data Kepala Unit yang bertanya pada *supervisor* apakah akan menyimpan data atau tidak. Jika *supervisor* memutuskan untuk menyimpan data, maka data akan tersimpan dalam *database*. Data yang tersimpan dalam *database*

merupakan Data Kepala Unit Valid. Proses selanjutnya secara otomatis mencari *Record* Data Qualify HEAD untuk menampilkan *Record* Informasi Kepala Unit sebagai parameter yang siap untuk ditampilkan. Saat *supervisor* mengelola data global bandara, Jika informasi tidak sesuai, akan muncul Pesan Olah Global Bandara yang memberitahu letak kesalahannya. Jika informasi sesuai, akan muncul pesan Pesan Olah Global Bandara yang yang memberitahu bahwa data telah tersimpan. Data yang tersimpan dalam *database* merupakan Data Global Bandara Valid. Proses selanjutnya secara otomatis menampilkan *Record* Informasi Global Bandara sebagai parameter yang siap untuk ditampilkan. Saat *supervisor* mengelola data hak akses,. Data yang tersimpan dalam *database* merupakan Data Hak Akses Valid.

40. P-SPEC 3.6.10

Masukan : *Field Update* Hak Akses *Enable*

Data Hak Akses

Level Akses

Keluaran : Data Valid Terkini

Record Informasi Hak Akses

Algoritma :

Proses mengoreksi hak akses *user* menerima *Field Update Hak Akses Enable* sebagai pemicu. Selanjutnya proses akan melakukan pencarian Data Hak Akses yang tersimpan dalam *database*. Yang menjadi acuan dan parameter pencarian adalah Level Akses yang diinginkan. Proses selanjutnya secara otomatis menampilkan *Record Informasi Hak Akses* sebagai parameter yang siap untuk ditampilkan. Proses selanjutnya akan mengelola rangkaian informasi terbaru ini yang akan menjadi calon Data Valid Terkini yang akan disimpan dalam *database*.

41. P-SPEC 3.7

Masukan : *Record* METAR

Tanggal / Periode METAR

Keluaran : Laporan Data METAR

Algoritma :

Proses membuat laporan data METAR menerima *Record* METAR sebagai parameter seluruh data METAR yang tersimpan dalam *database*. Pengguna unit BO cukup memasukan Tanggal / Periode METAR untuk mendapatkan informasi METAR yang diinginkan. Selanjutnya proses akan membuat informasi METAR ke

dalam bentuk Laporan Data METAR dalam bentuk microsoft excell.

42. P-SPEC 3.8

Masukan : *Voice Engine Ready State*

Keluaran : *Voice Engine ATIS*

Algoritma :

Proses aktivasi suara menerima *Voice Engine Ready State* sebagai parameter untuk sesi aktivasi suara. Selanjutnya proses akan menggunakan library suara untuk menghasilkan *Voice Engine ATIS* yang menginformasikan informasi cuaca terkini.

43. P-SPEC 3.9

Masukan : *Voice Record Ready State*

Suara

Keluaran : Rekaman Suara

Algoritma :

Proses merekam suara menerima *Voice Record Ready State* sebagai parameter untuk sesi perekaman suara. Pengguna unti BO cukup memasukan data suara dalam sesi ini. Selanjutnya proses akan menyimpan suara dan memainkan Rekaman Suara tersebut.

44. P-SPEC 3.10

Masukan : *Record* Cuaca

Tanggal / Periode Data Cuaca

Keluaran : Laporan Data Cuaca

Algoritma :

Proses membuat laporan data Cuaca menerima *Record* Cuaca sebagai parameter seluruh data Cuaca yang tersimpan dalam *database*. Pengguna unit BMG cukup memasukan Tanggal / Periode Cuaca untuk mendapatkan informasi Cuaca yang diinginkan. Selanjutnya proses akan membuat informasi Cuaca ke dalam bentuk Laporan Data Cuaca dalam bentuk microsoft excell.

45. P-SPEC 3.11

Masukan : Nama *User* Kepala Unit BMG Aktif

Nama *User* Kepala Unit BO Aktif

Nama *User* Pengguna Unit BMG Aktif

Nama *User* Pengguna Unit BO Aktif

Nama *User* Supervisor Aktif

Keluaran : Nama *User* Status *Online*

Algoritma :

Proses menutup menu menerima parameter Nama *User* Kepala Unit BMG Aktif, untuk kepala unit BMG yang sedang menggunakan aplikasi. Parameter Nama *User*

Kepala Unit BO Aktif, untuk kepala unit BO yang sedang menggunakan aplikasi. Parameter Nama *User* Pengguna Unit BMG Aktif, untuk pengguna unit BMG yang sedang menggunakan aplikasi. Parameter Nama *User* Pengguna Unit BO Aktif, untuk pengguna unit BO yang sedang menggunakan aplikasi. Parameter Nama *User* Supervisor Aktif, untuk *supervisor* yang sedang menggunakan aplikasi. Selanjutnya proses akan menandai semua masukan proses sebagai parameter Nama *User* Status *Online*, yaitu semua *user* yang sedang menggunakan aplikasi.

46. P-SPEC 4

Masukan : Data METAR Tersimpan

Data Cuaca Tersimpan

Record Informasi Global Bandara

Record Informasi Kepala Unit

Record Data Cuaca *Active* 1

Record Informasi Pengguna Unit BMG

Record Informasi Pengguna Unit BO

Record Data Cuaca Data NEW

Nama *User* Pengguna Unit BMG *Online*

Nama *User* Pengguna Unit BO *Online*

Form Olah Informasi

Keluaran : *Record* Seluruh Informasi Kepala Unit BMG

Record Seluruh Informasi Kepala Unit BO

Record Seluruh Informasi *Supervisor*

Record Seluruh Informasi Pengguna Unit BMG

Record Seluruh Informasi Pengguna Unit BO

Algoritma :

Untuk kepala unit BMG, proses display *record* data menerima parameter *Record* Informasi Pengguna Unit BMG, Nama *User* Pengguna Unit BMG *Online*, Data Cuaca Tersimpan. Kumpulan parameter tersebut menjadi parameter baru yaitu *Record* Seluruh Informasi Kepala Unit BMG, yaitu informasi yang hanya bisa ditampilkan dalam Form Olah Informasi kepala unit BMG. Untuk kepala unit BO proses display *record* data menerima parameter *Record* Informasi Pengguna Unit BO, Nama *User* Pengguna Unit BO *Online*, Data METAR Tersimpan. Kumpulan parameter tersebut menjadi parameter baru yaitu *Record* Seluruh Informasi Kepala Unit BO, yaitu informasi yang hanya bisa ditampilkan dalam Form Olah Informasi kepala unit BO. Untuk pengguna unit BMG proses display *record* data menerima parameter *Record* Data Cuaca *Active* 1, Nama *User* Pengguna Unit BO *Online*, Data Cuaca

Tersimpan. Kumpulan parameter tersebut menjadi parameter baru yaitu *Record* Seluruh Informasi Pengguna Unit BMG, yaitu informasi yang hanya bisa ditampilkan dalam Form Olah Informasi pengguna unit BMG. Untuk pengguna unit BO proses display *record* data menerima parameter *Record* Data Cuaca Data NEW, Nama *User* Pengguna Unit BMG *Online*, Data Cuaca Tersimpan, Data METAR Tersimpan Kumpulan parameter tersebut menjadi parameter baru yaitu *Record* Seluruh Informasi Pengguna Unit BO, yaitu informasi yang hanya bisa ditampilkan dalam Form Olah Informasi pengguna unit BO. Untuk *supervisor* proses display *record* data menerima parameter *Record* Informasi Global Bandara, *Record* Informasi Kepala Unit. Kumpulan parameter tersebut menjadi parameter baru yaitu *Record* Seluruh Informasi *Supervisor*, yaitu informasi yang hanya bisa ditampilkan dalam Form Olah Informasi *supervisor*

47. P-SPEC 5

Masukan : *User Status Online*

Keluaran : Nama *User Pengguna Unit BMG Online*

Nama *User Pengguna Unit BO Online*

Algoritma :

Proses mencari nama *user* berstatus *online* menerima parameter *User Status Online*. Selanjutnya proses akan mencari Nama *User Pengguna Unit BMG Online* dan Nama *User Pengguna Unit BO Online* sebagai parameter baru yang siap ditampilkan.

48. P-SPEC 6

Masukan : *Record Seluruh Informasi Kepala Unit BMG*

Record Seluruh Informasi Kepala Unit BO

Record Seluruh Informasi Supervisor

Record Seluruh Informasi Pengguna Unit BMG

Record Seluruh Informasi Pengguna Unit BO

Keluaran : *Display Informasi Kepala Unit BMG*

Display Informasi Kepala Unit BO

Display Informasi Pengguna Unit BMG

Display Informasi Pengguna Unit BO

Display Informasi Supervisor

Algoritma :

Untuk kepala unit BMG, proses menampilkan *record* data menerima parameter *Record* Seluruh Informasi Kepala Unit BMG, selanjutnya proses akan menampilkan parameter tersebut berupa Display Informasi Kepala Unit BMG, yaitu seluruh kumpulan informasi yang tampil dalam menu kepala unit BMG. Untuk kepala unit BO, proses menampilkan *record* data menerima parameter *Record* Seluruh Informasi Kepala Unit BO, selanjutnya proses akan menampilkan parameter tersebut berupa Display Informasi Kepala Unit BO, yaitu seluruh kumpulan informasi yang tampil dalam menu kepala unit BO. Untuk pengguna unit BMG, proses menampilkan *record* data menerima parameter *Record* Seluruh Informasi Pengguna Unit BMG, selanjutnya proses akan menampilkan parameter tersebut berupa Display Informasi Pengguna Unit BMG, yaitu seluruh kumpulan informasi yang tampil dalam menu pengguna unit BMG. Untuk pengguna unit BO, proses menampilkan *record* data menerima parameter *Record* Seluruh Informasi Pengguna Unit BO, selanjutnya proses akan menampilkan parameter tersebut berupa Display Informasi Pengguna Unit BO,

yaitu seluruh kumpulan informasi yang tampil dalam menu pengguna unit BO. Untuk *supervisor*, proses menampilkan *record* data menerima parameter *Record* Seluruh Informasi *Supervisor*, selanjutnya proses akan menampilkan parameter tersebut berupa Display Informasi *Supervisor*, yaitu seluruh kumpulan informasi yang tampil dalam menu *supervisor*.

49. P-SPEC 7

Masukan : Pesan Komunikasi

Keluaran : Pesan Komunikasi Pengguna Unit BO

Pesan Komunikasi Pengguna Unit BMG

Algoritma :

Proses mengirim pesan komunikasi menerima Pesan Komunikasi dalam sesi komunikasi. Yaitu pesan yang dikirimkan pengguna unit BMG dan pengguna unit BO. Pengguna unit BO akan menerima Pesan Komunikasi Pengguna Unit BMG, pengguna unit BMG akan menerima Pesan Komunikasi Pengguna Unit BO.

50. P-SPEC 8

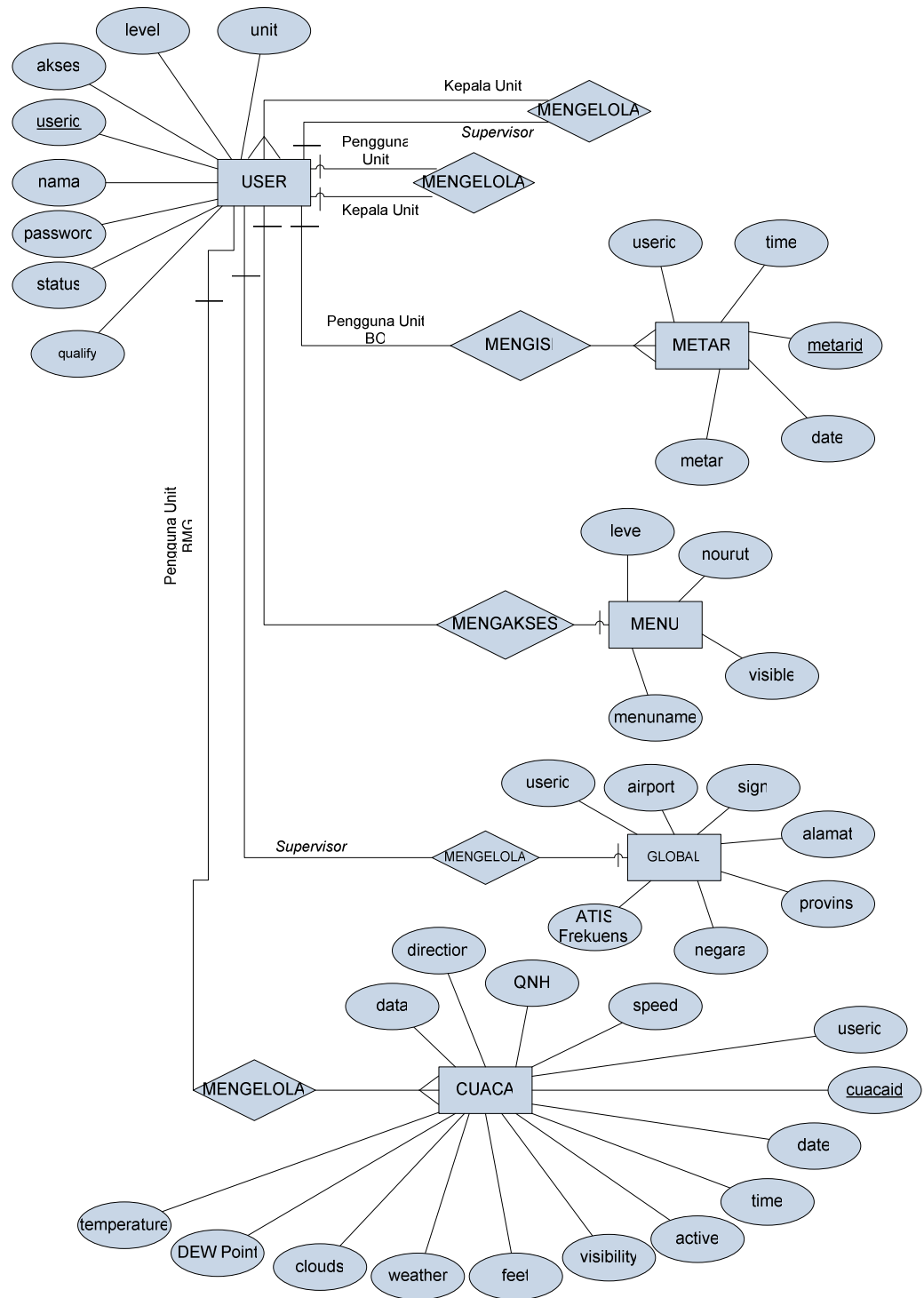
Masukan : Nama *User Status Online*

Keluaran : Informasi Status *Offline*

Algoritma :

Proses keluar dari aplikasi menerima Nama *User Status Online*, yaitu nama *user* siapapun yang sedang menggunakan aplikasi. Ketika *user* tersebut memutuskan untuk menyudahi sesi penggunaan aplikasi, maka proses selanjutnya akan memperbaharui atribut status dari *user* tersebut menjadi Informasi Status *Offline*, yang artinya *user* yang bersangkutan sudah tidak menggunakan aplikasi lagi.

d. Entity Relationship Diagram (ERD)

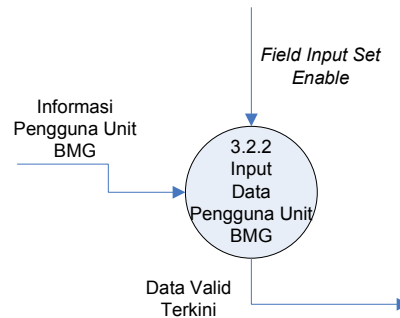


Gambar 4.11 Entity Relationship Diagram (ERD) P-ATIS

e. Kamus Data

1. Kamus Data Kepala Unit BMG

A. Kamus Data Proses 3.2.2



Gambar 4.12 Proses Input Data Pengguna BMG

Kamus Data

Field Input Set Enable = * Field form input data menjadi aktif *

Informasi Pengguna Unit BMG = *user id* + nama + *password* + unit
 + akses + *level* + status +
qualify

user id = * Terdiri dari deretan huruf atau angka atau kombinasi
 huruf dan angka *

nama = (gelar) + nama lengkap + (gelar)

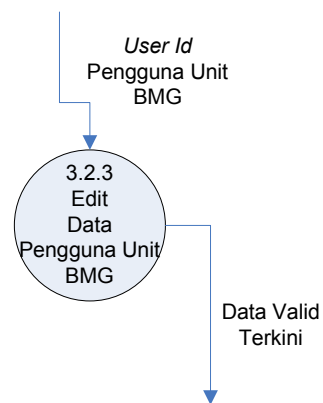
password = * Terdiri dari deretan huruf atau angka atau kombinasi
 huruf dan angka *

unit = * Unit BMG *

akses = [*Activated* | *Deactivated*]

- level* = * Terdiri dari satu digit bilangan *
 * Angka lima untuk aktivasi *
- status* = * Otomatis terisi “Offline” *
 * Jika Pengguna Unit BMG tersebut memakai aplikasi
 Maka otomatis terisi “Online” *
 * Akan otomatis kembali berubah menjadi “Offline”
 ketika Keluar dari aplikasi *
- qualify* = * Otomatis terisi “USER” *
- Data Valid Terkini = * data pengguna unit BMG yang telah
 dimasukan *

B. Kamus Data Proses 3.2.3



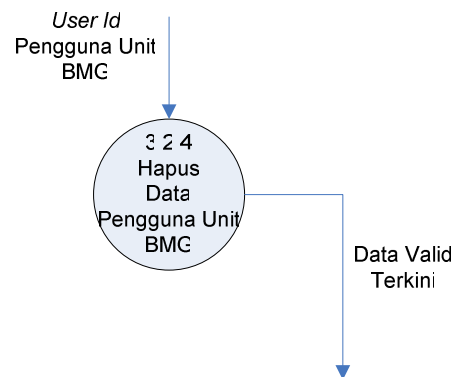
Gambar 4.13 Proses Edit Data Pengguna BMG

Kamus Data

User Id Pengguna Unit BMG = * *user id* dari Pengguna Unit BMG
 yang ingin *diedit* *

Data Valid Terkini = * data Pengguna Unit Telah diperbaharui *

C. Kamus Data Proses 3.2.4



Gambar 4.14 Proses Hapus Data Pengguna BMG

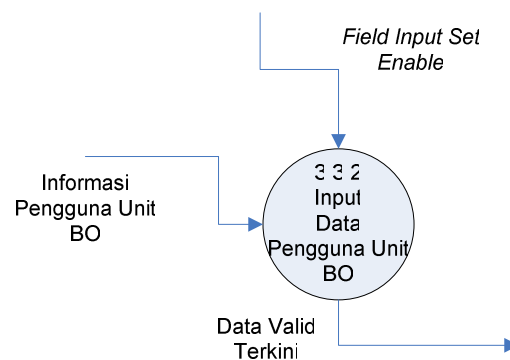
Kamus Data

User Id Pengguna Unit BMG = * *user id* dari Pengguna Unit BMG yang ingin dihapus *

Data Valid Terkini = * Penghapusan data pengguna unit BMG dari *database* *

2. Kamus Data Kepala Unit BO

A. Kamus Data Proses 3.3.2



Gambar 4.15 Proses Input Data Pengguna Unit BO

Kamus Data

Field Input Set Enable = * Field form input data menjadi aktif *

Informasi Pengguna Unit BO = user id + nama + *password* + unit +
akses + *level* + status + *qualify*

user id = * Terdiri dari deretan huruf atau angka atau kombinasi
huruf dan angka *

nama = (gelar) + nama lengkap + (gelar)

password = * Terdiri dari deretan huruf atau angka atau kombinasi
huruf dan angka *

unit = * Unit BO *

akses = [Activated | Deactivated]

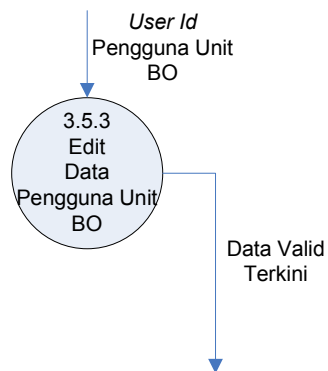
level = * Terdiri dari satu digit bilangan *
* Angka enam untuk aktifasi *

status = * Otomatis terisi “Offline” *
* Jika Pengguna Unit BO tersebut memakai aplikasi
Maka otomatis terisi “Online” *
* Akan otomatis kembali berubah menjadi “Offline”
ketika Keluar dari aplikasi *

qualify = * Otomatis terisi “USER” *

Data Valid Terkini = * data pengguna unit BO yang telah
dimasukan *

B. Kamus Data Proses 3.5.3



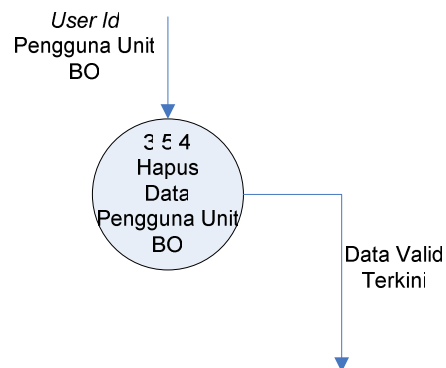
Gambar 4.16 Proses Edit Data Pengguna BO

Kamus Data

User Id Pengguna Unit BO = * *user id* dari Pengguna Unit BO yang ingin *diedit* *

Data Valid Terkini = * data Pengguna Unit BO yang telah diperbaharui *

C. Kamus Data Proses 3.5.4



Gambar 4.17 Proses Hapus Data Pengguna BO

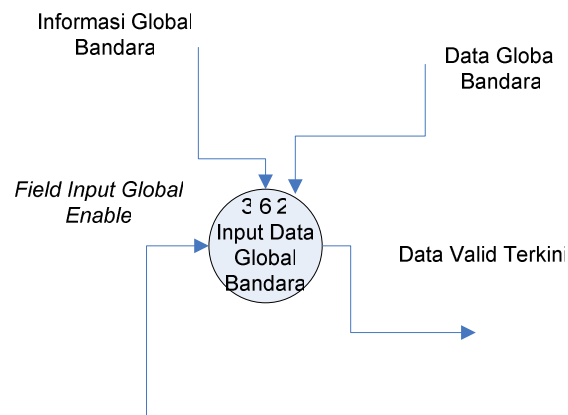
Kamus Data

User Id Pengguna Unit BO = * *user id* dari Pengguna Unit BO yang ingin dihapus *

Data Valid Terkini = * Penghapusan data pengguna unit BO dari *database* *

3. Kamus Data *Supervisor*

A. Kamus Data Proses 3.6.2



Gambar 4.18 Proses Input Data Global *Supervisor*

Kamus Data

Field Input Global Enable = * Field form input data global menjadi aktif *

Informasi Global Bandara = *airport* + *sign* + negara + propinsi + alamat + ATIS Frekuensi + User

airport = * terdiri dari deretan huruf *

sign = * terdiri dari deretan huruf *

negara = * terdiri dari deretan huruf *

propinsi = * terdiri dari deretan huruf *

alamat = * terdiri dari deretan huruf atau angka atau kombinasi huruf , angka, dan tanda baca koma dan titik *

ATIS frekuensi = * terdiri dari deretan angka serta tanda baca koma dan titik *

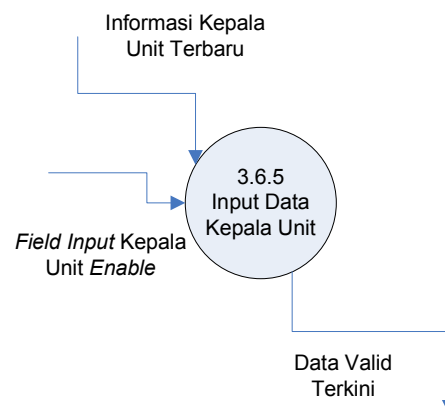
User = * terdiri dari deretan huruf *

* Nama Dari Supervisor yang mengelola kepala unit *

Data Global Bandara = * Data Global Bandara yang tersimpan dalam *database* *

Data Valid Terkini = * Kumpulan *record* informasi global bandara terkini yang dapat disimpan dalam *database* *

B. Kamus Data Proses 3.6.5



Gambar 4.19 Proses Input Data Kepala Unit *Supervisor*

Kamus Data

Field Input Kepala Unit Enable = * Field form input data kepala unit
menjadi aktif *

Informasi Kepala Unit = *user id* + nama + *password* + unit + akses +
level + status + *qualify*

user id = * terdiri dari deretan huruf atau angka atau kombinasi
huruf dan angka *

nama = (gelar) + nama lengkap + (gelar)

password = * terdiri dari deretan huruf atau angka atau kombinasi
huruf dan angka *

unit = [BO | BMG]

akses = [*Activated* | *Deactivated*]

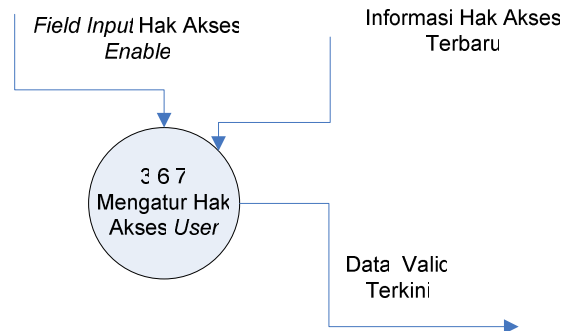
level = * terdiri dari satu digit bilangan *
* angka tiga untuk kepala unit BMG *
* angka empat untuk kepala unit BO
[3 | 4]

status = * Otomatis terisi “Offline” *
* Jika Kepala Unit tersebut memakai aplikasi
Maka otomatis terisi “Online” *
* Akan otomatis kembali berubah menjadi “Offline”
ketika Keluar dari aplikasi *

qualify = * Otomatis terisi “HEAD” *

Data Valid Terkini = * Kumpulan *record* informasi kepala unit terkini yang dapat disimpan dalam *database* *

C. Kamus Data Proses 3.6.7



Gambar 4.20 Proses Input Data Hak Akses *Supervisor*

Kamus Data

Field Input Hak Akses Enable = * Field form input hak akses menjadi aktif *

Informasi Hak Akses Terbaru = *level* + *menu*

level = * terdiri dari deretan angka *

menu = *File* + *Master Data* + (*Supervisor*) + (*Report*) + (*Record*) + *Help*

File = (*Global Setup*) + (*Set BMG User*) + (*Set BO User*) + (*Set Weather*) + (*Set ATIS*)

Master Data = (*User Data*) + (*Weather Data*) + (*METAR Info*)

(*Supervisor*) = (*Set User*) + (*Set User Menu*)

(Report) = *(Daily)* + *(Periodic)*

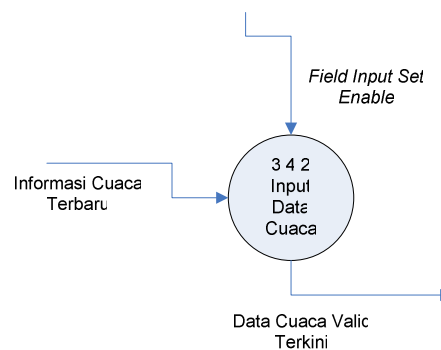
(Record) = * Format Suara *

Help = *(P-ATIS Info)* + *(P-ATIS Guide)* +
(Acces Denied)

Data Valid Terkini = * Kumpulan *record* informasi hak akses terkini
yang dapat disimpan dalam *database* *

4. Kamus Data Pengguna Unit BMG

A. Kamus Data Proses 3.4.2



Gambar 4.21 Proses Olah Data Cuaca Pengguna Unit BMG

Kamus Data

Field Input Set Enable = * Field form input data menjadi aktif *

Informasi Cuaca Terbaru = *date* + *time* + *direction* + *speed* +
visibility + *weather* + *clouds* + *feet* +
temperature + *DEW point* + *QNH* +
data + *active* + *user id*

date = * Tanggal informasi cuaca terbaru *

time = * Waktu informasi cuaca terbaru *

<i>direction</i>	= * Terdiri dari deretan angka tiga digit *
<i>speed</i>	= * Terdiri dari deretan angka dua digit *
<i>visibility</i>	= * Terdiri dari deretan angka lima digit *
<i>weather</i>	= [NIL HAZE RAIN FOG]
<i>clouds</i>	= [BKN SCT FEW SKC OVC]
<i>feet</i>	= * Terdiri dari deretan angka enam digit *
<i>temperature</i>	= * Terdiri dari deretan angka dua digit *
DEW <i>point</i>	= * Terdiri dari deretan angka dua digit *
QNH	= * Terdiri dari deretan angka empat digit *
<i>data</i>	= * Status data cuaca terbaru yang dimasukkan * [NEW OLD] * Saat menyimpan data, otomatis menjadi NEW * * Saat terjadi penambahan data baru, status data sebelumnya (NEW) otomatis berubah menjadi OLD *
<i>active</i>	= * Data cuaca yang dimasukkan pengguna unit BMG saat itu * [1 0] * Data cuaca yang dimasukkan bernilai satu (1) selama pengguna unit BMG bertugas * * Data bernilai satu ini yang ditampilkan oleh grid, selama pengguna unit BMG bertugas *

* Data bernilai nol (0) ketika pengguna unit
BMG menghentikan tugasnya *

* Data bernilai nol tidak akan tampil di grid *

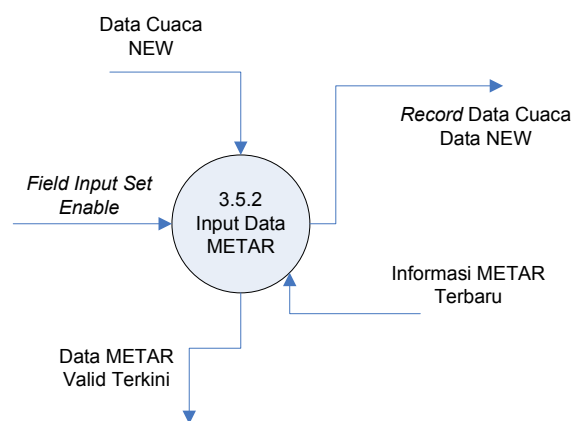
user id = * Pengguna Unit BMG yang bertugas saat itu *

* Otomatis terisi *

Data Valid Terkini = * Kumpulan *record* informasi cuaca terkini
yang dapat disimpan dalam *database* *

5. Kamus Data Pengguna Unit BO

A. Kamus Data Proses 3.5.2



Gambar 4.22 Proses Simpan Informasi METAR Pengguna Unit BO

Kamus Data

Field Input Set Enable = * Field form input data menjadi aktif *

Data Cuaca NEW = * data cuaca yang memiliki nilai atribut data

“NEW”*

Record Data Cuaca Data NEW = * *Record* data cuaca terbaru yang
 dimasukan pengguna unit BMG
 *

Informasi METAR Terbaru = *date* + *time* + METAR + *user*

Date = * Terdiri dari kombinasi angka *
 * Merupakan kombinasi tanggal, bulan, tahun *

Time = * Terdiri dari kombinasi angka *
 * Merupakan kombinasi jam dan menit *

METAR = [Alfa | Bravo | Charlie | Delta | Echo | Foxtrot |
 Golf | Hotel | India | Juliete | Kilo | Lima | Mike
 November | Oscar | Papa | Quebec | Romeo |
 Sierra | Tango | Uniform | Victor | Whiskey | X-
 ray | Yankee | Zulu]

User = * Pengguna BO yang sedang menggunakan aplikasi *

Data METAR Valid Terkini = * Kumpulan *record* informasi
 METAR terkini yang dapat
 disimpan dalam *database* *

3. Tahap Design

a. Desain Database

1. Desain Tabel

Tabel 4.3 Tabel USER

Field	Tipe	Null	Key	Ekstra
userid	Char (10)		PRI	
nama	Char (40)			
password	Char (10)			
unit	Char (20)			
akses	Char (20)			
level	Int (3)			
status	Char (10)			
qualify	Char (10)			

Tabel 4.4 Tabel GLOBAL

Field	Tipe	Null	Key	Ekstra
airport	Char (40)		PRI	
sign	Char (5)			
negara	Char (20)			
provinsi	Char (20)			
alamat	Char (40)			
frekuensi	Char (10)			
userid	Char (10)			

Tabel 4.5 Tabel MENU

Field	Tipe	Null	Key	Ekstra
level	Int (3)			
nourut	Int (3)			
menuname	Char (30)			
visible	Int (3)			

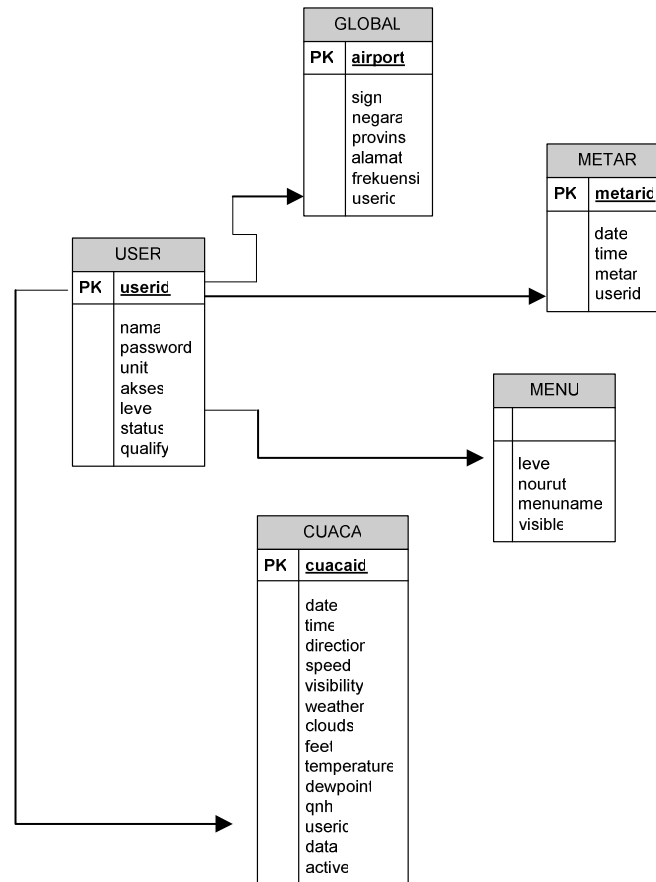
Tabel 4.6 Tabel METAR

Field	Type	Null	Key	Ekstra
metarid	Int (10)		PRI	Auto Increment
date	Date			
time	Time			
metar	Char (10)			
userid	Char (10)			

Tabel 4.7 Tabel CUACA

Field	Type	Null	Key	Ekstra
cuacaid	Int (10)		PRI	Auto Increment
date	Date			
time	Time			
direction	Int (5)			
speed	Int (3)			
visibility	Int (6)			
weather	Char (6)			
clouds	Char (5)			
feet	Int (7)			
temperature	Int (3)			
dewpoint	Int (3)			
qnh	Int (5)			
userid	Char (10)			
data	Char (5)			
active	Char (2)			

2. Database Relationship



Ket : PK = Primary Key

Gambar 4.23 Database Relationship P-ATIS

B. Desain User Interface

Desain ini dirancang sebagai gambaran awal bentuk-bentuk form yang digunakan oleh *user* untuk berinteraksi dengan perangkat lunak dalam melakukan aktifitasnya dalam mengelola informasi.

1. Supervisor

Userid	Nama	Unit	Akses	Status	Password
User Id :	<input type="text"/>				
Nama :	<input type="text"/>				
Password :	<input type="text"/>	<input type="text"/>			
Unit :	<input type="text"/>				
Akses :	<input type="text"/>				
<input type="button" value="Tambah"/> <input type="button" value="Koreksi"/> <input type="button" value="Simpan"/> <input type="button" value="Hapus"/> <input type="button" value="Batal"/> <input type="button" value="Keluar"/>					

Gambar 4.24 Desain *User Interface* Kelola Kepala Unit

2. Kepala Unit BMG

Userid	Nama	Unit	Akses	Status	Password
User Id :	<input type="text"/>				
Nama :	<input type="text"/>				
Password :	<input type="text"/>	<input type="text"/>			
Unit :	<input type="text" value="BMG"/>				
Akses :	<input type="text"/>				
<input type="button" value="Tambah"/> <input type="button" value="Koreksi"/> <input type="button" value="Simpan"/> <input type="button" value="Hapus"/> <input type="button" value="Batal"/> <input type="button" value="Keluar"/>					

Gambar 4.25 Desain *User Interface* Kelola Pengguna Unit BMG

3. Kepala Unit BO

Userid	Nama	Unit	Akses	Status	Password
UserId :	<input type="text"/>				
Nama :	<input type="text"/>				
Password :	<input type="text"/>	<input type="text"/>			
Unit :	<input type="text" value="BO"/>				
Akses :	<input type="text"/>				
<input type="button" value="Tambah"/> <input type="button" value="Koreksi"/> <input type="button" value="Simpan"/> <input type="button" value="Hapus"/> <input type="button" value="Batal"/> <input type="button" value="Keluar"/>					

Gambar 4.26 Desain *User Interface* Kelola Pengguna Unit BO

4. Pengguna Unit BMG

Date :	<input type="text"/>	Time :	<input type="text"/>
Direction :	<input type="text"/>		
Speed :	<input type="text"/>		
Visibility :	<input type="text"/>		
Weather :	<input type="text"/>		
Clouds :	<input type="text"/>	Feet :	<input type="text"/>
Temperature :	<input type="text"/>	DEW Point :	<input type="text"/>
ONH :	<input type="text"/>		
USER :	<input type="text"/>		
<input type="button" value="Tambah"/> <input type="button" value="Simpan"/> <input type="button" value="Batal"/>			

Gambar 4.27 Desain *User Interface* Kelola Data Cuaca

5. Pengguna Unit BO

Date : Time : METAR :

USER :

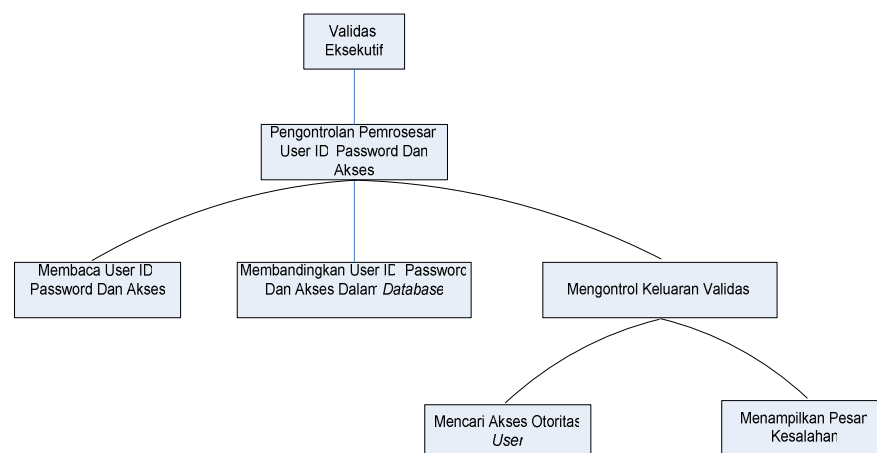
Gambar 4.28 Desain *User Interface* Kelola Data METAR

6. Aplikasi Komunikasi *Chatting*

Gambar 4.29 Desain *User Interface* Aplikasi *Chatting*

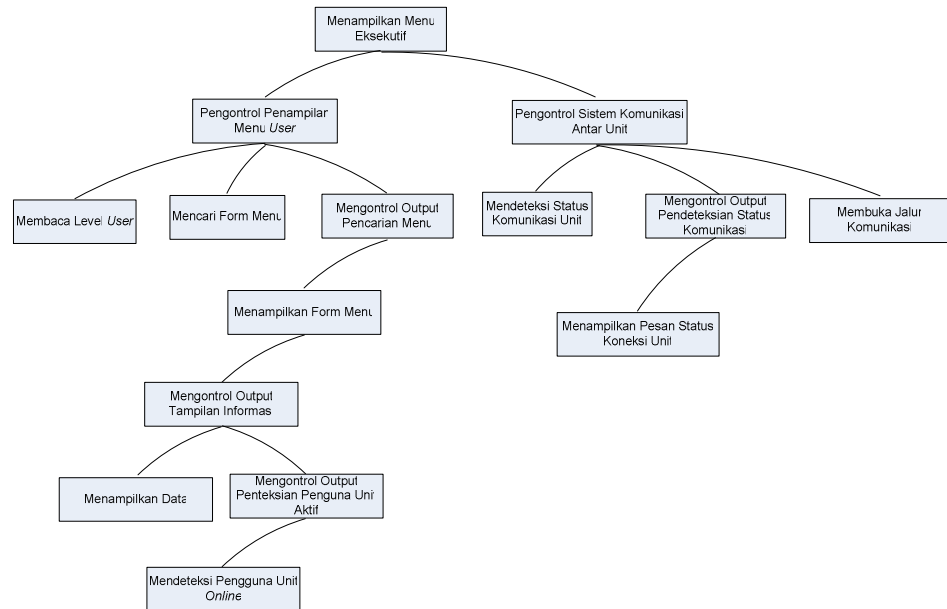
C. Desain Arsitektur Perangkat Lunak

1. Arsitektur Proses Validasi *User ID* Dan *Password*



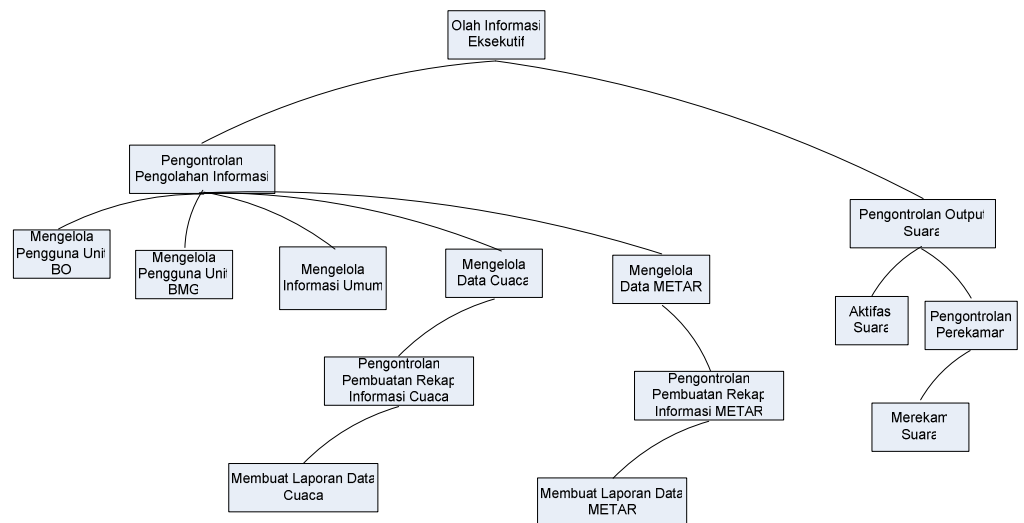
Gambar 4.30 Struktur Hirarki Validasi Eksekutif

2. Arsitektur Proses Menampilkan Menu



Gambar 4.31 Struktur Hirarki Menampilkan Menu Eksekutif

3. Arsitektur Proses Olah Informasi



Gambar 4.32 Struktur Hirarki Olah Informasi Eksekutif

4. Tahap Coding

a. Proses Otentifikasi

Proses ini merupakan yang pertama dilakukan oleh *user* yang ingin menggunakan aplikasi P-ATIS. *User* harus memasukan *user id* dan *password* yang telah diberikan, penggunaan huruf besar dan kecil harus diperhatikan. Dalam proses ini *case sensitive* berlaku. Aplikasi akan mencari ke dalam *database* apakah *user id* dan *password* yang dimasukan valid, selanjutnya aplikasi akan mengecek status otorisasi user, apakah *user* tersebut diaktifkan atau di non-aktifkan. Untuk dapat menggunakan aplikasi, status otorisasi *user* haruslah diaktifkan.

```
Dim dbUser As New ADODB.Recordset
Set dbUser = ExecSQL(GetDSN, "SELECT * FROM user WHERE userid='" & txtUserId.Text & "'
AND akses = 'Activated' ")
If dbUser.RecordCount > 0 Then
  If dbUser!Password = txtPassword.Text Then
    If dbUser!Akses = "Activated" Then
      If dbUser!UserID = txtUserId.Text Then
```

Gambar 4.33 Cuplikan Kode Program Proses Otentifikasi *User*

Cuplikan kode program diatas menjelaskan bahwa aplikasi mencari kecocokan *user id* dan *password* yang dimasukan dengan data yang terdapat dalam *database*, selanjutnya aplikasi akan mengecek status otorisasi dari *user* tersebut apakah diaktifkan atau di non-aktifkan.

1. Kode Program Kegagalan Otentifikasi

a. Kesalahan Memasukan User ID Atau Password

```
Else
  Pesan Array("Proses Otentifikasi Gagal, Periksa Kembali User ID atau Password anda!")
  txtPassword.SetFocus
End If
Else
  Pesan Array("Proses Otentifikasi Gagal, Periksa Kembali User ID atau Password anda!")
  txtUserId.SetFocus
End If
```

Gambar 4.34 Cuplikan Kode Program Kesalahan *User ID* Atau *Password*

Cuplikan kode program diatas merupakan realisasi dari proses 1.2. Cuplikan kode program ini menjelaskan pesan kesalahan yang mungkin terjadi. Kegagalan disebabkan karena *user id* atau *password* tidak cocok dengan data yang tersimpan dalam *database*, sehingga user tersebut tidak dapat menggunakan aplikasi P-ATIS.

b. Kesalahan Karena Hak Akses *User* Dideaktifasi

```
Else
  Pesan Array("Anda Telah Dideaktifasi, Harap Segera Menghubungi Kepala Unit Anda !")
  txtUserId.SetFocus
End If
```

Gambar 4.35 Cuplikan Kode Program *User* Di Non-Aktifkan

Cuplikan kode program diatas juga merupakan realisasi dari proses 1.2. Cuplikan kode program ini menjelaskan pesan kesalahan yang mungkin terjadi. kegagalan disebabkan status otorisasi *user* yang telah di non-aktifkan, sehingga walaupun *user id* dan *password user* tersebut cocok, *user* yang telah di non-aktifkan tersebut tidak akan bisa menggunakan aplikasi.

Jika aplikasi berhasil menemukan kecocokan *user id* dan *password* dalam *database* dan status otorisasi *user* tersebut tidak di non-aktifkan, maka *user* tersebut telah berhasil melewati proses otentifikasi.

```

If dbuser!Password = txtPassword.Text Then
  If dbuser!Akses = "Activated" Then
    If dbuser!UserID = txtUserId.Text Then
      Dim upUser As New ADODB.Recordset
      Set upUser = ExecSQL(GetDSN, "UPDATE user SET status='Online' WHERE userid='" & dbuser!UserID & "'")
    
```

Gambar 4.36 Cuplikan Kode Program *Update* Informasi Status

Cuplikan kode program diatas menjelaskan bahwa ketika *user* berhasil melewati proses otentifikasi maka *user* tersebut mempunyai hak untuk menggunakan aplikasi. Selama *user* tersebut menggunakan aplikasi, maka aplikasi akan memberikan suatu tanda yaitu “Online” dengan tujuan menandakan bahwa *user* yang bersangkutan sedang menggunakan aplikasi sekarang.

b. Proses Menampilkan Menu

1. Proses Penyaringan Hak Akses *User*

Karena setiap *user* memiliki form menu yang berbeda-beda, baik itu kepala unit BO, kepala unit BMG, pengguna unit BO dan pengguna unit BMG. Maka untuk itu perlu dibuat suatu penyaring untuk membatasi form apa saja yang dapat diakses oleh masing-masing *user*. Yang menjadi kunci dari penyaring itu adalah sebuah hak akses *user*.

```

Sub SetMenu(frm As Object, level As Long)
Dim rs As New ADODB.Recordset
Dim obj As Object

'level 1 = supervisor
If level = 1 Then Exit Sub

For Each obj In frm.Controls
If UCase(Mid(obj.Name, 1, 3)) = "MNU" Then
Set rs = ExecSQL(GetDSN, "SELECT * FROM menu WHERE level='" & level & "'
AND menuname='" & obj.Name & "'")
If rs.RecordCount > 0 Then
On Error Resume Next
If rs!Visible = 1 Then obj.Visible = True Else obj.Visible = False
End If
End If
Next
End Sub

```

Gambar 4.37 Cuplikan Kode Program Mencari Form Menu *User*

Cuplikan kode program tersebut merupakan realisasi dari proses 2.1 yaitu proses yang bertujuan melakukan penyaringan hak akses *user*. Masing-masing hak akses user diklasifikasikan dalam level-level tertentu

2. Proses Menampilkan Menu *User*

Setelah melewati tahap penyaringan hak akses, *user* akan dibawa ke dalam suatu menu *user*. Inilah ruang kerja masing-masing *user* untuk mengelola informasi. Menu akses dari tiap-tiap *user* diklasifikasikan melalui level tertentu. Jadi, kepala unit BMG, kepala unit BO, pengguna unit BMG, pengguna unit BO sudah diklasifikasikan memiliki level tertentu yang berbeda satu sama lain. Masing-masing level tersebut berisi form menu apa saja yang dapat diakses oleh tiap-tiap *user* tersebut.

```

MainForm.UserName = dbUser!nama
MainForm.UserID = dbUser!UserID
MainForm.UserLevel = dbUser!level
Unload Me
MainForm.Show
MainForm.StatusBar1.Panels(1).Text = "User Name : " & dbUser!nama
MainForm.StatusBar1.Panels(7).Text = Format(Date, "dddd, dd-mm-yyyy") + " " +
    Format(TIME, "hh:mm:ss ")
MainForm.StatusBar1.Panels(8).Text = "Unit : " & dbUser!Unit

```

Gambar 4.38 Cuplikan Kode Program Tampil Menu *User*

Cuplikan kode program diatas merupakan realisasi dari proses 2.2. Menjelaskan bahwa dari menu *user* yang ditampilkan akan terdapat informasi siapa *user* yang menggunakan, dari unit mana *user* tersebut. Menu yang ditampilkan merupakan form menu yang telah diklasifikasikan dalam berdasarkan level-level tertentu. Isi dari klasifikasi menu akses berdasarkan level-level tersebut merupakan kumpulan dari sub-menu utama dan sub-sub menu dari sub-menu utama.

c. Proses Keluar Dari Aplikasi

```

Dim dbuser As New ADODB.Recordset
Set dbuser = ExecSQL(GetDSN, "SELECT * FROM user WHERE nama = '" & MainForm.UserName & "'")
If dbuser.RecordCount > 0 Then
    If MainForm.StatusBar1.Panels(1).Text = "User Name : " & dbuser!nama Then
        If dbuser!Status = "Online" Then
            Dim upUser As New ADODB.Recordset
            Set upUser = ExecSQL(GetDSN, "UPDATE user SET status='offline' WHERE nama='" & dbuser!nama & "'")
        End If
    End If

```

Gambar 4.39 Cuplikan Kode Program Keluar Dari Aplikasi

Saat keluar dari aplikasi, maka secara otomatis aplikasi P-ATIS akan memberikan parameter “Offline” kepada *user* yang telah menutup aplikasi P-ATIS ini. Artinya user yang sedang bersangkutan sedang tidak menggunakan aplikasi saat itu.

d. Kode Program Masing-Masing User

1. Kode Program *Supervisor*

a. Kode Program Global Setup

```

vaField = Array("airport", "sign", "negara", "provinsi", _
               "alamat", "frekuensi")
vavalue = Array(txtAirport.Text, txtSign.Text, txtNegara.Text, _
               txtProvinsi.Text, txtAlamat.Text, txtFrekuensi.Text)

Set rs = ExecSQL(GetDSN, "DELETE FROM global")
Success = Insert(GetDSN, "global", vaField, vavalue)

```

Gambar 4.40 Cuplikan Kode Program Masukan Data Global Setup

Cuplikan kode program diatas merupakan realisasi dari proses 3.6.2. Digunakan untuk menyimpan record data global setup yang berisi informasi bandara secara umum ke dalam *database*.

```

If Not CheckData(txtAlamat.Text, "Isi Alamat") Then
    txtAlamat.SetFocus
    ValidSaving = False
End If
If Not CheckData(txtFrekuensi.Text, "Isi Nama Airport") Then
    txtFrekuensi.SetFocus
    ValidSaving = False
End If

```

Gambar 4.41 Cuplikan Kode Validasi Data Global Setup

Cuplikan kode program diatas digunakan untuk memeriksa apakah ada informasi yang akan dijadikan sebagai *record* global setup tidak sesuai atau ada yang terlewati.

```

Dim sTemplate As String
sTemplate = "!@#%&*()_+-.<?/\|[]{};'"
If Instr(1, sTemplate, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
If KeyAscii > 47 And KeyAscii < 58 Then
    KeyAscii = 0
End If

```

Gambar 4.42 Cuplikan Kode Program Pembatasan Karakter Global Setup

Informasi yang berkaitan dengan data global bandara ini juga dibatasi. Berdasarkan kamus data proses 3.6. Cuplikan kode program pada gambar 4.42 merupakan realisasi dari kamus data 3.6.2.

b. Kode Program Set Kepala Unit

```

vaField = Array("userid", "nama",
               "password", "unit", "akses", "level", "qualify")
vaValue = Array(txtUserID.Text, txtNama.Text,
               txtPassword.Text, cmbUnit.Text, cmbAkses.Text, txtLevel.Text, txtQualify.Text)

If ValidSaving Then
  If YesNo(Array("Simpan data ?", "User ID: " + txtUserID.Text)) Then
    If isUpdate Then
      Success = Update(GetDSN, "user", "userid= '" & txtUserID.Text & "'", vaField, vaValue)
    Else
      Success = Insert(GetDSN, "user", vaField, vaValue)
    End If
  End If

```

Gambar 4.43 Cuplikan Kode Program Simpan Dan Edit Data Kepala Unit

Cuplikan kode program diatas merupakan realisasi dari proses 3.6.5, yaitu simpan data kepala unit, dan proses 3.6.6, edit kepala unit.

```

set del = ExecSQL(GetDSN, "DELETE FROM user WHERE userid='" & txtUserID.Text & "'")
RefreshGrid
setStatus 5, Me, vaKode, vaEntry, vaGrid, vaUnit
isUpdate = False

```

Gambar 4.44 Cuplikan Kode Program Hapus Data Kepala Unit

Cuplikan kode program diatas merupakan realisasi proses 3.6.7 yaitu hapus data kepala unit.

```

If Not CheckData(txtLevel, "sorot Pilihan Unit Dan Pilihan Akses !") Then
  ValidSaving = False
  txtLevel.SetFocus
  Exit Function
End If

If txtPassword.Text <> txtCPassword.Text Then
  MsgBox "konfirmasi Password tidak sama!", , "Informasi"
  txtCPassword.SetFocus
  ValidSaving = False
  Exit Function
End If

```

Gambar 4.45 Cuplikan Kode Program Validasi Data Kepala Unit

Cuplikan kode program di atas digunakan untuk memeriksa apakah ada informasi yang akan dijadikan sebagai *record* kepala unit tidak sesuai atau ada yang terlewat.

c. Kode Program Set User Menu

```
Dim i As Long
For i = 0 To nMenu - 1
    If Treeview1.Nodes(i + 1).Checked Then
        vavalue = Array(txtLevel1.Text, i, menuname(i), "1")
    Else
        vavalue = Array(txtLevel1.Text, i, menuname(i), "0")
    End If
    Success = Insert(GetDSN, "menu", vaField, vavalue)
Next
```

Gambar 4.46 Cuplikan Kode Program Simpan Data Hak Akses

Cuplikan kode program di atas menjelaskan bahwa sub-menu yang telah dipilih akan disimpan ke dalam *database*.

```
Set rs = ExecSQL(GetDSN, "SELECT * FROM menu WHERE level = '" & txtLevel1.Text & "' ORDER BY nourut")
If rs.RecordCount > 0 Then
    For i = 1 To rs.RecordCount
        If rs!visible = "0" Then
            Treeview1.Nodes(i).Checked = False
        Else
            Treeview1.Nodes(i).Checked = True
        End If
        rs.MoveNext
    Next i
```

Gambar 4.47 Cuplikan Kode Program *Reload* Data Hak Akses

Cuplikan kode di atas menjelaskan bahwa, ketika *supervisor* mengetikkan angka *level user* dan menekan tombol "Reload", maka secara otomatis set *record* hak akses menu yang telah diklasifikasikan dalam *level user* tersebut akan tampil.

2. Kode Program Kepala Unit BMG

a. Kode Program Set User BMG

```
Dim sTemplate As String
sTemplate = "!@#%&*()_+<=>?/\|[]{}:;'"
If InStr(1, sTemplate, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
If KeyAscii > 47 And KeyAscii < 58 Then
    KeyAscii = 0
End If
```

Gambar 4.48 Cuplikan Kode Program Realisasi Kamus Data Proses 3.2.2

Cuplikan kode program diatas merupakan realisasi dari kamus data proses 3.2.2 untuk membatasi karakter yang dapat dijadikan *record* untuk data pengguna unit BMG.

```
If Not CheckData(txtLevel, "Sorot Pilihan Akses !") Then
    ValidSaving = False
    txtLevel.SetFocus
    Exit Function
End If
```

Gambar 4.49 Cuplikan Kode Validasi Data Pengguna Unit BMG

Cuplikan kode program diatas digunakan untuk memeriksa apakah ada informasi yang akan dijadikan sebagai *record* pengguna unit BMG tidak sesuai atau ada yang terlewati.

```
vaField = Array("userid", "nama", "password", "unit", "akses", "level", "qualify")
vavalue = Array(txtUserID.Text, txtNama.Text, txtPassword.Text, txtUnit.Text, cmbAkses.Text, txtLevel.Text, txtQualify.Text)

If ValidSaving Then
    If YesNo(Array("Simpan Data ?", "User ID: " + txtUserID.Text)) Then
        If isupdate Then
            success = update(GetDSN, "user", "userid= '" & txtUserID.Text & "'", vaField, vavalue)
        Else
            success = insert(GetDSN, "user", vaField, vavalue)
        End If
    End If
```

Gambar 4.50 Cuplikan Kode Simpan Dan Edit Data Pengguna Unit BMG

Cuplikan kode program diatas merupakan realisasi dari proses 3.2.2, yaitu simpan data pengguna unit BMG, dan proses 3.2.3, edit kepala unit.

```
Dim del As New ADODB.Recordset
If YesNo(Array("Hapus Data ?", "User ID: " + txtUserID.Text)) Then
    Set del = ExecSQL(GetDSN, "DELETE FROM user WHERE userid='" & txtUserID.Text & "'")
    RefreshGrid
    SetStatus 5, Me, vaKode, vaEntry, vaGrid, vaUnit
    isupdate = False
End If
```

Gambar 4.51 Cuplikan Kode Program Hapus Data Pengguna Unit BMG

Cuplikan kode program diatas merupakan realisasi proses 3.2.4 yaitu hapus data Pengguna Unit BMG.

3. Kode Program Kepala Unit BO

a. Kode Program Set User BO

```
Dim sTemplate As String
sTemplate = "@#%&*( )_+-=<>?/\|[]{}:;'""
If InStr(1, sTemplate, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
If KeyAscii > 47 And KeyAscii < 58 Then
    KeyAscii = 0
End If
```

Gambar 4.52 Cuplikan Kode Program Realisasi Kamus Data Proses 3.3.2

Cuplikan kode program pada gambar 4.52 merupakan realisasi dari kamus data proses 3.3.2 untuk membatasi karakter yang dapat dijadikan *record* untuk data pengguna unit BO.

```
If Not CheckData(txtNama.Text, "Isi Nama User") Then
    ValidSaving = False
    txtNama.SetFocus
    Exit Function
End If
```

Gambar 4.53 Cuplikan Kode Validasi Data Pengguna Unit BO

Cuplikan kode program diatas digunakan untuk memeriksa apakah ada informasi yang akan dijadikan sebagai *record* pengguna unit BO tidak sesuai atau ada yang terlewati.

```

vaField = Array("userid", "nama", "password", "unit", "akses", "level", "qualify")
vavalue = Array(txtUserID.Text, txtNama.Text, txtPassword.Text, txtUnit.Text, cmbAkses.Text, txtLevel.Text, txtQualify.Text)

If ValidSaving Then
    If YesNo(Array("Simpan Data ?", "User ID: " + txtUserID.Text)) Then
        If !isupdate Then
            success = Update(GetDSN, "user", "userid= '" & txtUserID.Text & "'", vaField, vavalue)
        Else
            success = Insert(GetDSN, "user", vaField, vavalue)
        End If
    End If

```

Gambar 4.54 Cuplikan Kode Simpan Dan Edit Data Pengguna Unit BO

Cuplikan kode program diatas merupakan realisasi dari proses 3.3.2, yaitu simpan data pengguna unit BMG, dan proses 3.3.3, edit kepala unit.

```

Dim del As New ADODB.Recordset
If YesNo(Array("Hapus Data ?", "User ID: " + txtuserID.Text)) Then
    Set del = ExecSQL(GetDSN, "DELETE FROM user WHERE userid='" & txtuserID.Text & "'")
    RefreshGrid
    SetStatus 5, Me, vakode, vaEntry, vagrid, vaunit
    !isupdate = False
End If

```

Gambar 4.55 Cuplikan Kode Program Hapus Data Pengguna Unit BO

Cuplikan kode program diatas merupakan realisasi proses 3.3.4 yaitu hapus data Pengguna Unit BO.

4. Kode Program Pengguna Unit BMG

a. Kode Program Set Data Cuaca

```

sTemplate = "!@#%&*^&()*_+-.=<>?/\|[]{};:'""
If Instr(1, sTemplate, Chr(keyAscii)) > 0 Then
    keyAscii = 0
End If
sChar = "abcdefghijklmnopqrstuvwxy"
If Instr(1, sChar, Chr(keyAscii)) > 0 Then
    keyAscii = 0
End If
sBig = "ABCDEFGHIJKLMNPOQRSTUVWXYZ"
If Instr(1, sBig, Chr(keyAscii)) > 0 Then
    keyAscii = 0
End If

```

Gambar 4.56 Cuplikan Kode Program Pembatasan Karakter Data Cuaca

Cuplikan kode program pada gambar 3.51 merupakan realisasi dari kamus data proses 3.7.1 untuk membatasi karakter yang dapat dijadikan *record* untuk data cuaca.

```

If Not CheckData(txtQnh, "Isi Field QNH") Then
    ValidSaving = False
    txtQnh.SetFocus
    Exit Function
End If

```

Gambar 4.57 Cuplikan Kode Validasi Data Cuaca

Cuplikan kode program diatas digunakan untuk memeriksa apakah ada informasi yang akan dijadikan sebagai *record* data cuaca tidak sesuai atau ada yang terlewati.

```

vaField = Array("date", "time", "_",
               "direction", "speed", "visibility", "weather", "clouds", "feet", "temperature", "dewpoint",
               "qnh", "userid", "data", "active")
vaValue = Array(txtDate.Text, txtTime.Text, _
               txtDirection.Text, txtSpeed.Text, txtVisibility.Text, cmbWeather.Text, cmbClouds.Text,
               txtFeet.Text, txtTemperature.Text, txtDew.Text, txtQnh.Text, txtKeterangan.Text, txtData.Text,
               txtActive.Text)
If ValidSaving Then
    If YesNo(Array("Simpan data ?", "date: " + txtDate.Text, "time: " + txtTime.Text, "direction: " +
                txtDirection.Text, "speed: " + txtSpeed.Text, "visibility: " + txtVisibility.Text,
                "weather: " + cmbWeather.Text, "clouds: " + cmbClouds.Text, "feet: " +
                txtFeet.Text, "temperature: " + txtTemperature.Text, "dew point: " +
                txtDew.Text, "QNH: " + txtQnh.Text)) Then
        Success = Insert(GetDSN, "cuaca", vaField, vaValue)
    End If
End If

```

Gambar 4.58 Cuplikan Kode Program Simpan Data Cuaca

Cuplikan kode program diatas merupakan realisasi dari proses 3.7.2, yaitu *input* data cuaca.

```

Dim ubAktif As New ADODB.Recordset
Set ubAktif = ExecSQL(GetDSN, "SELECT * FROM cuaca WHERE active = '1'")
If ubAktif.RecordCount > 0 Then
    If ubAktif!Active = "1" Then
        Dim upAktif As New ADODB.Recordset
        Set upAktif = ExecSQL(GetDSN, "UPDATE cuaca SET Active = '0' WHERE cuacaId='" & ubAktif!CuacaId & "'")
    End If
End If

```

Gambar 4.59 Cuplikan Kode Program Update Atribut Active Data Cuaca

Ketika pengguna unit BMG menyimpan sebuah *record* data cuaca, maka secara otomatis aplikasi akan mengisi atribut “Active” dalam tabel cuaca dengan angka satu. Angka satu inilah yang menjadi tanda bahwa *record* data cuaca tersebut adalah *record* terbaru. Ketika pengguna unit BMG memutuskan untuk memasukan *record* data cuaca selanjutnya, maka secara otomatis aplikasi akan *update* atribut “Active” *record* data yang sebelumnya dengan angka nol.

b. Kode Program *Chat*

```
winsock1.SendData txtNick.Text & ": " & txtSendText.Text
txtIncomingData.Text = txtIncomingData.Text + vbCrLf + txtNick.Text + ": " + txtSendText.Text
txtSendText.Text = ""
```

Gambar 4.60 Cuplikan Kode Program Kirim Pesan *Chat* Pengguna Unit BMG

Cuplikan kode program di atas digunakan oleh aplikasi untuk mengirim pesan komunikasi oleh pengguna unit BMG kepada pengguna unit BO.

```
Dim Data As String
winsock1.GetData Data 'gets the data
txtIncomingData.Text = txtIncomingData.Text + vbCrLf & Data
```

Gambar 4.61 Cuplikan Kode Program Terima Pesan *Chat* Pengguna Unit BMG

Cuplikan kode program di atas digunakan oleh aplikasi untuk menerima pesan komunikasi yang dikirimkan oleh pengguna unit BO kepada pengguna unit BMG.

c. Kode Program Pembuatan Laporan Harian Data Cuaca

```

Set rsGlobal = ExecSQL(GetDSN, "SELECT * FROM cuaca " & _
  "WHERE date >=" & dtAwal.Value & "' AND " & _
  "date <=" & dtAwal.Value & "'")

If rsGlobal.RecordCount > 0 Then
  OpenExcel True
  OpenWorkBook App.Path & "\\LaporanHarianDataCuaca.xls"

```

Gambar 4.62 Cuplikan Kode Program Membuat Laporan Harian Data Cuaca

Cuplikan kode program di atas digunakan untuk membuat laporan harian data cuaca. Pertama-tama aplikasi mencari *record* data cuaca sesuai tanggal yang diminta oleh pengguna unit BMG, setelah data ditemukan *record* tersebut akan diletakan pada *file* format laporan harian data cuaca yang sebelumnya telah dibuat dalam aplikasi microsoft excell. Dalam aplikasi ini *file* format laporan harian data cuaca diberi nama “LaporanHarianDataCuaca.xls”.

d. Kode Program Pembuatan Laporan Periodik Data Cuaca

```

Set rsGlobal = ExecSQL(GetDSN, "SELECT * FROM cuaca " & _
  "WHERE date >=" & dtAwal.Value & "' AND " & _
  "date <=" & dtAkhir.Value & "'")

If rsGlobal.RecordCount > 0 Then
  OpenExcel True
  OpenWorkBook App.Path & "\\LaporanPeriodikDataCuaca.xls"

```

Gambar 4.63 Cuplikan Kode Membuat Laporan Periodik Data Cuaca

Cuplikan kode program di atas digunakan untuk membuat laporan periodik data cuaca. Pertama-tama aplikasi mencari *record* data cuaca sesuai *range* tanggal yang diminta oleh pengguna unit BMG, setelah data ditemukan *record* tersebut akan diletakan pada *file* format laporan periodik data cuaca yang sebelumnya telah dibuat dalam aplikasi microsoft excell.

Dalam aplikasi ini file format laporan periodik data cuaca ini diberi nama “LaporanPeriodikDataCuaca.xls”.

5. Kode Program Pengguna Unit BO

a. Kode Program Set Data METAR

```
sTemplate = "1@#%$A&*()_+-.=<>?/\|[]{};:'""
If Instr(1, sTemplate, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
sChar = "abcdefghijklmnopqrstuvwyz"
If Instr(1, sChar, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
sBig = "ABCDEFGHIJKLMNPOQRSTUVWXYZ"
If Instr(1, sBig, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
```

Gambar 4.64 Cuplikan Kode Program Pembatasan Karakter Data METAR

Cuplikan kode program pada gambar di atas merupakan realisasi dari kamus data proses 3.5.2 untuk membatasi karakter yang dapat dijadikan *record* untuk data METAR.

```
If Not CheckData(txtMetar, "Sorot Pilihan METAR !") Then
    ValidSaving = False
    cmbMetar.SetFocus
    Exit Function
End If
```

Gambar 4.65 Cuplikan Kode Validasi Data Cuaca

Cuplikan kode program diatas digunakan untuk memeriksa apakah ada informasi yang akan dijadikan sebagai *record* data METAR tidak sesuai atau ada yang terlewat.

```
vaField = Array("date", "time", _
    "metar", "userid")
vaValue = Array(txtDate.Text, txtTime.Text, _
    cmbMetar.Text, txtKeterangan.Text)

If ValidSaving Then
    If YesNo(Array("Simpan Data ?", "Date: " + txtDate.Text, "Time: " + txtTime.Text, "METAR: " + cmbMetar.Text))
        Success = Insert(GetDSN, "metar", vaField, vaValue)
```

Gambar 4.66 Cuplikan Kode Program Simpan Data METAR

Cuplikan kode program diatas merupakan realisasi dari proses 3.5.2, yaitu *input* informasi METAR.

b. Kode Program *Chat*

```
winsock1.SendData txtNick.Text & ": " & txtSendText.Text
txtIncomingData.Text = txtIncomingData.Text + vbCrLf + txtNick.Text + ": " + txtSendText.Text
txtSendText.Text = ""
```

Gambar 4.67 Cuplikan Kode Program Kirim Pesan *Chat* Pengguna Unit BO

Cuplikan kode program di atas digunakan oleh aplikasi untuk mengirim pesan komunikasi oleh pengguna unit BO kepada pengguna unit BMG.

```
Dim Data As String
winsock1.GetData Data 'gets the data
txtIncomingData.Text = txtIncomingData.Text + vbCrLf & Data
```

Gambar 4.68 Cuplikan Kode Program Terima Pesan *Chat* Pengguna Unit BO

Cuplikan kode program di atas digunakan oleh aplikasi untuk menerima pesan komunikasi yang dikirimkan oleh pengguna unit BMG kepada pengguna unit BO.

c. Kode Program ATIS

```
z = baca.Speak(" Radin Inten 2 information " + txtMetar2, SVSFlagsAsync)
TIME
WIND
SPEED
VISIBILITY
weather
clouds
FEET
TEMPERATURE
DEW
QNH
z = baca.Speak(" Runway " + " One Four ", SVSFlagsAsync)
z = baca.Speak(" Runway " + " Three Two ", SVSFlagsAsync)
z = baca.Speak(" End of Information " + txtMetar2, SVSFlagsAsync)
```

Gambar 4.69 Cuplikan Kode Program ATIS *Engine*

Cuplikan kode program di atas digunakan untuk memproduksi suara yang digunakan untuk membaca *text field* yang berisi data-data sebagai sebuah informasi cuaca.

d. Kode Program Pembuatan Laporan Harian Data METAR

```

Set rsglobal = ExecSQL(GetDSN, "SELECT * FROM metar " & _
  "WHERE date >='" & dtAwal.Value & "' AND " & _
  "date <='" & dtAwal.Value & "'")

If rsglobal.RecordCount > 0 Then
  openExcel True
  openworkBook App.Path & "\\LaporanHarianDataMetar.xls"

```

Gambar 4.70 Cuplikan Kode Program Membuat Laporan Harian Data METAR

Cuplikan kode program di atas digunakan untuk membuat laporan harian data METAR. Pertama-tama aplikasi mencari *record* data METAR sesuai tanggal yang diminta oleh pengguna unit BO, setelah data ditemukan *record* tersebut akan diletakan pada *file* format laporan harian data METAR yang sebelumnya telah dibuat dalam aplikasi microsoft excell.

Dalam aplikasi ini *file* format laporan harian data cuaca diberi nama “LaporanHarianDataMetar.xls”.

e. Kode Program Pembuatan Laporan Periodik Data METAR

```

Set rsglobal = ExecSQL(GetDSN, "SELECT * FROM metar " & _
  "WHERE date >='" & dtAwal.Value & "' AND " & _
  "date <='" & dtAkhir.Value & "'")

If rsglobal.RecordCount > 0 Then
  openExcel True
  openworkBook App.Path & "\\LaporanPeriodikDataMetar.xls"

```

Gambar 4.71 Cuplikan Kode Membuat Laporan Periodik Data METAR

Cuplikan kode program di atas digunakan untuk membuat laporan periodik data METAR. Pertama-tama aplikasi mencari *record* data cuaca sesuai *range* tanggal yang diminta oleh pengguna unit BO, setelah data ditemukan *record* tersebut akan diletakan pada *file*

format laporan periodik data METAR yang sebelumnya telah dibuat dalam aplikasi microsoft excell.

Dalam aplikasi ini file format laporan periodik data cuaca ini diberi nama "LaporanPeriodikDataMetar.xls".

f. Kode Program Perekaman Suara

```
i = mciRekam("close all", 0&, 0, 0)
    i = mciRekam("open new type waveaudio alias capture", 0&, 0, 0)
    i = mciRekam("set capture channels 2", 0&, 0, 0)
    i = mciRekam("seek capture to start", 0&, 0, 0)
    i = mciRekam("set capture samplespersec 44100", 0&, 0, 0)
    i = mciRekam("set capture bitspersample 16", 0&, 0, 0)
    i = mciRekam("record capture", 0&, 0, 0)
```

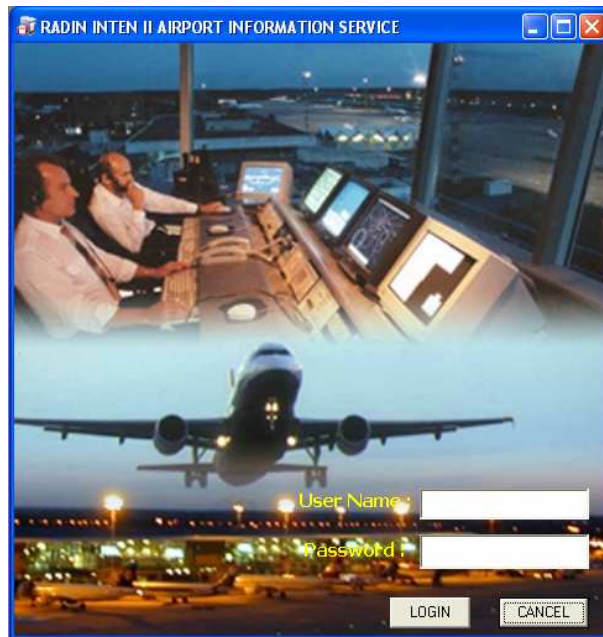
Gambar 4.72 Cuplikan Kode Program Perekaman Suara

Cuplikan kode program di atas digunakan oleh palikasi yang ditujukan untuk pengguna unit BO sebagai fasilitas untuk merekam suara sebagai sebuah informasi.

B. Pengujian (Tahapan *Testing*)

1. Otentifikasi *User*

Pengujian dimulai dengan tampilan form login. Melalui form ini *user* dapat melakukan otentifikasi *user id* dan *password* untuk dapat masuk ke aplikasi P-ATIS.

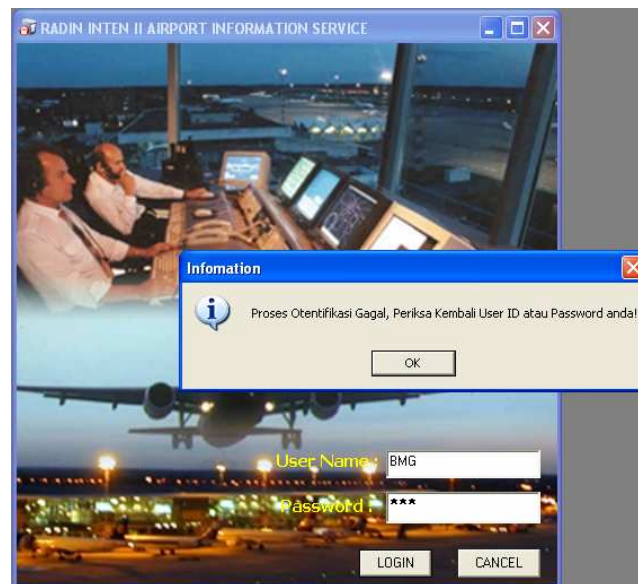


Gambar 4.73 Tampilan Form Otentifikasi *User*

1. Kegagalan Otentifikasi

a. Kesalahan Memasukan User ID Atau Password

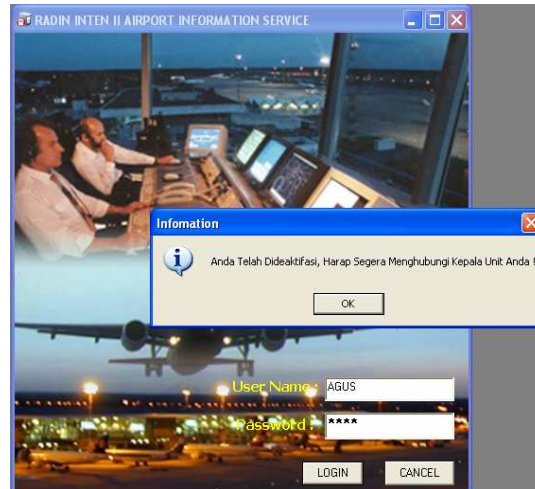
Pesan kesalahan otentifikasi ini muncul ketika seorang *user* yang mencoba untuk masuk ke aplikasi P-ATIS namun memasukan *user id* atau *password* yang salah.



Gambar 4.74 Tampilan Pesan Kesalahan *User ID* Atau *Password*

b. Kesalahan Karena Hak Akses *User* Dideaktifasi

Pesan kesalahan otentifikasi ini muncul ketika seorang *user* yang hendak masuk ke aplikasi P-ATIS ternyata tidak dapat masuk ke aplikasi, walaupun *user id* dan *password* yang telah dimasukan benar. Ini dikarenakan *user* yang bersangkutan ternyata telah dibekukan hak aksesnya untuk masuk ke aplikasi.

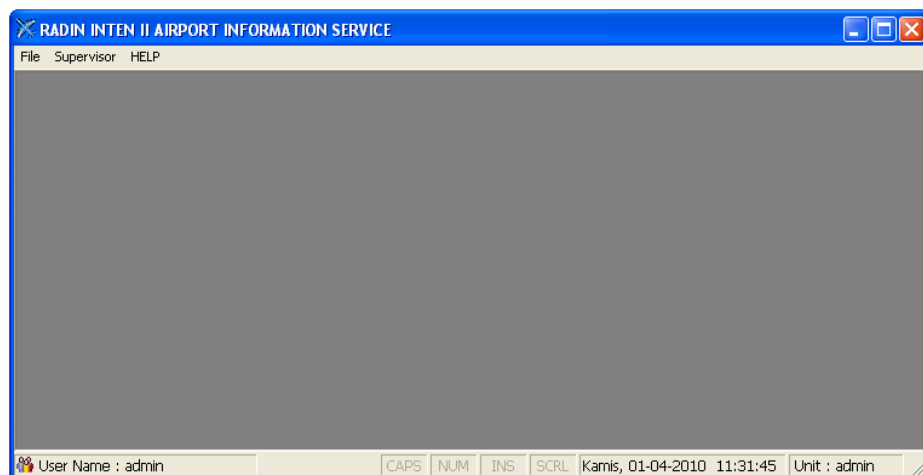


Gambar 4.75 Tampilan Pesan *User* Di Non-Aktifkan

2. Menu *User*

Setelah *user* berhasil melakukan otentifikasi, selanjutnya *user* akan dibawa ke dalam menu *user*. Di dalam menu *user* inilah *user* melakukan pengelolaan informasi. Menu *user* adalah suatu form yang memuat kegiatan apa saja yang dapat dilakukan oleh *user*. Menu tiap *user* berbeda-beda, tergantung oleh hak akses yang dimilikinya.

a. Menu *Supervisor*



Gambar 4.76 Tampilan Menu Admin

Ketika kita melakukan pengujian dengan melakukan otentifikasi sebagai seorang admin, maka setelah melewati tahap penyiangan hak akses akan tampil menu *user* seperti gambar di atas. Seorang admin memiliki dua buah sub-menu utama yaitu “File” dan “Supervisor” ditambah sebuah menu bantu “HELP”. Masing-masing sub-menu utama tersebut juga memiliki sub-menu tersendiri.

Di pojok kiri bawah terdapat informasi tentang siapa user yang sedang mengakses menu ini, kemudian terdapat status bar yang berisi informasi keyascii “CAPS”, “NUM”, “INS”, “SCRL”. Di pojok kanan bawah terdapat informasi tanggal dan jam terkini dalam format waktu Indonesia. Juga terdapat informasi unit dari *user*

1. Sub-Menu Utama File

Sub-menu utama file ini berisi dua buah sub-menu yaitu sub-menu Global Setup dan sub-menu Set Kepala Unit, ditambah sebuah sub-menu Log Off untuk keluar dari aplikasi. Masing masing sub-menu memberikan admin untuk melakukan pengolahan informasi.



Gambar 4.77 Tampilan Sub-Menu File Admin

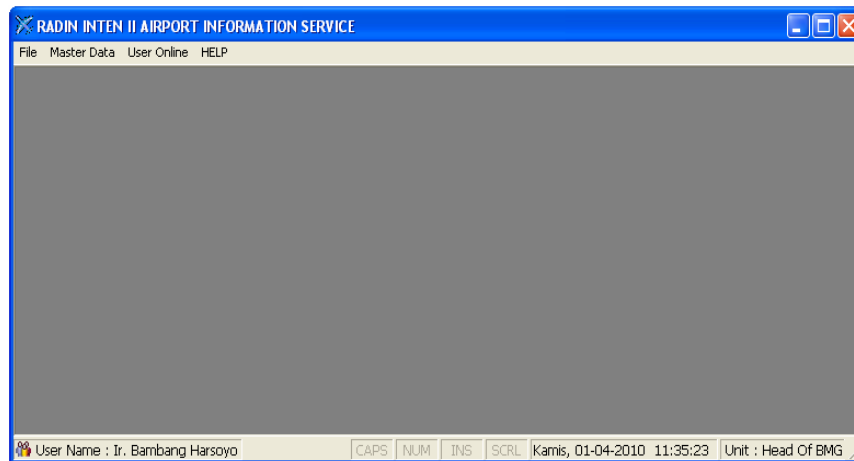
2. Sub-Menu Utama Supervisor

Sub-menu utama supervisor ini berisi satu buah sub-menu yaitu sub-menu Set User Menu. sub-menu set user menu ini memberikan admin untuk melakukan pengolahan informasi hak akses *user*.



Gambar 4.78 Tampilan Sub-Menu Utama Supervisor Admin

b. Menu Kepala Unit BMG



Gambar 4.79 Tampilan Menu Kepala Unit BMG

Ketika kita melakukan pengujian dengan melakukan otentifikasi sebagai seorang kepala unit BMG, maka setelah melewati tahap penyaringan hak akses akan tampil menu *user* seperti gambar di atas

Seorang kepala unit BMG memiliki tiga buah sub-menu utama yaitu “File” , “Master Data”, dan “User Online” ditambah sebuah menu bantu “HELP”. Masing-masing sub-menu utama tersebut juga memiliki sub-menu tersendiri.

Di pojok kiri bawah terdapat informasi tentang siapa *user* yang sedang mengakses menu ini, kemudian terdapat status bar yang berisi informasi keyascii “CAPS”, “NUM”, “INS”, “SCRL”. Di pojok kanan bawah terdapat informasi tanggal dan jam terkini dalam format waktu Indonesia. Juga terdapat informasi unit dari *user*.

1. Sub-Menu Utama File

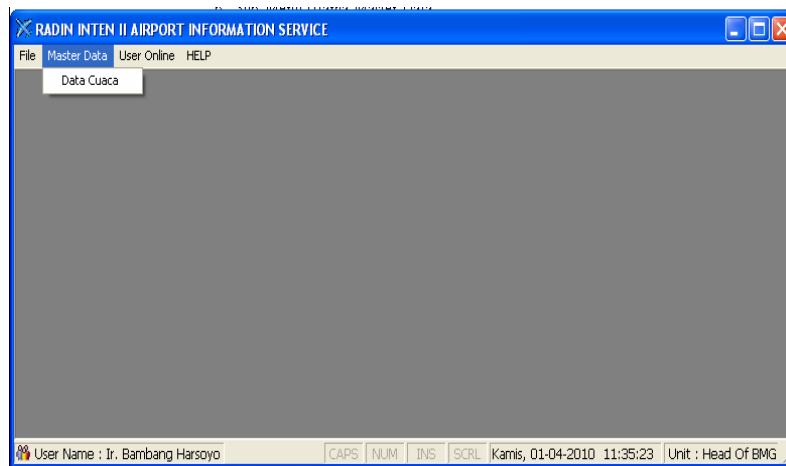
Sub-menu utama file ini berisi satu buah sub-menu yaitu sub-menu Set User BMG, ditambah sebuah sub-menu Log Off untuk keluar dari aplikasi. Sub-menu ini memberikan kepala unit BMG hak untuk mengelola pengguna unit BMG.



Gambar 4.80 Tampilan Sub-Menu File Kepala Unit BMG

2. Sub-Menu Utama Master Data

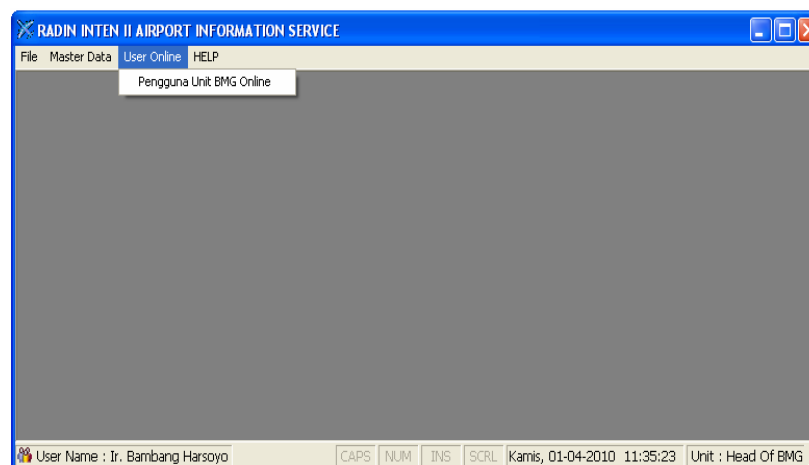
Sub-menu utama master data ini memiliki satu buah sub-menu lain yaitu sub-menu data cuaca. Sub-menu utama master data ini digunakan untuk melihat informasi.



Gambar 4.81 Tampilan Sub-Menu Utama Master Data KepalaUnit BMG

3. Sub-Menu Utama User Online

Sub-menu utama user online ini memiliki satu buah sub-menu yaitu sub menu Pengguna Unit BMG Online.



Gambar 4.82 Tampilan Sub-Menu Utama User Online KepalaUnit BMG

c. Menu Kepala Unit BO



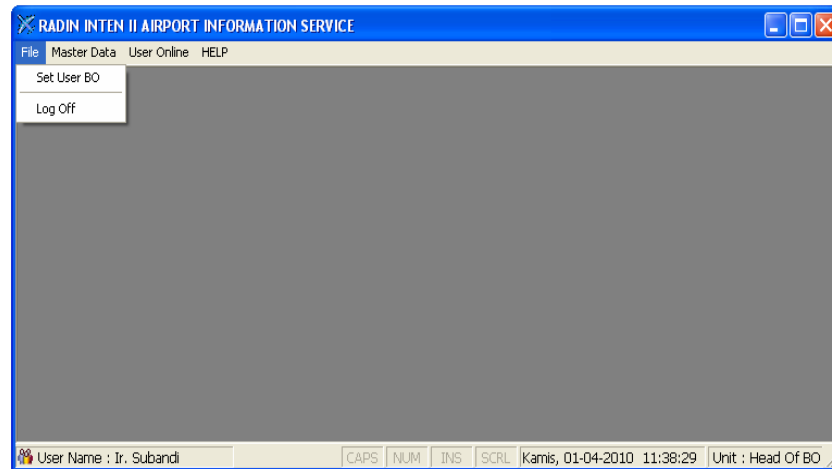
Gambar 4.83 Tampilan Menu Kepala Unit BO

Ketika kita melakukan pengujian dengan melakukan otentifikasi sebagai seorang kepala unit BO, maka setelah melewati tahap penyaringan hak akses akan tampil menu *user* seperti gambar di atas. Seorang kepala unit BO memiliki tiga buah sub-menu utama yaitu “File”, “Master Data”, dan “User Online” ditambah sebuah menu bantu “HELP”. Masing-masing sub-menu utama tersebut juga memiliki sub-menu tersendiri.

Di pojok kiri bawah terdapat informasi tentang siapa *user* yang sedang mengakses menu ini, kemudian terdapat status bar yang berisi informasi keyascii “CAPS”, “NUM”, “INS”, “SCRL”. Di pojok kanan bawah terdapat informasi tanggal dan jam terkini dalam format waktu Indonesia. Juga terdapat informasi unit dari *user*.

1. Sub-Menu Utama File

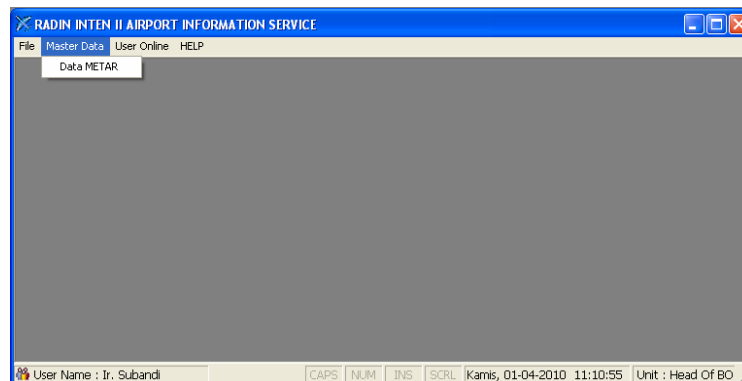
Sub-menu utama file ini berisi satu buah sub-menu yaitu sub-menu Set User BO, ditambah sebuah sub-menu Log Off untuk keluar dari aplikasi. Sub-menu ini memberikan kepala unit BO hak untuk mengelola pengguna unit BO.



Gambar 4.84 Tampilan Sub-Menu File Kepala Unit BO

2. Sub-Menu Utama Master Data

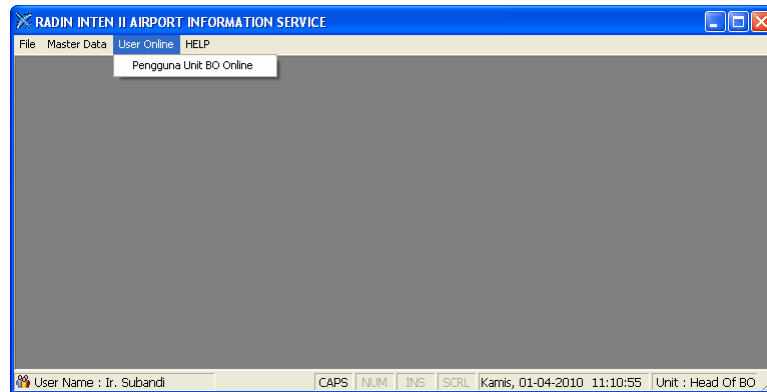
Sub-menu utama master data ini memiliki satu buah sub-menu lain yaitu sub-menu data METAR. Sub-menu utama master data ini digunakan untuk melihat informasi.



Gambar 4.85 Tampilan Sub-Menu Utama Master Data Kepala Unit BO

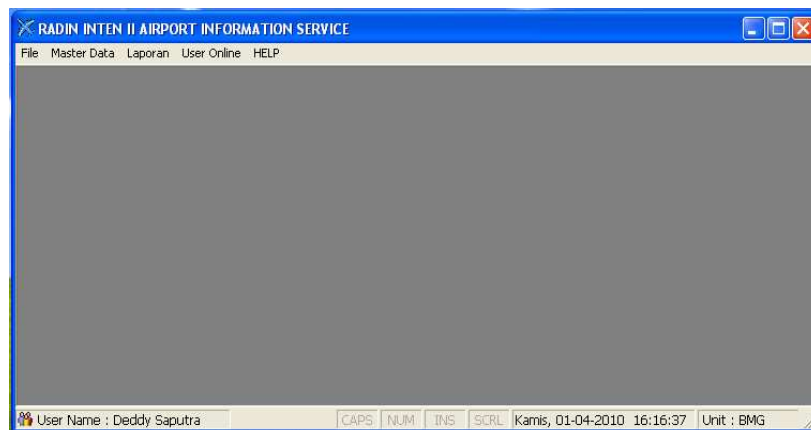
3. Sub-Menu Utama User Online

Sub-menu utama user online ini memiliki satu buah sub-menu yaitu sub menu Pengguna Unit BO Online.



Gambar 4.86 Tampilan Sub-Menu Utama User Online Kepala Unit BO

d. Menu Pengguna Unit BMG



Gambar 4.87 Tampilan Menu Pengguna Unit BMG

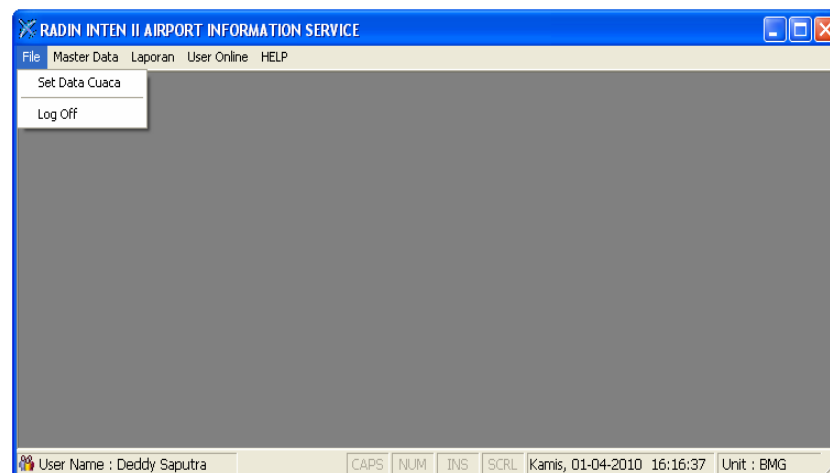
Ketika kita melakukan pengujian dengan melakukan otentifikasi sebagai seorang pengguna unit BMG, maka setelah melewati tahap penyaringan hak akses akan tampil menu *user* seperti gambar di atas. Seorang Pengguna Unit BMG memiliki empat buah sub-menu utama yaitu “File” , “Master Data”, “Laporan”, “User Online”

ditambah sebuah menu bantu “HELP”. Masing-masing sub-menu utama tersebut juga memiliki sub-menu tersendiri.

Di pojok kiri bawah terdapat informasi tentang siapa pengguna unit BMG yang sedang mengakses menu ini, kemudian terdapat status bar yang berisi informasi keyascii “CAPS”, “NUM”, “INS”, “SCRL”. Di pojok kanan bawah terdapat informasi tanggal dan jam terkini dalam format waktu Indonesia. Juga terdapat informasi unit dari *user*.

1. Sub-Menu Utama File

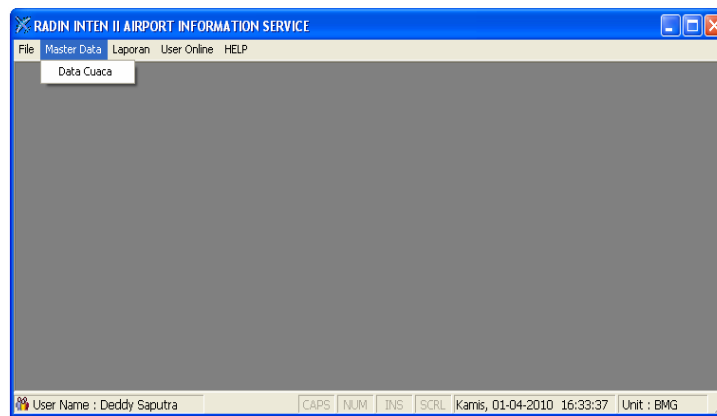
Sub-menu utama file ini berisi satu buah sub-menu yaitu sub-menu Set Data Cuaca, ditambah sebuah sub-menu Log Off untuk keluar dari aplikasi. Sub-menu ini memberikan pengguna unit BMG hak untuk mengelola data-data cuaca.



Gambar 4.88 Tampilan Sub-Menu Utama File Pengguna Unit BMG

2. Sub-Menu Utama Master Data

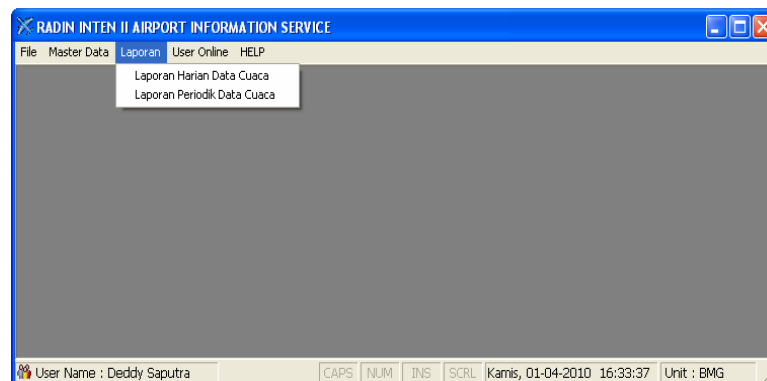
Sub-menu utama master data ini memiliki satu buah sub-menu lain yaitu sub-menu data cuaca. Sub-menu utama master data ini digunakan untuk melihat informasi.



Gambar 4.89 Tampilan Sub-Menu Utama Master Data Pengguna Unit BMG

3. Sub-Menu Utama Laporan

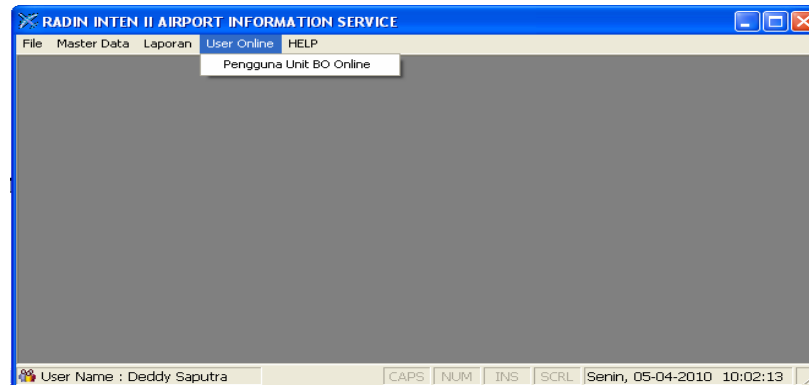
Sub-menu utama laporan memiliki sub-menu lain yaitu sub-menu laporan harian data cuaca dan sub-menu laporan periodik data cuaca. Kedua sub-menu tersebut memiliki fungsi untuk membantu pengguna unit BMG dalam membuat dan mencetak laporan data cuaca.



Gambar 4.90 Tampilan Sub-Menu Utama Laporan Pengguna Unit BMG

4. Sub-Menu Utama User Online

Sub-menu utama user online ini memiliki satu buah sub-menu yaitu sub menu Pengguna Unit BO Online.



Gambar 4.91 Tampilan Sub-Menu Utama User Online Pengguna Unit BMG

e. Menu Pengguna Unit BO



Gambar 4.92 Tampilan Menu Pengguna Unit BO

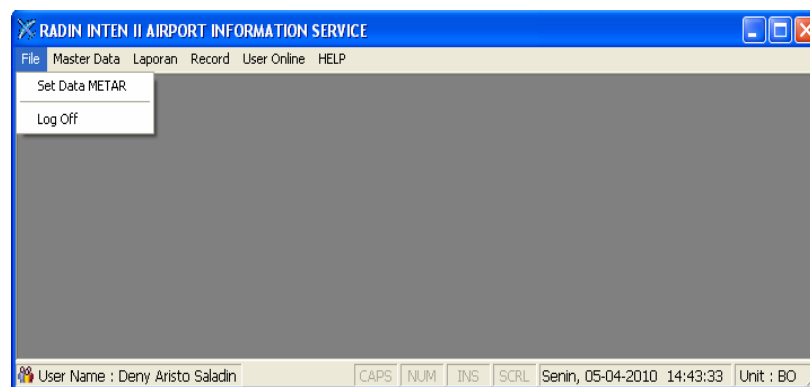
Ketika kita melakukan pengujian dengan melakukan otentifikasi sebagai seorang pengguna unit BO, maka setelah melewati tahap penyaringan hak akses akan tampil menu *user* seperti gambar di atas. Seorang Pengguna Unit BO memiliki lima buah sub-menu utama yaitu “File”, “Master Data”, “Laporan”, “Record”, “User Online”

ditambah sebuah menu bantu “HELP”. Masing-masing sub-menu utama tersebut juga memiliki sub-menu tersendiri.

Di pojok kiri bawah terdapat informasi tentang siapa pengguna unit BMG yang sedang mengakses menu ini, kemudian terdapat status bar yang berisi informasi keyascii “CAPS”, “NUM”, “INS”, “SCRL”. Di pojok kanan bawah terdapat informasi tanggal dan jam terkini dalam format waktu Indonesia. Juga terdapat informasi unit dari *user*.

1. Sub-Menu Utama File

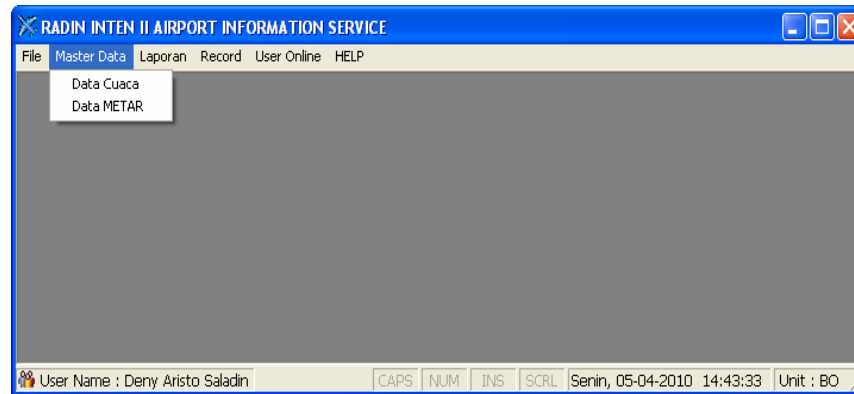
Sub-menu utama file ini berisi satu buah sub-menu yaitu sub-menu Set Data METAR, ditambah sebuah sub-menu Log Off untuk keluar dari aplikasi. Sub-menu ini memberikan pengguna unit BO hak untuk mengisi data-data METAR.



Gambar 4.93 Tampilan Sub-Menu Utama File Pengguna Unit BO

2. Sub-Menu Utama Master Data

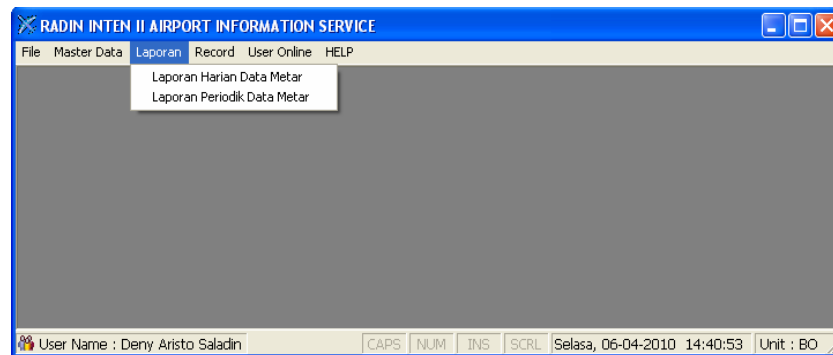
Sub-menu utama master data ini memiliki dua buah sub-menu lain yaitu sub-menu data cuaca dan sub-menu data METAR. Sub-menu utama master data ini digunakan untuk melihat informasi.



Gambar 4.94 Tampilan Sub-Menu Utama Master Data Pengguna Unit BO

3. Sub-Menu Utama Laporan

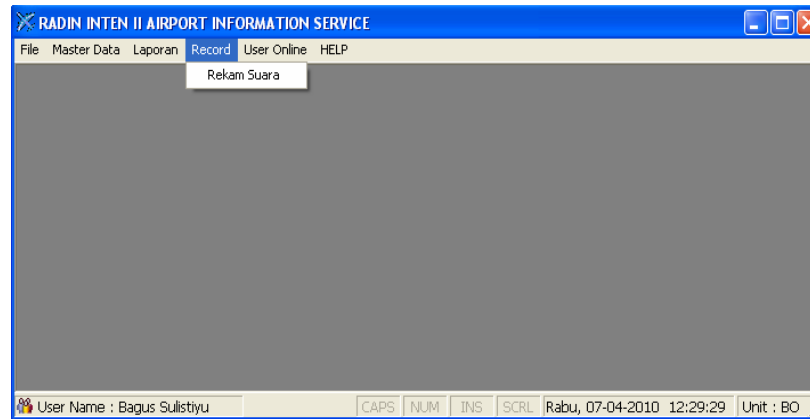
Sub-menu utama laporan memiliki sub-menu lain yaitu sub-menu laporan harian data METAR dan sub-menu laporan periodik data METAR. Kedua sub-menu tersebut memiliki fungsi untuk membantu pengguna unit BO dalam membuat dan mencetak laporan data METAR



Gambar 4.95 Tampilan Sub-Menu Utama Laporan Pengguna Unit BO

4. Sub-Menu Utama Record

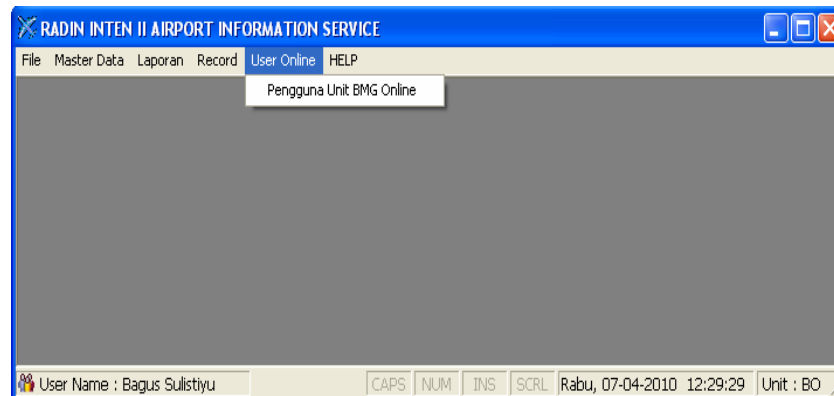
Sub-menu utama record ini memiliki satu buah sub-menu lain yaitu sub-menu Rekam Suara.



Gambar 4.96 Tampilan Sub-Menu Utama Record Pengguna Unit BO

5. Sub-Menu Utama User Online

Sub-menu utama user online ini memiliki satu buah sub-menu yaitu sub menu Pengguna Unit BMG Online.



Gambar 4.97 Tampilan Sub-Menu Utama User Online Pengguna Unit BO

3. Olah Informasi

a. User Hak Akses *Supervisor*

1. Sub-Menu Global Setup

Sub-menu global setup memberikan *supervisor* suatu form pengisian informasi untuk data-data bandara

Gambar 4.98 Tampilan Sub-Menu Global Setup

Seorang *supervisor* bertugas mengisi informasi global bandara yang menggunakan aplikasi P-ATIS ini. Form global setup ini hanya didesain mampu menampung satu set *record* sebuah bandara saja. Jadi satu aplikasi P-ATIS digunakan untuk satu buah bandara.

Setelah data global bandara dimasukan oleh *supervisor* selanjutnya aplikasi akan mengecek kevalidan dari data tersebut, apakah ada data yang tidak sesuai.

Gambar 4.99 Data Global Bandara Yang Dimasukan Tidak Valid

Form global setup sudah didesain untuk meminimalkan kesalahan saat memasukan informasi, terutama berkaitan dengan karakter ASCII. Jadi, apabila informasi tersebut tidak memerlukan karakter huruf, angka atau karakter spesial lainnya, maka karakter yang tidak diperlukan tersebut tidak akan tampil.

Setelah data global bandara valid dimasukan sebagai sebuah *record* ke dalam *database*, maka saat form global setup akan dibuka lagi, data global bandara tersebut akan langsung mengisi *text field*, bukan berupa *text field* kosong.

2. Sub-Menu Set Kepala Unit

Seorang *supervisor* juga mempunyai hak penuh untuk memutuskan apakah kepala unit tersebut diizinkan atau tidak untuk mengoperasikan aplikasi P-ATIS.

Tampilan sub-menu set kepala unit ini merupakan realisasi dari proses masukan data kepala unit dan proses otorisasi kepala unit. Melalui form ini *supervisor* melakukan pengolahan informasi kepala unit BMG dan kepala unit BO. Termasuk memutuskan mengaktifasi atau mendeaktifasi hak akses kepala unit.

The screenshot shows a window titled "Form Administrator" with a table and a form below it. The table has columns: User id, Nama, Unit, Akses, Status, and Password. The form has fields for User ID, Nama, Password, Unit, and Akses, along with buttons for "Tambah", "Koreksi", "Simpan", "Hapus", "Batal", and "Keluar".

User id	Nama	Unit	Akses	Status	Password
BMG	Ir. Bambang Haryono	Head Of BMG	Activated	Offline	bmg
BO	Ir. Subandi	Head Of BO	Activated	Offline	bo

Form fields:

- User ID: bmg
- Nama: Ir. Bambang Haryono
- Password: [masked]
- Unit: Head Of BMG
- Akses: Activated

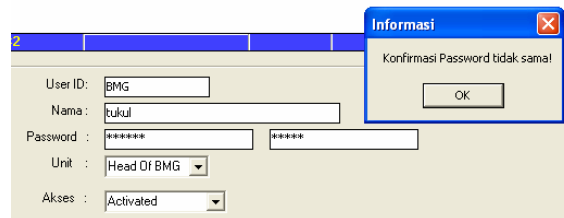
Buttons: Tambah, Koreksi, Simpan, Hapus, Batal, Keluar

Gambar 4.100 Tampilan Sub-Menu Set Kepala Unit

Informasi yang berkaitan dengan data kepala unit ini juga dibatasi.. Terutama berkaitan dengan karakter ASCII. Jadi, apabila informasi tersebut tidak memerlukan karakter huruf, angka atau karakter

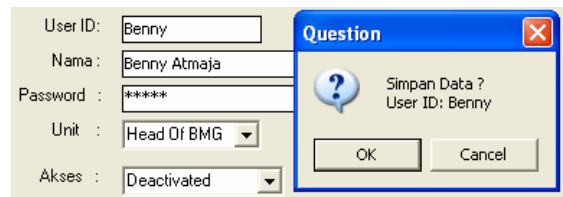
spesial lainnya, maka karakter yang tidak diperlukan tersebut tidak akan tampil.

Sebelum data kepala unit dimasukkan ke dalam *database* sebagai sebuah *record* oleh *supervisor* aplikasi akan mengecek kevalidan dari data tersebut, apakah ada data yang tidak sesuai.



Gambar 4.101 Data Kepala Unit Yang Dimasukan Tidak Valid

Setelah data kepala unit berhasil melewati pengecekan kevalidan data, akan muncul suatu form yang kembali meminta konfirmasi apakah *supervisor* yakin untuk menyimpan set *record* tersebut.

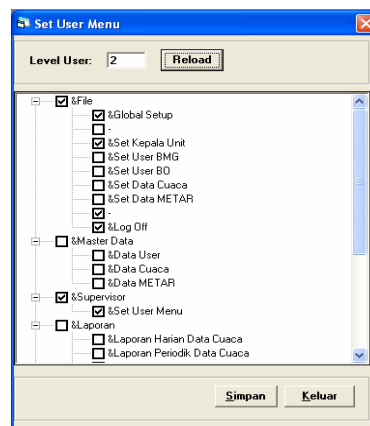


Gambar 4.102 Konfirmasi Penyimpanan Data Kepala Unit

Jika *supervisor* menekan tombol "OK", maka data kepala unit akan tersimpan ke dalam *database*. Namun, jika *supervisor* menekan tombol "Cancel", maka data tidak akan tersimpan ke dalam *database*.

3. Sub-Menu Set User Menu

Sub-menu set user menu ini memberikan admin kemampuan untuk mengelola hak akses kepala unit dan pengguna unit baik BMG atau BO. Tiap *user* memiliki urutan level user yang berbeda.



Gambar 4.103 Tampilan Form Set *User* Menu

Tampilan form di atas merupakan realisasi dari proses 3.14.4, 3.14.7, 3.14.8. supervisor hanya diharuskan mencentang sub-menu yang terdapat dalam form tersebut dan diklasifikasikan menurut *level user*. Setiap kepala unit dan pengguna unit, diklasifikasikan ke dalam *level user* yang berbeda satu sama lain. Untuk menyimpan data hak akses, supervisor cukup mengetikkan angka untuk level user dan selanjutnya menekan tombol “Simpan”.

Untuk mengoreksi sub-menu, *supervisor* cukup mengetikkan angka dari level user dan menekan tombol “Reload”. Dari kamus data secara default, level user untuk supervisor adalah “2”, kepala unit BMG adalah “3”, kepala unit BO adalah “4”, pengguna unit BMG adalah “5”, dan pengguna unit BO adalah “6”.

b. User Hak Akses Kepala Unit BMG

1. Sub-Menu Set User BMG

Sub-menu set user BMG memberikan kepala unit BMG suatu form pengolahan informasi pengguna unit BMG. Termasuk memutuskan mengaktifasi atau mendeaktifasi hak akses pengguna unit BMG ke aplikasi P-ATIS.

User id	Nama	Unit	Akses	Status	Password
AGUS	Agus Susanto	BMG	Deactivated	Offline	agus
BOIM	Immanuel Hotmian Hulapea	BMG	Activated	Offline	boim
DEDDY	Deddy Saputra	BMG	Activated	Offline	deddy

User ID:

Nama:

Password:

Unit:

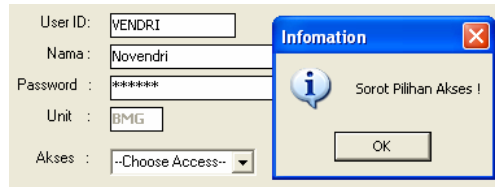
Akses:

Tambah Koreksi Simpan Hapus Batal Keluar

Gambar 4.104 Tampilan Sub-Menu Set User BMG

Informasi yang berkaitan dengan data pengguna unit BMG ini juga dibatasi. Jadi, apabila informasi tersebut tidak memerlukan karakter huruf, angka atau karakter spesial lainnya, maka karakter yang tidak diperlukan tersebut tidak akan tampil.

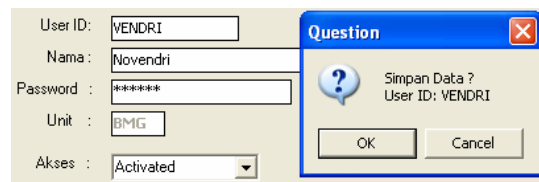
Sebelum data pengguna unit dimasukan ke dalam *database* sebagai sebuah *record* oleh kepala unit BMG, aplikasi akan mengecek kevalidan dari data tersebut, apakah ada data yang akan dimasukan sesuai atau tidak.



Gambar 4.105 Data Pengguna Unit BMG Yang Dimasukan Tidak Valid

Untuk data pengguna unit BMG, *text field* “Unit” secara *default* akan merujuk pada unit kerja calon pengguna unit tersebut dan tidak bisa dirubah.

Setelah data pengguna unit BMG berhasil melewati pengecekan kevalidan data, akan muncul suatu form yang kembali meminta konfirmasi apakah kepala unit BMG yakin untuk menyimpan set *record* tersebut.



Gambar 4.106 Konfirmasi Penyimpanan Data Pengguna Unit BMG

Jika kepala unit BMG menekan tombol “OK”, maka data pengguna unit BMG akan tersimpan ke dalam *database*. Namun, jika kepala unit BMG menekan tombol “Cancel”, maka data tidak akan tersimpan ke dalam *database*.

Untuk melakukan pengeditan atau penghapusan data pengguna unit BMG, kepala unit BMG cukup menyorot set *record* data

kepala unit yang akan dilakukan pengeditan atau penghapusan selanjutnya menekan tombol “Koreksi”.

DEDDY	Deddy Saputra	BMG	Activated	Offline	deddy
-------	---------------	-----	-----------	---------	-------

User ID:	DEDDY
Nama:	Deddy Saputra
Password:	****
Unit:	BMG
Akses:	Activated

Tambah Koreksi **Simpan** Hapus Batal Keluar

Gambar 4.107 Tampilan Form Edit Dan Hapus Data Pengguna Unit BMG

Jika kepala unit BMG hanya ingin mengedit data pengguna unit BMG, cukup dengan menekan tombol “Simpan”. Dan jika kepala unit BMG ingin menghapus data pengguna unit BMG, cukup dengan menekan tombol “Hapus”.

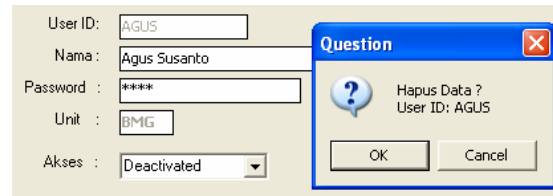
Sama seperti proses penyimpanan, saat akan dilakukan proses pengeditan atau penghapusan data, aplikasi sekali lagi akan melakukan pengecekan kevalidan data dan konfirmasi penyimpanan (kecuali proses penghapusan yang tidak melakukan pengecekan kevalidan data).

AGUS	Agus Susanto	BMG	Deactivated	Offline	agus
BOIM	Immanuel Hotmian Hulapea	BMG	Activated	Offline	boim
DEDDY	Deddy Saputra	BMG	Activated	Offline	deddy

User ID:	AGUS
Nama:	Agus Susanto
Password:	****
Unit:	BMG
Akses:	Activated

Question
 ? Simpan Data ?
 User ID: AGUS
 OK Cancel

Gambar 4.108 Tampilan Form Edit Data Pengguna Unit BMG



Gambar 4.109 Tampilan Form Hapus Data Pengguna Unit BMG

Pada proses pengeditan data, saat kepala unit BMG menekan tombol “OK”, maka set *record* data pengguna unit BMG baru akan memperbaharui data pengguna unit BMG yang lama. Dan pada proses penghapusan data, saat kepala unit BMG menekan tombol “OK”, maka data pengguna unit tersebut akan dihapus dari *database*.

2. Sub-Menu Data Cuaca

Sub-menu data cuaca memberikan informasi data cuaca yang sebelumnya sudah dimasukan oleh pengguna unit BMG kepada kepala unit BMG.

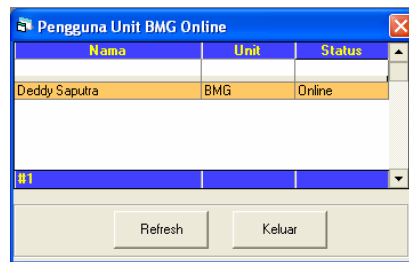
Date	Time	Direction	Speed	Visibility	Weather	Clouds	Feet	Temperature	DEW	qNH	Active User
15/03/2010	13:11	123	12	111	NILI	BKN	11	11	11	111	Raden Manang
15/03/2010	19:56	101	24	10010	RAIN	SKC	567811	45	11	1222	Deddy Saputra
16/03/2010	12:17	123	11	11	RAIN	FEW	11	11	11	11	Deddy Saputra
17/03/2010	11:10	567	23	245	FG	SCT	555	23	23	1011	Deddy Saputra
28/03/2010	12:27	234	12	1345	FG	SCT	11000	23	24	1011	Deddy Saputra
29/03/2010	17:42	12	12	12	RAIN	SCT	12	12	12	12	Deddy Saputra

Gambar 4.110 Tampilan Sub-Menu Data Cuaca

Melalui form ini, kepala unit BMG dapat melihat seluruh data cuaca yang tersimpan dalam *database* yang telah dimasukan oleh pengguna unit BMG.

3. Sub-Menu Pengguna Unit BMG Online

Sub-menu ini memberikan informasi siapakah pengguna unit BMG yang sedang bertugas mengolah data cuaca kepada kepala unit BMG.



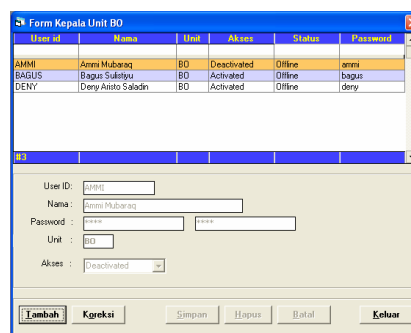
Gambar 4.111 Tampilan Sub-Menu Pengguna Unit BMG Online

Tampilan sub-menu di atas merupakan realisasi dari proses 3.2.7. Melalui form ini, kepala unit BMG dapat melihat siapakah pengguna unit BMG yang sedang bertugas.

c. User Hak Akses Kepala Unit BO

1. Sub-Menu Set User BO

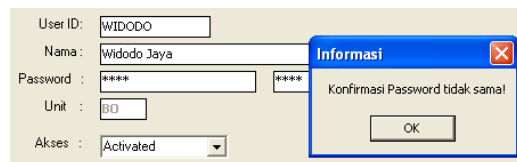
Sub-menu set user BO memberikan kepala unit BO suatu form pengolahan informasi pengguna unit BO. Termasuk memutuskan mengaktifasi atau mendeaktifasi hak akses pengguna unit BO.



Gambar 4.112 Tampilan Sub-Menu Set User BO

Informasi yang berkaitan dengan data pengguna unit BO ini juga dibatasi. Jadi, apabila informasi tersebut tidak memerlukan karakter huruf, angka atau karakter spesial lainnya, maka karakter yang tidak diperlukan tersebut tidak akan tampil.

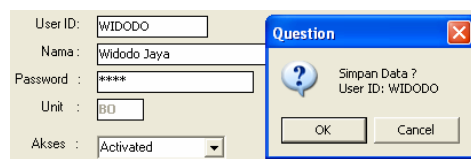
Sebelum data pengguna unit dimasukkan ke dalam *database* sebagai sebuah *record* oleh kepala unit BO, aplikasi akan mengecek kevalidan dari data tersebut, apakah ada data yang akan dimasukkan sesuai atau tidak.



Gambar 4.113 Data Pengguna Unit BO Yang Dimasukan Tidak Valid

Untuk data pengguna unit BO, *text field* "Unit" secara *default* akan merujuk pada unit kerja calon pengguna unit tersebut dan tidak bisa dirubah.

Setelah data pengguna unit BO berhasil melewati pengecekan kevalidan data, akan muncul suatu form yang kembali meminta konfirmasi apakah kepala unit BO yakin untuk menyimpan set *record* tersebut.



Gambar 4.114 Konfirmasi Penyimpanan Data Pengguna Unit BO

Jika kepala unit BO menekan tombol “OK”, maka data pengguna unit BO akan tersimpan ke dalam *database*. Namun, jika kepala unit BO menekan tombol “Cancel”, maka data tidak akan tersimpan ke dalam *database*.

Untuk melakukan pengeditan atau penghapusan data pengguna unit BO, kepala unit BO cukup menyrot set *record* data kepala unit yang akan dilakukan pengeditan atau penghapusan selanjutnya menekan tombol “Koreksi”.

DENY	Deny Aristo Saladin	BO	Activated	Offline	deny
User ID: <input type="text" value="DENY"/> Nama: <input type="text" value="Deny Aristo Saladin"/> Password: <input type="password" value="****"/> <input type="password" value="****"/> Unit: <input type="text" value="BO"/> Akses: <input type="text" value="Activated"/> <ul style="list-style-type: none"> Activated Deactivated 					
<input type="button" value="Tambah"/> <input type="button" value="Koreksi"/> <input type="button" value="Simpan"/> <input type="button" value="Hapus"/> <input type="button" value="Batal"/> <input type="button" value="Keluar"/>					

Gambar 4.115 Tampilan Form Edit Dan Hapus Data Pengguna Unit BO

Jika kepala unit BO hanya ingin mengedit data pengguna unit BO, cukup dengan menekan tombol “Simpan”. Dan jika kepala unit BO ingin menghapus data pengguna unit BO, cukup dengan menekan tombol “Hapus”.

Sama seperti proses penyimpanan, saat akan dilakukan proses pengeditan atau penghapusan data, aplikasi sekali lagi akan melakukan pengecekan kevalidan data dan konfirmasi penyimpanan (kecuali proses penghapusan yang tidak melakukan pengecekan kevalidan data).

AMMI	Nama Mubaraq	BO	Deactivated	Offline	ammi
BAGUS	Bagus Sulistyhu	BO	Activated	Offline	bagus
DENY	Deny Aristo Saladin	BO	Activated	Offline	deny

User ID :	<input type="text" value="AMMI"/>
Nama :	Amni Mubaraq
Password :	<input type="password" value="****"/>
Unit :	<input type="text" value="BO"/>
Akses :	Activated

Question

Simpan Data ?
User ID: AMMI

OK Cancel

Gambar 4.116 Tampilan Form Edit Data Pengguna Unit BO

User ID :	<input type="text" value="BAGUS"/>
Nama :	Bagus Sulistyhu
Password :	<input type="password" value="****"/>
Unit :	<input type="text" value="BO"/>
Akses :	Activated

Question

Hapus Data ?
User ID: BAGUS

OK Cancel

Gambar 4.117 Tampilan Form Hapus Data Pengguna Unit BO

Pada proses pengeditan data, saat kepala unit BO menekan tombol “OK”, maka set *record* data pengguna unit BO baru akan memperbaharui data pengguna unit BO yang lama. Dan pada proses penghapusan data, saat kepala unit BO menekan tombol “OK”, maka data pengguna unit tersebut akan dihapus dari *database*.

2. Sub-Menu Data METAR

Sub-menu data METAR memberikan informasi data METAR yang sebelumnya sudah dimasukan oleh pengguna unit BO kepada kepala unit BO.

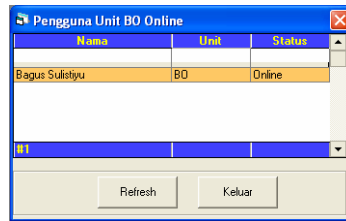
Date	Time	METAR	User Active
17/03/2010	11:10	Alla	Deny Aristo Saladin
28/03/2010	12:27	Foxtrol	Deny Aristo Saladin
28/03/2010	12:27	India	Deny Aristo Saladin

Keluar

Gambar 4.118 Tampilan Sub-Menu Data METAR

3. Sub-Menu Pengguna Unit BO Online

Sub-menu ini memberikan informasi siapakah pengguna unit BO yang sedang bertugas kepada kepala unit BO.



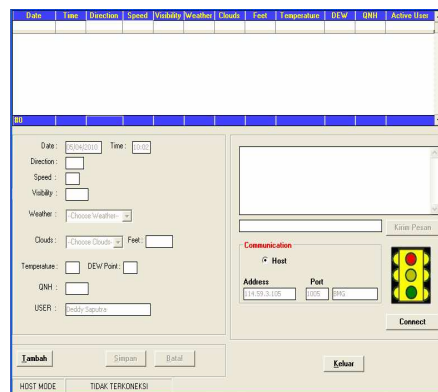
Gambar 4.119 Tampilan Sub-Menu Pengguna Unit BO Online

Tampilan sub-menu di atas merupakan realisasi dari proses 3.5.7. Melalui form ini, kepala unit BO dapat melihat siapakah pengguna unit BO yang sedang bertugas.

d. User Hak Akses Pengguna Unit BMG

1. Sub-Menu Set Data Cuaca

Sub-menu set data cuaca memberikan pengguna unit BMG suatu form pengolahan informasi data cuaca. Melalui form ini juga pengguna unit BMG juga dapat melakukan komunikasi dengan pengguna unit BO.



Gambar 4.120 Tampilan Sub-Menu Set Data Cuaca

The image shows a web-based form for entering weather data. The form is organized into several rows of input fields. At the top, there are two text boxes for 'Date' (containing '03/05/2010') and 'Time' (containing '15:13'). Below these are three more text boxes for 'Direction', 'Speed', and 'Visibility'. The 'Weather' field is a dropdown menu with the text '-Choose Weather-'. The 'Clouds' field is another dropdown menu with '-Choose Clouds-' and a 'Feet' text box next to it. Below that are two text boxes for 'Temperature' and 'DEW Point'. The 'QNH' field is a single text box. The 'USER' field contains the name 'Deddy Saputra'. At the bottom of the form, there are three buttons: 'Tambah' (Add), 'Simpan' (Save), and 'Batal' (Cancel).

Gambar 4.121 Tampilan *Field* Set Data Cuaca

Tampilan sub-menu set data cuaca secara penuh dapat dilihat dari gambar 4.118. Sedangkan pada gambar 4.119 itu merupakan *field* yang digunakan pengguna unit BMG untuk menambahkan data-data cuaca terkini. Untuk *field* “Date”, “Time” akan secara otomatis terisi keterangan waktu saat itu. Untuk *field* “USER”, akan secara otomatis terisi nama pengguna unit BMG yang sedang bertugas saat itu.

Untuk menambahkan data-data cuaca terkini terlebih dahulu pengguna unit BMG harus menekan tombol “Tambah”, barulah seorang pengguna unit BMG dapat mengisi *field-field* set data cuaca tersebut. Informasi yang berkaitan dengan data cuaca ini juga dibatasi. Form set data cuaca ini sudah didesain untuk meminimalkan kesalahan saat memasukan informasi, terutama berkaitan dengan karakter ASCII. Jadi, apabila informasi tersebut tidak memerlukan karakter huruf atau karakter spesial

lainnya, maka karakter yang tidak diperlukan tersebut tidak akan tampil.

Setelah data cuaca dimasukkan oleh pengguna unit BMG selanjutnya aplikasi akan mengecek kevalidan dari data tersebut, apakah ada data yang tidak sesuai.

The screenshot shows a weather data entry form with the following fields: Date: 03/05/2010, Time: 17:19, Direction: 120, Speed: 4, Visibility: 5, Weather: RAIN, Clouds: SKC, Feet: 2200, Temperature: 20, DEW Point: 19, QNH: (empty), and USER: Deddy Saputra. An 'Information' dialog box is open, displaying an information icon and the text 'Isi Field QNH' with an 'OK' button.

Gambar 4.122 Data Cuaca Yang Dimasukkan Tidak Valid

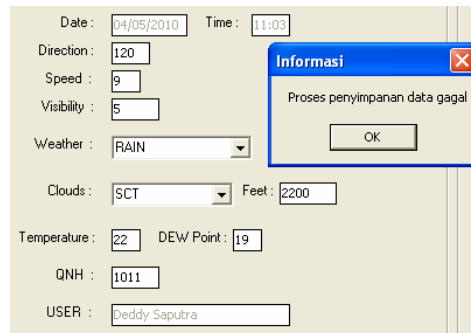
Setelah data cuaca berhasil melewati pengecekan kevalidan data, akan muncul suatu form yang kembali meminta konfirmasi apakah pengguna unit BMG yakin untuk menyimpan set *record* tersebut

The screenshot shows the same weather data entry form, but with the following updated values: Date: 04/05/2010, Time: 10:59, Direction: 120, Speed: 9, Visibility: 5, Weather: FOG, Clouds: SCT, Feet: 2200, Temperature: 22, DEW Point: 19, QNH: 1014, and USER: Deddy Saputra. A 'Question' dialog box is open, displaying a question mark icon and the text 'Simpan Data ?' followed by the current data values: Date: 04/05/2010, Time: 10:59, Direction: 120, Speed: 9, Visibility: 5, Weather: FOG, Clouds: SCT, Feet: 2200, Temperature: 22, DEW Point: 19, QNH: 1014. The dialog box has 'OK' and 'Cancel' buttons.

Gambar 4.123 Konfirmasi Penyimpanan Data Cuaca

Jika Pengguna unit BMG menekan tombol “OK”, maka data cuaca akan tersimpan ke dalam *database*. Namun, jika pengguna unit

BMG menekan tombol “Cancel”, maka data tidak akan tersimpan ke dalam *database* dan akan muncul sebuah *message box*.



Gambar 4.124 *Message Box* Pembatalan Penyimpanan Data Cuaca

Record data yang ditampilkan dalam sub-menu set data cuaca ini adalah *record* yang terbaru yang baru saja dimasukan oleh pengguna unit BMG yang bertugas saat itu.

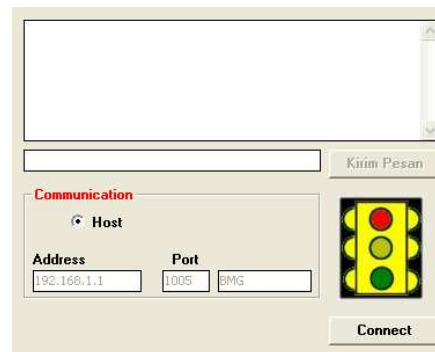
Date	Time	Direction	Speed	Visibility	Weather	Clouds	Feet	Temperature	DEW	QNH	Active User
04/05/2010	11:26	136	10	5	NILL	BKN	2200	26	24	1014	Deddy Saputra

Gambar 4.125 Tampilan *Record* Data Cuaca Sub-Menu Set Data Cuaca

Ketika pengguna unit BMG memutuskan untuk memasukan data cuaca terbaru lagi, maka ketika pengguna unit BMG menekan tombol “Simpan”, data cuaca terbaru secara otomatis akan tampil menggantikan data cuaca sebelumnya.

Pengguna unit BMG haruslah selalu berkomunikasi dengan pengguna unit BO, terutama untuk selalu memberikan informasi data cuaca yang terbaru. Untuk dapat melakukan komunikasi dengan pengguna unit BO,

maka aplikasi P-ATIS ini menyediakan fasilitas *chatting* sebagai sarana komunikasi antara pengguna unit BMG dengan pengguna unit BO.



Gambar 4.126 Tampilan Fasilitas *Chatting* Pengguna Unit BMG

Alamat IP komputer yang sebelumnya telah diatur akan secara otomatis tertera dalam *text field* address dari aplikasi *chatting* ini. *Port* yang digunakan untuk komunikasi ini secara default adalah port 1005. *Port* ini tidak dapat diubah oleh pengguna unit BMG. Karena untuk mencegah adanya *error* dari aplikasi terutama dalam komunikasi. Pada komunikasi dua arah dalam *chatting*, kedua pengguna unit haruslah berada dalam *port chat* yang sama.

Untuk dapat terhubung dengan pengguna unit BO melalui fasilitas chat ini, maka pengguna unit BMG terlebih dahulu harus menekan tombol “Connect” dari fasilitas *chat* ini. Pada saat pengguna unit BMG belum terkoneksi atau telah terkoneksi dengan pengguna unit BO, maka di sebelah pojok kiri bawah dari sub-menu set data cuaca ini akan terdapat informasi.



Gambar 4.127 Informasi Status Koneksi Pengguna Unit BMG Tidak Terkoneksi

HOST MODE	LISTENING ON PORT: 1005
-----------	-------------------------

Gambar 4.128 Informasi Status Koneksi Pengguna Unit BMG Mencari Koneksi

HOST MODE	CLIENT CONNECTED
-----------	------------------

Gambar 4.129 Informasi Status Koneksi Pengguna Unit BMG Telah Terkoneksi

Saat pengguna unit BMG telah terkoneksi dengan pengguna unit BO, maka akan muncul suatu aplikasi yang secara otomatis menghitung lamanya durasi kedua pengguna unit terhubung melalui fasilitas *chat* ini.

Address	Port
192.168.1.1	1005
	BMG

Time :
Minute(s) 10 Seconds

Gambar 4.130 Aplikasi Penghitung Durasi Koneksi Pengguna Unit BMG

Untuk dapat mengirim pesan komunikasi ke pengguna unit BO, pengguna unit BMG cukup mengetikkan pesan komunikasi pada *text field* di sebelah tombol “Kirim Pesan”, kemudian menekan tombol “Kirim Pesan” atau langsung menekan tombol *enter* pada *keyboard*. Secara otomatis pesan yang terkirim akan tercantum di *text box* yang terdapat di atas *text field* untuk mengetik pesan.

BMG: Update data cuaca terbaru	
<input type="text"/>	Kirim Pesan

Gambar 4.131 Pesan Komunikasi Pengguna Unit BMG

Ketika komunikasi chat ini terputus, bisa terjadi karena pengguna unit BO menutup fasilitas *chat* atau karena pengguna unit BO keluar dari aplikasi P-ATIS, maka aplikasi akan memberikan informasi putusnya koneksi tersebut kepada pengguna unit BMG.



Gambar 4.132 Informasi Komunikasi Dengan Pengguna Unit BO Terputus

Setelah pengguna unit BMG menekan tombol “OK”, maka koneksi dengan pengguna unit BO akan kembali terhubung dengan syarat pengguna BO harus mencoba mengkoneksikan kembali aplikasi *chat*nya dengan pengguna unit BMG.

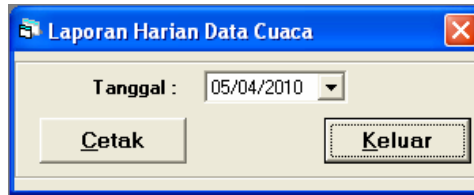
2. Sub-Menu Data Cuaca

Sub-menu data cuaca memberikan informasi data cuaca yang sebelumnya sudah dimasukan oleh pengguna unit BMG yang bertugas. Berupa kumpulan *record* data cuaca yang tersimpan dalam *database*.

Gambar 4.133 Tampilan Sub-Menu Data Cuaca Pengguna Unit BMG

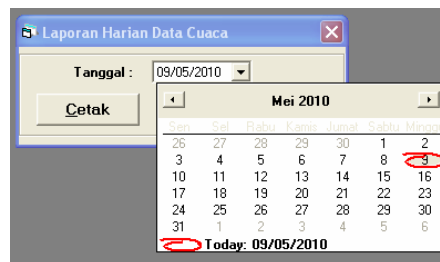
3. Sub-Menu Laporan Harian Data Cuaca

Seluruh *record* data cuaca yang telah dibuat oleh pengguna unit BMG setiap harinya harus selalu dibuat laporan. Melalui sub-menu laporan harian data cuaca inilah pengguna unit BMG membuat laporan data cuaca dengan periode waktu harian.



Gambar 4.134 Tampilan Sub-Menu Laporan Harian Data Cuaca

Pertama-tama pengguna unit BMG harus memasukan tanggal dari *record* cuaca yang diinginkan dan selanjutnya menekan tombol “Cetak”.



Gambar 4.135 Tampilan Memilih Tanggal Data Cuaca

RADIN INTEN II
Dinas Bantahan, Pelembagaan, dan Pengawasan
Indonesi
Laporan Harian
15/03/2010

No.	Date	Time	Wind Direction	Wind Speed	Visibility	Present Weather	Clouds	Feet	Temperature	DEW Point	QNH	User On Duty
1.	15/03/2010	19:56	101	24	11	FOG	SKC	2100	22	15	1010	Dedy Saputra

Gambar 4.136 Bentuk Laporan Harian Data Cuaca

Setelah menekan tombol “Cetak”, pengguna unit BMG akan dibawa ke area *print preview* dari microsoft excell. Pengguna unit BMG tidak mempunyai izin untuk mengedit atau memanipulasi data cuaca untuk laporan harian data cuaca ini. Selanjutnya pengguna unit BMG dapat mencetak laporan data tersebut. jenis kertas yang dipakai secara default adalah jenis F4 dengan *print orientation landscape*.

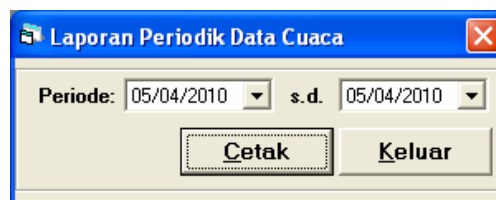
Jika *record* data cuaca yang diinginkan ternyata tidak ditemukan, maka akan muncul sebuah pesan kesalahan.



Gambar 4.137 Pesan Kesalahan Harian Data Cuaca Tidak Ditemukan

4. Sub-Menu Laporan Periodik Data Cuaca

Sub-menu laporan periodik data cuaca ini digunakan oleh pengguna unit BMG untuk membuat laporan data cuaca dengan periode mingguan, bulanan, atau dalam periode waktu tertentu.



Gambar 4.138 Tampilan Sub-Menu Laporan Periodik Data Cuaca

Sedikit berbeda dengan cara pembuatan laporan data cuaca harian, pertama-tama pengguna unit BMG harus memasukan *range* dari *record* data cuaca yang ingin dicetak, kemudian barulah menekan tombol “Cetak”.

RADIN INTEN II
 Dinas Perhubungan, Kecamatan Sukaraja - Lampung
 INMINTA
 Laporan Per Periode
 08/03/2010 s.d. 16/03/2010

No.	Date	Time	Wind Direction	Wind Speed	Visibility	Present Weather	Clouds	Feet	Temperature	DBW Point	QNH	User On Duty
1.	08/04/2010	09:53	200	11	12	RAIN	BKN	2200	15	13	1001	Immanuel Hotmian Hutapea
2.	15/03/2010	19:56	101	24	11	FOG	SKC	2100	22	15	1010	Deddy Saputra
3.	16/03/2010	12:17	270	6	10	HAZE	FEW	2300	17	14	1011	Deddy Saputra

Gambar 4.139 Bentuk Laporan Periodik Data Cuaca

Setelah menekan tombol “Cetak”, pengguna unit BMG akan dibawa ke area *print preview* dari microsoft excell. Pengguna unit BMG tidak mempunyai izin untuk mengedit atau memanipulasi data cuaca untuk laporan harian data cuaca ini. Selanjutnya pengguna unit BMG dapat mencetak laporan data tersebut. jenis kertas yang dipakai secara default adalah jenis F4 dengan *print orientation landscape*.

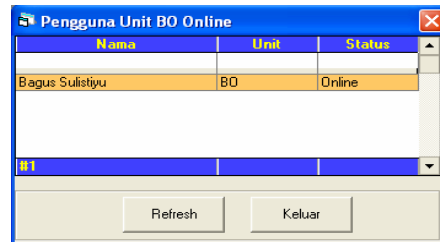
Jika *record* data cuaca yang diinginkan ternyata tidak ditemukan, maka akan muncul sebuah pesan kesalahan.



Gambar 4.140 Pesan Kesalahan Periode Data Cuaca Tidak Ditemukan

5. Sub-Menu Pengguna Unit BO Online

Sub-menu ini memberikan informasi siapakah pengguna unit BO yang sedang bertugas kepada pengguna unit BMG.

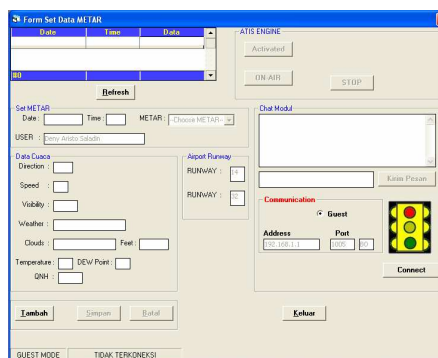


Gambar 4.141 Tampilan Sub-Menu Pengguna Unit BO Online

e. User Hak Akses Pengguna Unit BO

1. Sub-Menu Set Data METAR

Sub-menu set data METAR memberikan pengguna unit BO suatu form pengolahan informasi data METAR. Melalui form ini juga pengguna unit BO juga dapat melakukan komunikasi dengan pengguna unit BMG.



Gambar 4.142 Tampilan Sub-Menu Set Data METAR

Date	Time	Data
04/05/2010	11:26	NEW

Refresh

Set METAR

Date: 04/05/2010 Time: 11:26 METAR: --Choose METAR--

USER: Deny Aristo Saladin

Data Cuaca

Direction: 136

Speed: 10

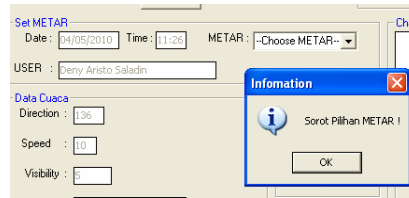
- Alfa
- Bravo
- Charlie
- Delta
- Echo
- Foxtrot
- Golf
- Hotel

Gambar 4.143 Tampilan Tampilan *Field* Set Data METAR

Tampilan sub-menu set data METAR secara penuh dapat dilihat dari gambar 4.140. Sedangkan pada gambar 4.141 itu merupakan *field* yang digunakan pengguna unit BO untuk mengisi data METAR. Untuk *field* “Date”, “Time” akan secara otomatis terisi keterangan waktu saat itu. Untuk *field* “USER”, akan secara otomatis terisi nama pengguna unit BO yang sedang bertugas saat itu. Untuk kotak METAR, pengguna unit BO harus memilih keterangan untuk data METAR yang akan disimpan.

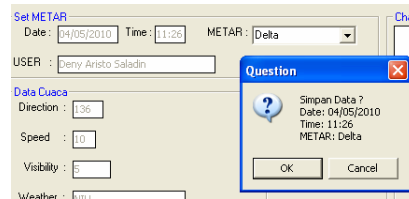
Untuk mengisi dan menyimpan data METAR terlebih dahulu pengguna unit BO harus menekan tombol “Tambah”, barulah seorang pengguna unit BO dapat mengisi dan menyimpan data METAR tersebut.

Setelah data METAR diisi oleh pengguna unit BO selanjutnya aplikasi akan mengecek kevalidan dari data tersebut, apakah ada data yang tidak sesuai.



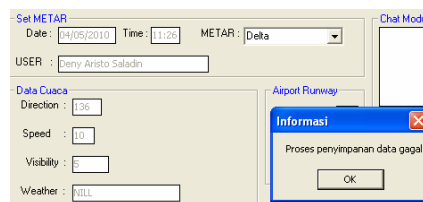
Gambar 4.144 Data METAR Yang Dimasukan Tidak Valid

Setelah data METAR berhasil melewati pengecekan kevalidan data, akan muncul suatu form yang kembali meminta konfirmasi apakah pengguna unit BO yakin untuk menyimpan set *record* tersebut



Gambar 4.145 Konfirmasi Penyimpanan Data METAR

Jika Pengguna unit BO menekan tombol “OK”, maka data METAR akan tersimpan ke dalam *database*. Namun, jika pengguna unit BO menekan tombol “Cancel”, maka data tidak akan tersimpan ke dalam *database* dan akan muncul sebuah *message box*.



Gambar 4.146 *Message Box* Pembatalan Penyimpanan Data METAR

Record data yang ditampilkan dalam sub-menu set data METAR ini adalah *record* data cuaca terbaru yang baru saja dimasukan oleh pengguna unit BMG yang bertugas saat itu. Data cuaca ini memiliki parameter, yaitu “NEW”.

Date	Time	Data
04/05/2010	11:26	NEW

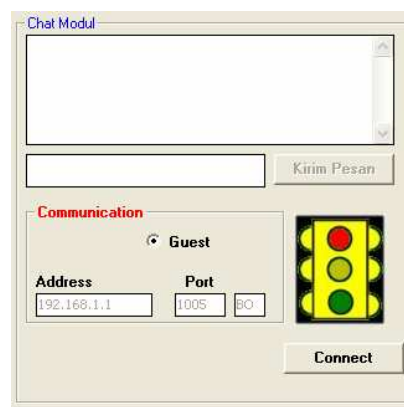
Refresh

Gambar 4.147 Tampilan *Record* Data Cuaca Sub-Menu Set Data METAR

Selanjutnya, jika pengguna unit BO ingin mendapatkan informasi data cuaca terbaru yang telah dimasukan oleh pengguna unit BMG, pengguna unit BO cukup menekan tombol “Refresh”.

Pengguna unit BO haruslah selalu berkomunikasi dengan pengguna unit BMG, terutama untuk selalu mendapatkan informasi data cuaca yang terbaru.

Untuk dapat melakukan komunikasi dengan pengguna unit BMG, maka aplikasi P-ATIS ini menyediakan fasilitas *chatting* sebagai sarana komunikasi antara pengguna unit BO dengan pengguna unit BMG.



Gambar 4.148 Tampilan Fasilitas *Chatting* Pengguna Unit BO

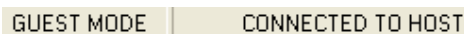
Alamat IP komputer yang tampil dalam *field address* dalam *chat modul* pengguna unit BO ini adalah alamat IP *address* pengguna unit BMG. *Port* yang digunakan untuk komunikasi ini secara default adalah port 1005. *Port* ini tidak dapat diubah oleh pengguna unit BO. Karena untuk mencegah adanya *error* dari aplikasi terutama dalam komunikasi. Pada komunikasi dua arah dalam *chatting*, kedua pengguna unit haruslah berada dalam *port chat* yang sama.

Untuk dapat terhubung dengan pengguna unit BMG melalui fasilitas chat ini, maka pengguna unit BO terlebih dahulu harus menekan tombol “Connect” dari fasilitas *chat* ini. Pada saat pengguna unit BO belum terkoneksi atau telah terkoneksi dengan pengguna unit BMG, maka di sebelah pojok kiri bawah dari sub-menu set data cuaca ini akan terdapat informasi.



A horizontal status bar with two sections. The left section is labeled 'GUEST MODE' and the right section is labeled 'TIDAK TERKONEKSI'.

Gambar 4.149 Informasi Status Koneksi Pengguna Unit BO Tidak Terkoneksi



A horizontal status bar with two sections. The left section is labeled 'GUEST MODE' and the right section is labeled 'CONNECTED TO HOST'.

Gambar 4.150 Informasi Status Koneksi Pengguna Unit BO Telah Terkoneksi

Saat pengguna unit BO telah terkoneksi dengan pengguna unit BMG, maka akan muncul suatu aplikasi yang secara otomatis menghitung lamanya durasi kedua pengguna unit terhubung melalui fasilitas *chat* ini.

Address	Port
192.168.1.1	1005 BO

Time
Minute[s] 3 Seconds

Gambar 4.151 Aplikasi Penghitung Durasi Koneksi Pengguna Unit BO

Untuk dapat mengirim pesan komunikasi ke pengguna unit BMG, pengguna unit BO cukup megetikan pesan komunikasi pada *text field* di sebelah tombol “Kirim Pesan”, kemudian menekan tombol “Kirim Pesan” atau langsung menekan tombol *enter* pada *keyboard*. Secara otomatis pesan yang terkirim akan tercantum di *text box* yang terdapat diatas *text field* untuk mengetik pesan

BO: Ada Data Terbaru..?	
<input type="text"/>	Kirim Pesan

Gambar 4.152 Pesan Komunikasi Pengguna Unit BO

Ketika komunikasi chat ini terputus, bisa terjadi karena pengguna unit BMG menutup fasilitas *chat* atau karena pengguna unit BMG keluar dari aplikasi P-ATIS, maka aplikasi akan memberikan informasi putusnya koneksi tersebut kepada pengguna unit BO.



Gambar 4.153 Informasi Komunikasi Dengan Pengguna Unit BMG Terputus

Setelah pengguna unit BO menekan tombol “OK”, maka koneksi dengan pengguna unit BMG akan kembali terhubung dengan syarat pengguna BO harus mencoba mengkoneksikan kembali aplikasi *chat*nya dengan pengguna unit BMG.



Gambar 4.154 Informasi Komunikasi Gagal Terkoneksi

Pesan kesalahan ini muncul ketika koneksi chat antara pengguna unit BO dan pengguna unit BMG tidak bisa terkoneksi, bisa dikarenakan IP *address* di antara kedua komputer pengguna unit tersebut belum diatur, atau terdapat kerusakan di media penghubung.



Gambar 4.155 Fasilitas ATIS

Setelah pengguna unit BO menyimpan data METAR ke dalam *database*, maka secara otomatis fasilitas ATIS akan dapat diakses oleh pengguna unit BO. Fasilitas ini akan mengeluarkan suara dengan membaca teks data cuaca yang terdapat dalam sub-menu ini. Parameter “Ready” menandakan fasilitas ATIS ini sudah siap untuk digunakan.



Gambar 4.156 ATIS Siap Membaca Teks Data Cuaca

Setelah pengguna unit BO menekan tombol “Activated”, akan muncul parameter “ON-AIR” yang menandakan fasilitas ATIS ini sudah siap untuk membaca teks data cuaca. Selanjutnya pengguna unit BO cukup menekan tombol “ON-AIR” dan fasilitas ATIS ini akan membaca teks data cuaca tersebut berulang-ulang selama kurang-lebih 10 menit, atau sampai pengguna unit BO menekan tombol “STOP”. Kemudian akan muncul parameter “Stop” yang menandakan proses pembacaan teks data cuaca dihentikan.



Gambar 4.157 Fasilitas ATIS Dihentikan

2. Sub-Menu Data Cuaca

Sub-menu data cuaca memberikan informasi data cuaca yang sebelumnya sudah dimasukan oleh pengguna unit BMG yang bertugas. Berupa kumpulan *record* data cuaca yang tersimpan dalam *database*. Melalui sub-menu ini pengguna unit BO dapat melihat keseluruhan data cuaca yang tersimpan.

Date	Time	Direction	Speed	Visibility	Weather	Clouds	Feet	Temperature	DEW	QNH	Active User
15/03/2010	13:11	123	12	111	NULL	BKN	11	11	111	111	Raden Manang
15/03/2010	19:56	101	24	10010	RAIN	SKC	957811	45	11	1222	Deddy Saputra
15/03/2010	12:17	123	11	11	RAIN	FEW	11	11	11	11	Deddy Saputra
17/03/2010	11:10	567	23	245	FOG	SCT	555	23	23	1011	Deddy Saputra
28/03/2010	12:27	234	12	1345	FOG	SCT	11000	23	24	1011	Deddy Saputra
29/03/2010	17:42	12	12	12	RAIN	SCT	12	12	12	12	Deddy Saputra

Gambar 4.158 Tampilan Sub-Menu Data Cuaca Pengguna Unit BO

3. Sub-Menu Data METAR

Sub-menu data METAR memberikan informasi data METAR yang sebelumnya sudah dimasukan oleh pengguna unit BO yang bertugas. Berupa kumpulan *record* data METAR yang tersimpan dalam *database*.

Date	Time	METAR	User Active
17/03/2010	11:10	Alfa	Dery Aristo Saladin
28/03/2010	12:27	Foxtrot	Dery Aristo Saladin
28/03/2010	12:27	India	Dery Aristo Saladin

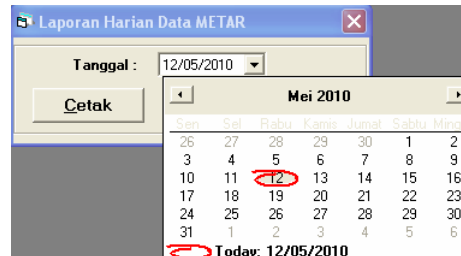
Gambar 4.159 Tampilan Sub-Menu Data METAR

4. Sub-Menu Laporan Harian Data METAR

Sub-menu laporan harian data METAR ini digunakan oleh pengguna unit BO untuk membuat laporan data METAR dengan periode waktu harian.

Gambar 4.160 Tampilan Sub-Menu Laporan Harian Data METAR

Pertama-tama pengguna unit BO harus memasukan tanggal dari *record* METAR yang diinginkan dan selanjutnya menekan tombol “Cetak”.



Gambar 4.161 Tampilan Memilih Tanggal Data METAR

RADIN INTEN II
 Desa Branti Jaya, Kecamatan Natar - Lampung
 Indonesia

Laporan Harian
 17/03/2010
 Frekuensi ATIS : 126,78 MHz

No.	Date	Time	METAR	Active User
1.	17/03/2010	11:10	Alfa	Deny Aristo Saladin

Gambar 4.162 Bentuk Laporan Harian Data METAR

Setelah menekan tombol “Cetak”, pengguna unit BO akan dibawa ke area *print preview* dari microsoft excell. Pengguna unit BO tidak mempunyai izin untuk mengedit atau memanipulasi data METAR untuk laporan harian data METAR ini. Selanjutnya pengguna unit BO dapat mencetak laporan data tersebut. jenis kertas yang dipakai secara default adalah jenis F4 dengan *print orientation landscape*.

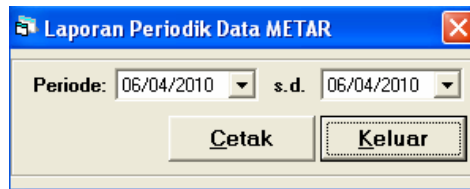
Jika *record* data METAR yang diinginkan ternyata tidak ditemukan, maka akan muncul sebuah pesan kesalahan.



Gambar 4.163 Pesan Kesalahan Harian Data METAR Tidak Ditemukan

5. Sub-Menu Laporan Periodik Data METAR

Sub-menu laporan periodik data METAR ini digunakan oleh pengguna unit BO untuk membuat laporan data METAR dengan periode mingguan, bulanan, atau dalam periode waktu tertentu.



Gambar 4.164 Tampilan Sub-Menu Laporan Periodik Data METAR

Yang harus dilakukan terlebih dahulu adalah, pengguna unit BO harus memasukan *range* dari *record* data METAR yang ingin dicetak, kemudian barulah menekan tombol “Cetak”.

RADIN INTEN II
Desa Branti Jaya, Kecamatan Natar - Lampung
Indonesia

Laporan Per Periode
17/03/2010 s.d. 28/05/2010
Frekuensi ATIS : 126,78 MHz

No.	Date	Time	METAR	Active User
1.	17/03/2010	11:10	Alfa	Deny Aristo Saladin
2.	28/03/2010	12:27	India	Deny Aristo Saladin
3.	28/03/2010	12:27	Foxtrot	Deny Aristo Saladin

Gambar 4.165 Bentuk Laporan Periodik Data METAR

Setelah menekan tombol “Cetak”, pengguna unit BO akan dibawa ke area *print preview* dari microsoft excell. Pengguna unit BO tidak mempunyai izin untuk mengedit atau memanipulasi data METAR untuk laporan periodik data METAR ini. Selanjutnya pengguna unit BO dapat mencetak laporan data tersebut. jenis kertas yang dipakai secara default adalah jenis F4 dengan *print orientation landscape*.

Jika *record* data METAR yang diinginkan ternyata tidak ditemukan, maka akan muncul sebuah pesan kesalahan

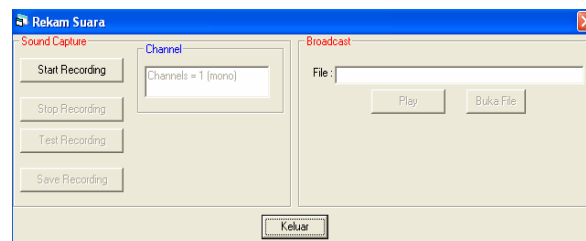


Gambar 4.166 Pesan Kesalahan Periode Data METAR Tidak Ditemukan

6. Sub-Menu Rekam Suara

Sub-menu rekam suara ini digunakan oleh pengguna unit BO

Untuk merekam suara sebagai pesan darurat.

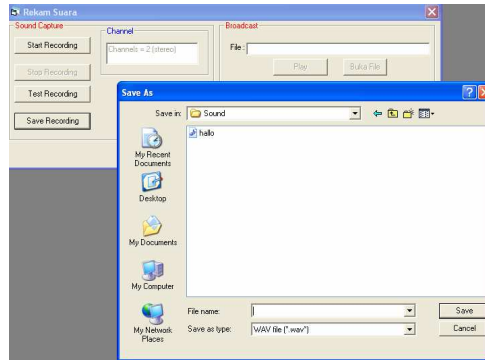


Gambar 4.167 Tampilan Sub-Menu Rekam Suara

Untuk melakukan perekaman suara, pengguna unit BO terlebih dahulu harus menekan tombol “Start Recording” kemudian barulah pengguna unit BO merekam suara dengan menggunakan *headset*. Kotak bertuliskan “Channel” dalam sub-menu ini memberikan informasi jenis *channel* audio yang dipakai untuk merekam yang sebelumnya sudah diatur pada komputer. Apakah jenis *stereo* atau *mono*.

Untuk menghentikan proses perekaman suara, pengguna unit BO cukup menekan tombol “Stop Recording”, maka proses perekaman suara akan dihentikan. Untuk melakukan uji dengar, pengguna unit BO dapat melakukannya dengan menekan tombol “Test Recording”, maka pengguna unit BO dapat mendengarkan terlebih dahulu hasil rekaman suaranya.

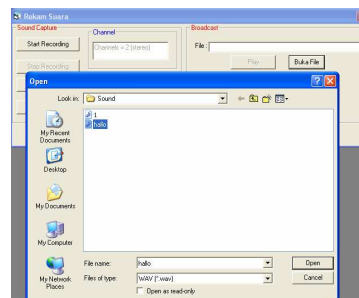
Jika pengguna unit BO merasa kurang puas dengan hasil rekamannya, pengguna unit BO dapat melakukan perekaman kembali dengan langsung menekan tombol “Start Recording” kembali. Setelah pengguna BO merasa hasil dari rekaman suaranya sudah cukup jelas dan baik, pengguna unit BO dapat melakukan penyimpanan rekaman suara tersebut dengan menekan tombol “Save Recording”.



Gambar 4.168 Penyimpanan Rekaman Suara

Secara *default*, tempat untuk melakukan penyimpanan rekaman suara ini adalah pada *folder* “Sound” di dalam *folder* tempat menyimpan aplikasi P-ATIS. *File* inilah yang nantinya akan diputar secara terus-menerus sebagai pesan darurat.

Untuk melakukan pemutaran *file*, pertama-tama pengguna unit BO harus menekan tombol “Buka File”, dan mencari *file* suara yang sebelumnya sudah direkam. Setelah menemukan file rekaman suara tersebut, untuk memutarinya pengguna unit BO cukup menekan tombol “Play”, dan akan berhenti jika pengguna unit BO menekan tombol “Stop”.

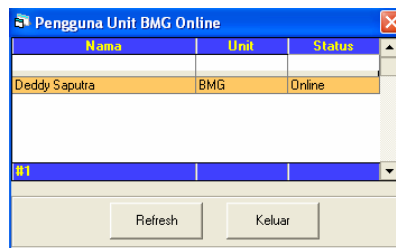


Gambar 4.169 Mengambil Rekaman Suara

Ketika sedang melakukan pemutaran rekaman suara, pengguna unit BO tidak dapat melakukan perekaman suara. Dari hasil pengujian diketahui bahwa suara dari proses perekaman tercampur dengan suara dari proses pemutaran suara. Ini bisa disebabkan karena penggunaan dari penggunaan kanal yang sama untuk proses perekaman dan pemutaran suara. Sehingga suara rekaman tercampur dengan suara rekaman yang sedang diputar.

7. Sub-Menu Pengguna Unit BMG Online

Sub-menu ini memberikan informasi siapakah pengguna unit BMG yang sedang bertugas kepada pengguna unit BO.



Nama	Unit	Status
Deddy Saputra	BMG	Online

#1

Refresh Keluar

Gambar 4.170 Tampilan Sub-Menu Pengguna Unit BMG Online

V. KESIMPULAN DAN SARAN

A. Simpulan

1. Fasilitas ATIS dapat diimplementasikan menggunakan *personal computer* (PC). Tetapi, dalam perancangannya tetap harus mengikuti standar-standar yang telah ditetapkan *international civil aviation organization* (ICAO), baik dalam regulasi yang mengatur ATIS itu sendiri atau juga yang berkaitan dengan komunikasi penerbangan.
2. Dengan menggunakan PC sebagai sebuah fasilitas ATIS, kita dapat menggunakan *voice engine* berbasis *text to speech* sebagai *library* yang digunakan untuk memproduksi suara.
3. Dengan merancang fasilitas ATIS menggunakan konsep *client/server* dapat tercipta suatu aliran komunikasi yang lebih terorganisir dan efisien dalam menyampaikan informasi cuaca.
4. Dengan merancang fasilitas ATIS menggunakan konsep sistem informasi dapat tercipta suatu kondisi yang efektif dalam menyampaikan informasi cuaca penerbangan.
5. Kekeliruan-kekeliruan yang mungkin terjadi dalam memasukan data-data cuaca harus diperhatikan dan diminimalkan. Karena informasi cuaca dalam dunia penerbangan merupakan salah satu hal yang penting.

B. Saran

1. Akan lebih efisien lagi jika data-data cuaca dapat diakuisisi secara otomatis dan langsung dari instrumen-instrumen yang digunakan untuk mengambil data-data cuaca.
2. Kemungkinan untuk pengembangan selanjutnya, fasilitas ini dapat dikembangkan menggunakan konsep nirkabel.
3. Regulasi mengenai ATIS ini juga harus diperhatikan, karena kemungkinan adanya perubahan aturan-aturan mengenai fasilitas ATIS ini, yang membuat perancang harus melakukan *maintenance* terhadap aplikasi ini.

DAFTAR PUSTAKA

- www.ilmuterbang.com. 12 Desember 2008. Diakses pukul 12:41.
- www.MihirEnterprises.com. 14 Desember 2008. Diakses Pukul 12:22.
- www.geocities.com/periyadi2000/tugas/standar_nasional.pdf.
8 Januari 2009. Diakses Pukul 11:08.
- www.geocities.com/BSN/SNI_ref02.pdf. 13 Januari 2009. Diakses Pukul 11:08
- Hartono, Jogyanto. 1999. *Pengenalan Komputer*. Penerbit ANDI. Yogyakarta.
- www.freewebs.com/siskapipin/konsepSI.doc. 6 Februari 2009. Diakses Pukul 16:23.
- www.118.98.171.131/Dokumen/Oleh-oleh%201/jis/Client_Server.doc.
15 Februari 2009. Diakses Pukul 13:22.
- Coolen, John. 1984. *Komunikasi elektronika*. Erlangga. Jakarta.
- Nugroho, Bunafit. 2002. *Belajar Visual Basic 6.0*. Elex Media Komputindo. Jakarta.
- Hutabarat, Bernaridho. 2005. *Pemrograman MySQL*. Dian Rakyat. Jakarta.
- www.evodevelopment.com/SQLyog.pdf. 27 Januari 2009. Diakses Pukul 20:00.
- Pressman, Roger S, Ph.D. 1997. *Rekayasa Perangkat Lunak: Pendekatan Praktisi*. Penerbit ANDI. Yogyakarta.
- <http://oc.itb.ac.id/TTS.pdf>. 16 Januari 2009. Diakses Pukul 16:30).

LAMPIRAN

LAMPIRAN A (Kode Program)

1. Main Form

```
Public UserName As String
Public UserID As String 'menampilkan field form
Public UserLevel As Long

Private Sub MDIForm_Load()
    SetMenu Me, UserLevel 'menampilkan form menu olah informasi
End Sub

Private Sub mnuGlobalSetup_Click(Index As Integer)
    frmGlobalSetup.Show
End Sub

Private Sub mnuSetKepalaUnit_Click(Index As Integer)
    frmSetKepalaUnit.Show
End Sub

Private Sub mnuSetUserBmg_Click(Index As Integer)
    frmSetUserBmg.Show
End Sub

Private Sub mnuSetUserBo_Click(Index As Integer)
    frmSetUserBo.Show
End Sub

Private Sub mnuSetDataCuaca_Click(Index As Integer)
    frmSetDataCuaca.Show
End Sub

Private Sub mnuSetUserMenu_Click(Index As Integer)
    frmUserMenu.Show
End Sub

Private Sub mnuSetDataMetar_Click(Index As Integer)
    frmSetDataMetar.Show
End Sub

Private Sub mnuDataUser_Click(Index As Integer)
    frmDataUser.Show
End Sub

Private Sub mnuDataCuaca_Click(Index As Integer)
    frmDataCuaca.Show
End Sub

Private Sub mnuDataMetar_Click(Index As Integer)
    frmDataMetar.Show
End Sub

Private Sub mnuLaporanPeriodikDataMetar_Click(Index As Integer)
    frmLaporanPeriodikDataMetar.Show
End Sub

Private Sub mnuLaporanHarianDataMetar_Click(Index As Integer)
    frmLaporanHarianDataMetar.Show
End Sub

Private Sub mnuLaporanPeriodikDataCuaca_Click(Index As Integer)
    frmLaporanPeriodikDataCuaca.Show
End Sub

Private Sub mnuLaporanHarianDataCuaca_Click(Index As Integer)
    frmLaporanHarianDataCuaca.Show
End Sub

Private Sub mnuPegunaUnitBmgOnline_Click(Index As Integer)
```

```
frmBrowsePbm.Show
End Sub
```

```
Private Sub mnuPeggunaUnitBoOnline_Click(Index As Integer)
    frmBrowsePbo.Show
End Sub
```

```
Private Sub mnuRekamSuara_Click(Index As Integer)
    frmRecord.Show
End Sub
```

```
Private Sub mnuLogoff_Click(Index As Integer)
    menutup_menu
End Sub
```

```
Sub menutup_menu()
Dim dbUser As New ADODB.Recordset
Set dbUser = ExecSQL(GetDSN, "SELECT * FROM user WHERE nama = " & MainForm.UserName & " ")
'menutup menu
If dbUser.RecordCount > 0 Then
    If dbUser!Status = "Online" Then
        Dim upUser As New ADODB.Recordset
        Set upUser = ExecSQL(GetDSN, "UPDATE user SET status='Offline' WHERE nama=" & dbUser!nama & " ")
        End If
    End If
    Unload Me
    FrmLogin.Show vbModal
    Me.Hide
    Me.Show
End Sub
```

2. Login

```
Option Explicit
Private Sub cmdCancel_Click()
    Unload Me
End Sub
```

```
Private Sub cmdLogin_Click()
    validasi_user
End Sub
```

```
Private Sub Form_KeyDown(KeyCode As Integer, Shift As Integer)
    If KeyCode = 13 Then SendKeys "{TAB}"
End Sub
```

```
Sub validasi_user()
Dim dbUser As New ADODB.Recordset
Set dbUser = ExecSQL(GetDSN, "SELECT * FROM user WHERE userid=" & txtUserID.Text & " AND akses = 'Activated' ")
'membandingkan user id, password, dan akses dalam database
If dbUser.RecordCount > 0 Then
    If dbUser!Password = txtPassword.Text Then
        If dbUser!Akses = "Activated" Then
            If dbUser!UserID = txtUserID.Text Then
                Dim upUser As New ADODB.Recordset
                'mencari akses
otoritas user
                Set upUser = ExecSQL(GetDSN, "UPDATE user SET status='Online' WHERE userid=" & dbUser!UserID & " ")
                MainForm.UserName = dbUser!nama
                MainForm.UserID = dbUser!UserID
                MainForm.UserLevel = dbUser!level
                Unload Me

                MainForm.Show 'menampilkan record data

                MainForm.StatusBar1.Panels(1).Text = "User Name : " & dbUser!nama
                nama user berstatus online 'mencari
```

```

        MainForm.StatusBar1.Panels(7).Text = Format(Date, "dddd, dd-mm-yyyy") + " " + Format(TIME,
"hh:mm:ss ")
        MainForm.StatusBar1.Panels(8).Text = "Unit : " & dbUser!Unit
    Else
        Pesan Array("Proses Otentifikasi Gagal, Periksa Kembali User ID atau Password anda!")
'menampilkan pesan kesalahan
        txtPassword.SetFocus
    End If
    Else
        Pesan Array("Proses Otentifikasi Gagal, Periksa Kembali User ID atau Password anda!")
        txtUserID.SetFocus
    End If
    Else
        Pesan Array("Proses Otentifikasi Gagal, Periksa Kembali User ID atau Password anda!")
        txtUserID.SetFocus
    End If
    Else
        Pesan Array("Anda Telah Dideaktifasi, Harap Segera Menghubungi Kepala Unit Anda !")
        txtUserID.SetFocus
    End If
    Exit Sub
End Sub

```

3. Set Kepala Unit

Option Explicit

```

Dim conn As New ADODB.Connection
Dim rs As New ADODB.Recordset
Dim rsclone As New ADODB.Recordset

```

```

Dim cOrder As String
Dim vaKode As Variant
Dim vaEntry As Variant
Dim vaGrid As Variant
Dim VaUnit As Variant
Dim isUpdate As Boolean

```

```

Public Col As TrueOleDBGrid70.Column
Public Cols As TrueOleDBGrid70.Columns

```

```

Private Sub Form_Load() 'olah data kepala unit

```

```

    InitForm Me
    InitValue

```

```

    vaKode = Array(txtUserID, cmbUnit)
    vaEntry = Array(txtUserID, txtNama, txtPassword, txtCPassword, cmbUnit, cmbAkses, txtLevel, txtQualify)
    vaGrid = Array(datagrid)

```

```

    TabIndex Array(txtUserID, txtNama, txtPassword, txtCPassword, _
    cmbUnit, cmbAkses, txtLevel, txtQualify, cmdAdd, cmdChange, cmdOk, _
    cmdCancel, cmdExit)

```

```

    SetStatus 3, Me, vaKode, vaEntry, vaGrid, VaUnit
    cOrder = "userid"
    RefreshGrid

```

```

End Sub

```

```

Private Sub cmdAdd_Click()
    SetStatus 0, Me, vaKode, vaEntry, vaGrid, VaUnit
    isUpdate = False
End Sub

```

```

Private Sub cmdCancel_Click() 'input data kepala unit
    SetStatus 2, Me, vaKode, vaEntry, vaGrid, VaUnit
    GetList ValidNull(datagrid.Columns(0).Value)

```

```

End Sub

Private Sub cmdChange_Click() 'edit data kepala unit
    SetStatus 1, Me, vaKode, vaEntry, vaGrid, VaUnit
    isUpdate = True
End Sub

Private Sub cmdDel_Click()
    hapus_data_kepala_unit
End Sub

Private Sub cmdExit_Click()
    Unload Me
End Sub

Private Sub cmdOk_Click()
    kelola_kepala_unit
End Sub

Private Sub cmbAkses_Click()
    combo
End Sub

Private Sub cmbUnit_Click()
    combo
End Sub

Private Sub RefreshGrid()
On Error Resume Next
    Set rs = ExecSQL(GetDSN, "SELECT * FROM user WHERE qualify = 'HEAD' ORDER by " & cOrder)
    Set rsclone = rs.Clone
    DoRefreshGrid datagrid, rsclone, Array(2, 4, 2, 2, 2, 2)
End Sub

Sub hapus_data_kepala_unit() 'hapus data kepala unit
Dim del As New ADODB.Recordset
If YesNo(Array("Hapus Data ?", "User ID: " + txtUserID.Text)) Then
    Set del = ExecSQL(GetDSN, "DELETE FROM user WHERE userid=" & txtUserID.Text & "")
    RefreshGrid
    SetStatus 5, Me, vaKode, vaEntry, vaGrid, VaUnit
    isUpdate = False
End If
End Sub

Sub kelola_kepala_unit()
Dim vaField As Variant
Dim vaValue As Variant
Dim Success As Boolean

vaField = Array("userid", "nama", _
    "password", "unit", "akses", "level", "qualify")
vaValue = Array(txtUserID.Text, txtNama.Text, _
    txtPassword.Text, cmbUnit.Text, cmbAkses.Text, txtLevel.Text, txtQualify.Text)

If ValidSaving Then 'validasi data

    If YesNo(Array("Simpan Data ?", "User ID: " + txtUserID.Text)) Then
        If isUpdate Then
            Success = Update(GetDSN, "user", "userid= " & txtUserID.Text & "", vaField, vaValue)
        Else
            Success = Insert(GetDSN, "user", vaField, vaValue)
        End If
        If Not Success Then
            MsgBox "Proses penyimpanan data gagal", , "Informasi"
        End If
        RefreshGrid
        SetStatus btInit, Me, vaKode, vaEntry, vaGrid, VaUnit
        cmdAdd.SetFocus
    End If
End If

```


End Sub

Sub InitValue()

```

txtUserID.Text = "": txtUserID.MaxLength = 12
txtNama.Text = "": txtNama.MaxLength = 40
txtPassword.Text = "": txtPassword.MaxLength = 20
txtCPassword.Text = "": txtCPassword.MaxLength = 20
txtLevel.Text = "": txtLevel.MaxLength = 2
txtQualify.Text = "": txtQualify.MaxLength = 10
cmbUnit.Text = "--Choose Unit--"
cmbAkses.Text = "--Choose Access--"
txtPassword.PasswordChar = "*"
txtCPassword.PasswordChar = "*"

```

End Sub

Private Function GetList(pKode As String) 'mencari user id kepala unit

Dim rs As New ADODB.Recordset

```
Set rs = ExecSQL(GetDSN, "SELECT * FROM user WHERE userid=" & pKode & "'")
```

With rs

If .RecordCount > 0 Then

```

txtUserID.Text = ValidNull(!UserID)
txtNama.Text = ValidNull(!nama)
txtPassword.Text = ValidNull(!Password)
txtCPassword.Text = ValidNull(!Password)
cmbUnit.Text = ValidNull(!Unit)
cmbAkses.Text = ValidNull(!Akses)
txtLevel.Text = ValidNull(!level)
txtQualify.Text = ValidNull(!qualify)
On Error Resume Next

```

GetList = True

Else

InitValue

GetList = False

End If

End With

End Function

Private Function ValidSaving()

On Error Resume Next

ValidSaving = True

If Not CheckData(txtUserID.Text, "Isi Kode") Then

ValidSaving = False

txtUserID.SetFocus

Exit Function

End If

If Not CheckData(txtNama.Text, "Isi Nama User") Then

ValidSaving = False

txtNama.SetFocus

Exit Function

End If

If Not CheckData(txtLevel, "Sorot Pilihan Unit Dan Pilihan Akses !") Then

ValidSaving = False

txtLevel.SetFocus

Exit Function

End If

If txtPassword.Text <> txtCPassword.Text Then

MsgBox "Konfirmasi Password tidak sama!", , "Informasi"

txtCPassword.SetFocus

ValidSaving = False

Exit Function

End If

End Function

```

Private Sub datagrid_HeadClick(ByVal ColIndex As Integer)
    cOrder = GetGridOrder(datagrid, ColIndex, False)
    RefreshGrid
End Sub

Private Sub datagrid_FilterChange()
    GoFilter datagrid, rsclone
End Sub

Private Sub datagrid_RowColChange(LastRow As Variant, ByVal LastCol As Integer)
    If Not IsNull(datagrid.Columns(0).Value) Then
        GetList datagrid.Columns(0).Value
    End If
End Sub

Sub combo()
If cmbUnit.ListIndex = 0 Then
    If cmbAkses.ListIndex = 0 Then
        txtLevel.Text = 3
        txtQualify.Text = "HEAD"
    End If
End If
If cmbUnit.ListIndex = 0 Then
    If cmbAkses.ListIndex = 1 Then
        txtLevel.Text = 7
        txtQualify.Text = "HEAD"
    End If
End If
If cmbUnit.ListIndex = 1 Then
    If cmbAkses.ListIndex = 0 Then
        txtLevel.Text = 4
        txtQualify.Text = "HEAD"
    End If
End If
If cmbUnit.ListIndex = 1 Then
    If cmbAkses.ListIndex = 1 Then
        txtLevel.Text = 7
        txtQualify.Text = "HEAD"
    End If
End If
End Sub

Sub InitDelete()

    txtUserID.Text = "": txtUserID.MaxLength = 10
    txtNama.Text = "": txtNama.MaxLength = 20
    txtPassword.Text = "": txtPassword.MaxLength = 20
    txtCPassWord.Text = "": txtCPassWord.MaxLength = 20
    txtLevel.Text = "": txtLevel.MaxLength = 2
    txtQualify.Text = "": txtQualify.MaxLength = 10
    cmbUnit.Text = ""
    cmbAkses.Text = ""
    txtPassword.PasswordChar = "*"
    txtCPassWord.PasswordChar = "*"

End Sub

Private Sub txtUserId_KeyPress(KeyAscii As Integer)
Dim sTemplate As String

sTemplate = "!@#%&*()_+-.,<>?/\[\]{};:":""
    If InStr(1, sTemplate, Chr(KeyAscii)) > 0 Then
        KeyAscii = 0
    End If
End Sub

Private Sub txtNama_KeyPress(KeyAscii As Integer)
Dim sTemplate As String

sTemplate = "!@#%&*()_+-.=<>?/\[\]{};:":""
    If InStr(1, sTemplate, Chr(KeyAscii)) > 0 Then

```

```

    KeyAscii = 0
End If
If KeyAscii > 47 And KeyAscii < 58 Then
    KeyAscii = 0
End If
End Sub

```

```

Private Sub txtPassword_KeyPress(KeyAscii As Integer)
Dim sTemplate As String
sTemplate = "!@#$%^&*()_+-.,<>?^[|]{ };"
If InStr(1, sTemplate, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
End Sub

```

```

Private Sub txtCPassword_KeyPress(KeyAscii As Integer)
Dim sTemplate As String
sTemplate = "!@#$%^&*()_+-.,<>?^[|]{ };"
If InStr(1, sTemplate, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
End Sub

```

```

Private Sub cmbUnit_KeyPress(KeyAscii As Integer)
If KeyAscii >= 0 Then KeyAscii = 0
End Sub

```

```

Private Sub cmbAkses_KeyPress(KeyAscii As Integer)
If KeyAscii >= 0 Then KeyAscii = 0
End Sub

```

4. Set User BMG

Option Explicit

```

Dim conn As New ADODB.Connection
Dim rs As New ADODB.Recordset
Dim rclone As New ADODB.Recordset

```

```

Dim cOrder As String
Dim vaKode As Variant
Dim vaEntry As Variant
Dim vaGrid As Variant
Dim VaUnit As Variant
Dim isUpdate As Boolean

```

```

Public Col As TrueOleDbGrid70.Column
Public Cols As TrueOleDbGrid70.Columns

```

```

Private Sub Form_Load() 'olah data pengguna unit BMG

```

```

    InitForm Me
    InitValue

```

```

    vaKode = Array(txtUserID)
    VaUnit = Array(txtUnit)
    vaEntry = Array(txtUserID, txtNama, txtPassword, txtCPassword, txtUnit, cmbAkses, txtLevel, txtQualify)
    vaGrid = Array(datagrid)

```

```

    TabIndex Array(txtUserID, txtNama, txtPassword, txtCPassword, _
    txtUnit, cmbAkses, txtLevel, txtQualify, cmdAdd, cmdChange, cmdOk, _
    cmdCancel, cmdExit)

```

```

    SetStatus 3, Me, vaKode, vaEntry, vaGrid, VaUnit
    cOrder = "userid"
    RefreshGrid

```

```

End Sub

```

```

Private Sub cmdAdd_Click() 'input data pengguna unit BMG
    SetStatus 0, Me, vaKode, vaEntry, vaGrid, VaUnit
    isUpdate = False
End Sub

Private Sub cmdCancel_Click()
    SetStatus 2, Me, vaKode, vaEntry, vaGrid, VaUnit
    GetList ValidNull(datagrid.Columns(0).Value)
End Sub

Private Sub cmdChange_Click() 'edit data pengguna unit BMG
    SetStatus 1, Me, vaKode, vaEntry, vaGrid, VaUnit
    isUpdate = True
End Sub

Private Sub cmdDel_Click()
    hapus_data_pengguna_unit_BMG
End Sub

Private Sub cmdExit_Click()
    Unload Me
End Sub

Private Sub cmdOk_Click()
    kelola_data_pengguna_unit_BMG
End Sub

Private Sub cmbAkses_Click()
    combo
End Sub

Sub hapus_data_pengguna_unit_BMG() 'hapus data pengguna unit BMG
    Dim del As New ADODB.Recordset
    If YesNo(Array("Hapus Data ?", "User ID: " + txtUserID.Text)) Then
        Set del = ExecSQL(GetDSN, "DELETE FROM user WHERE userid=" & txtUserID.Text & "")
        RefreshGrid
        SetStatus 5, Me, vaKode, vaEntry, vaGrid, VaUnit
        isUpdate = False
    End If
End Sub

Sub kelola_data_pengguna_unit_BMG()
    Dim vaField As Variant
    Dim vaValue As Variant
    Dim Success As Boolean

    vaField = Array("userid", "nama", _
        "password", "unit", "akses", "level", "qualify")
    vaValue = Array(txtUserID.Text, txtNama.Text, _
        txtPassword.Text, txtUnit.Text, cmbAkses.Text, txtLevel.Text, txtQualify.Text)

    If ValidSaving Then 'validasi data
        If YesNo(Array("Simpan Data ?", "User ID: " + txtUserID.Text)) Then
            If isUpdate Then
                Success = Update(GetDSN, "user", "userid=" & txtUserID.Text & "", vaField, vaValue)
            Else
                Success = Insert(GetDSN, "user", vaField, vaValue)
            End If
            If Not Success Then
                MsgBox "Proses penyimpanan data gagal", , "Informasi"
            End If
            RefreshGrid
            SetStatus btInit, Me, vaKode, vaEntry, vaGrid, VaUnit
            cmdAdd.SetFocus
        End If
    End If
End Sub

Private Sub RefreshGrid()
    On Error Resume Next

```

```

Set rs = ExecSQL(GetDSN, "SELECT * FROM user WHERE unit = 'BMG' ORDER by " & cOrder)
Set rsclone = rs.Clone
DoRefreshGrid datagrid, rsclone, Array(2, 5, 1, 2, 2, 2)
End Sub

```

```

Sub InitValue()

```

```

    txtUserID.Text = "": txtUserID.MaxLength = 12
    txtNama.Text = "": txtNama.MaxLength = 40
    txtPassword.Text = "": txtPassword.MaxLength = 20
    txtCPassword.Text = "": txtCPassword.MaxLength = 20
    txtLevel.Text = "": txtLevel.MaxLength = 2
    txtQualify.Text = "": txtQualify.MaxLength = 10
    txtUnit.Text = "BMG"
    cmbAkses.Text = "--Choose Access--"
    txtPassword.PasswordChar = "*"
    txtCPassword.PasswordChar = "*"

```

```

End Sub

```

```

Private Function GetList(pKode As String) 'mencari user id pengguna unit BMG

```

```

    Dim rs As New ADODB.Recordset

    Set rs = ExecSQL(GetDSN, "SELECT * FROM user WHERE userid=" & pKode & "")
    With rs
        If .RecordCount > 0 Then
            txtUserID.Text = ValidNull(!UserID)
            txtNama.Text = ValidNull(!nama)
            txtPassword.Text = ValidNull(!Password)
            txtCPassword.Text = ValidNull(!Password)
            txtUnit.Text = ValidNull(!Unit)
            cmbAkses.Text = ValidNull(!Akses)
            txtLevel.Text = ValidNull(!level)
            txtQualify.Text = ValidNull(!qualify)
            On Error Resume Next

            GetList = True
        Else
            InitValue
            GetList = False
        End If
    End With
End Function

```

```

Private Function ValidSaving()

```

```

    On Error Resume Next
    ValidSaving = True
    If Not CheckData(txtUserID.Text, "Isi Kode") Then
        ValidSaving = False
        txtUserID.SetFocus
        Exit Function
    End If
    If Not CheckData(txtNama.Text, "Isi Nama User") Then
        ValidSaving = False
        txtNama.SetFocus
        Exit Function
    End If

    If Not CheckData(txtLevel, "Sorot Pilihan Akses !") Then
        ValidSaving = False
        txtLevel.SetFocus
        Exit Function
    End If

    If txtPassword.Text <> txtCPassword.Text Then
        MsgBox "Konfirmasi Password tidak sama!", , "Informasi"
        txtCPassword.SetFocus
        ValidSaving = False
        Exit Function
    End If

```

```

End Function

Private Sub datagrid_HeadClick(ByVal ColIndex As Integer)
    cOrder = GetGridOrder(datagrid, ColIndex, False)
    RefreshGrid
End Sub

Private Sub datagrid_FilterChange()
    GoFilter datagrid, rsclone
End Sub

Private Sub datagrid_RowColChange(LastRow As Variant, ByVal LastCol As Integer)
    If Not IsNull(datagrid.Columns(0).Value) Then
        GetList datagrid.Columns(0).Value
    End If
End Sub

Sub combo()
    If cmbAkses.ListIndex = 0 Then
        txtLevel.Text = 5
        txtQualify.Text = "USER"
    End If
    If cmbAkses.ListIndex = 1 Then
        txtLevel.Text = 7
        txtQualify.Text = "USER"
    End If
End Sub

Sub InitDelete()

    txtUserID.Text = "": txtUserID.MaxLength = 10
    txtNama.Text = "": txtNama.MaxLength = 20
    txtPassword.Text = "": txtPassword.MaxLength = 20
    txtCPassWord.Text = "": txtCPassWord.MaxLength = 20
    txtLevel.Text = "": txtLevel.MaxLength = 2
    txtQualify.Text = "": txtQualify.MaxLength = 10
    txtUnit.Text = ""
    cmbAkses.Text = ""
    txtPassword.PasswordChar = "*"
    txtCPassWord.PasswordChar = "*"

End Sub

Private Sub txtUserId_KeyPress(KeyAscii As Integer)
    Dim sTemplate As String

    sTemplate = "!@#%&*()_+-.,<>?/[|]{ };"
    If InStr(1, sTemplate, Chr(KeyAscii)) > 0 Then
        KeyAscii = 0
    End If
End Sub

Private Sub txtNama_KeyPress(KeyAscii As Integer)
    Dim sTemplate As String
    sTemplate = "!@#%&*()_+-.<>?/[|]{ };"
    If InStr(1, sTemplate, Chr(KeyAscii)) > 0 Then
        KeyAscii = 0
    End If
    If KeyAscii > 47 And KeyAscii < 58 Then
        KeyAscii = 0
    End If
End Sub

Private Sub txtPassword_KeyPress(KeyAscii As Integer)
    Dim sTemplate As String
    sTemplate = "!@#%&*()_+-.,<>?/[|]{ };"
    If InStr(1, sTemplate, Chr(KeyAscii)) > 0 Then
        KeyAscii = 0
    End If

```

```

End Sub

Private Sub txtCPasswOrd_KeyPress(KeyAscii As Integer)
Dim sTemplate As String
sTemplate = "!@#%&*()_+-.<>?/[|]{ }:"
If InStr(1, sTemplate, Chr(KeyAscii)) > 0 Then
KeyAscii = 0
End If
End Sub

```

```

Private Sub cmbAkses_KeyPress(KeyAscii As Integer)
If KeyAscii >= 0 Then KeyAscii = 0
End Sub

```

5. Set User BO

Option Explicit

```

Dim conn As New ADODB.Connection
Dim rs As New ADODB.Recordset
Dim rsclose As New ADODB.Recordset

```

```

Dim cOrder As String
Dim vaKode As Variant
Dim vaEntry As Variant
Dim vaGrid As Variant
Dim VaUnit As Variant
Dim isUpdate As Boolean

```

```

Public Col As TrueOleDbGrid70.Column
Public Cols As TrueOleDbGrid70.Columns

```

```

Private Sub Form_Load() 'olah data pengguna unit BO

```

```

InitForm Me
InitValue

```

```

vaKode = Array(txtUserID)
VaUnit = Array(txtUnit)
vaEntry = Array(txtUserID, txtNama, txtPassword, txtCPasswOrd, txtUnit, cmbAkses, txtLevel, txtQualify)
vaGrid = Array(datagrid)

```

```

TabIndex Array(txtUserID, txtNama, txtPassword, txtCPasswOrd, _
txtUnit, cmbAkses, txtLevel, txtQualify, cmdAdd, cmdChange, cmdOk, _
cmdCancel, cmdExit)

```

```

SetStatus 3, Me, vaKode, vaEntry, vaGrid, VaUnit
cOrder = "userid"
RefreshGrid

```

```

End Sub

```

```

Private Sub cmdAdd_Click() 'input data pengguna unit BO
SetStatus 0, Me, vaKode, vaEntry, vaGrid, VaUnit
isUpdate = False
End Sub

```

```

Private Sub cmdCancel_Click()
SetStatus 2, Me, vaKode, vaEntry, vaGrid, VaUnit
GetList ValidNull(datagrid.Columns(0).Value)
End Sub

```

```

Private Sub cmdChange_Click() 'edit data pengguna unit BO
SetStatus 1, Me, vaKode, vaEntry, vaGrid, VaUnit
isUpdate = True
End Sub

```

```

Private Sub cmdDel_Click()
hapus_data_pengguna_unit_BO

```

```

End Sub

Private Sub cmdExit_Click()
    Unload Me
End Sub

Private Sub cmdOk_Click()
    kelola_data_pengguna_unit_BO
End Sub

Private Sub cmbAkses_Click()
    combo
End Sub

Private Sub RefreshGrid()
On Error Resume Next
    Set rs = ExecSQL(GetDSN, "SELECT * FROM user WHERE unit = 'BO' ORDER by " & cOrder)
    Set rsclone = rs.Clone
    DoRefreshGrid datagrid, rsclone, Array(2, 3, 1, 2, 2, 2)
End Sub

Sub hapus_data_pengguna_unit_BO() 'hapus data pengguna unit BO
Dim del As New ADODB.Recordset
If YesNo(Array("Hapus Data ?", "User ID: " + txtUserID.Text)) Then
    Set del = ExecSQL(GetDSN, "DELETE FROM user WHERE userid=" & txtUserID.Text & "")
    RefreshGrid
    SetStatus 5, Me, vaKode, vaEntry, vaGrid, VaUnit
    isUpdate = False
End If
End Sub

Sub kelola_data_pengguna_unit_BO()
Dim vaField As Variant
Dim vaValue As Variant
Dim Success As Boolean

vaField = Array("userid", "nama", _
    "password", "unit", "akses", "level", "qualify")
vaValue = Array(txtUserID.Text, txtNama.Text, _
    txtPassword.Text, txtUnit.Text, cmbAkses.Text, txtLevel.Text, txtQualify.Text)

If ValidSaving Then 'validasi data
    If YesNo(Array("Simpan Data ?", "User ID: " + txtUserID.Text)) Then
        If isUpdate Then
            Success = Update(GetDSN, "user", "userid=" & txtUserID.Text & "", vaField, vaValue)
        Else
            Success = Insert(GetDSN, "user", vaField, vaValue)
        End If
        If Not Success Then
            MsgBox "Proses penyimpanan data gagal", , "Informasi"
        End If
        RefreshGrid
        SetStatus btInit, Me, vaKode, vaEntry, vaGrid, VaUnit
        cmdAdd.SetFocus
    End If
End If
End Sub

Sub InitValue()

    txtUserID.Text = "": txtUserID.MaxLength = 12
    txtNama.Text = "": txtNama.MaxLength = 40
    txtPassword.Text = "": txtPassword.MaxLength = 20
    txtCPassword.Text = "": txtCPassword.MaxLength = 20
    txtLevel.Text = "": txtLevel.MaxLength = 2
    txtQualify.Text = "": txtQualify.MaxLength = 10
    txtUnit.Text = "BO"
    cmbAkses.Text = "--Choose Access--"
    txtPassword.PasswordChar = "*"
    txtCPassword.PasswordChar = "*"

```


End Sub

Private Function GetList(pKode As String) 'mencari user id pengguna unit BO
Dim rs As New ADODB.Recordset

Set rs = ExecSQL(GetDSN, "SELECT * FROM user WHERE userid=" & pKode & """)

With rs

If .RecordCount > 0 Then

txtUserID.Text = ValidNull(!UserID)

txtNama.Text = ValidNull(!nama)

txtPassword.Text = ValidNull(!Password)

txtCPassord.Text = ValidNull(!Password)

txtUnit.Text = ValidNull(!Unit)

cmbAkses.Text = ValidNull(!Akses)

txtLevel.Text = ValidNull(!level)

txtQualify.Text = ValidNull(!qualify)

On Error Resume Next

GetList = True

Else

InitValue

GetList = False

End If

End With

End Function

Private Function ValidSaving()

On Error Resume Next

ValidSaving = True

If Not CheckData(txtUserID.Text, "Isi Kode") Then

ValidSaving = False

txtUserID.SetFocus

Exit Function

End If

If Not CheckData(txtNama.Text, "Isi Nama User") Then

ValidSaving = False

txtNama.SetFocus

Exit Function

End If

If Not CheckData(txtLevel, "Sorot Pilihan Akses !") Then

ValidSaving = False

txtLevel.SetFocus

Exit Function

End If

If txtPassword.Text <> txtCPassord.Text Then

MsgBox "Konfirmasi Password tidak sama!", , "Informasi"

txtCPassord.SetFocus

ValidSaving = False

Exit Function

End If

End Function

Private Sub datagrid_HeadClick(ByVal ColIndex As Integer)

cOrder = GetGridOrder(datagrid, ColIndex, False)

RefreshGrid

End Sub

Private Sub datagrid_FilterChange()

GoFilter datagrid, rsclone

End Sub

Private Sub datagrid_RowColChange(LastRow As Variant, ByVal LastCol As Integer)

If Not IsNull(datagrid.Columns(0).Value) Then

GetList datagrid.Columns(0).Value

End If

End Sub

```

Sub combo()
If cmbAkses.ListIndex = 0 Then
    txtLevel.Text = 6
    txtQualify.Text = "USER"
End If
If cmbAkses.ListIndex = 1 Then
    txtLevel.Text = 7
    txtQualify.Text = "USER"
End If
End Sub

Sub InitDelete()

    txtUserID.Text = "": txtUserID.MaxLength = 10
    txtNama.Text = "": txtNama.MaxLength = 20
    txtPassword.Text = "": txtPassword.MaxLength = 20
    txtCPassword.Text = "": txtCPassword.MaxLength = 20
    txtLevel.Text = "": txtLevel.MaxLength = 2
    txtQualify.Text = "": txtQualify.MaxLength = 10
    txtUnit.Text = ""
    cmbAkses.Text = ""
    txtPassword.PasswordChar = "*"
    txtCPassword.PasswordChar = "*"

End Sub

Private Sub txtUserId_KeyPress(KeyAscii As Integer)
Dim sTemplate As String

sTemplate = "!@#%&*()_+-.,<>?^|[]{};:":""
If InStr(1, sTemplate, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
End Sub

Private Sub txtNama_KeyPress(KeyAscii As Integer)
Dim sTemplate As String
sTemplate = "!@#%&*()_+-.<>?^|[]{};:":""
If InStr(1, sTemplate, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
If KeyAscii > 47 And KeyAscii < 58 Then
    KeyAscii = 0
End If
End Sub

Private Sub txtPassword_KeyPress(KeyAscii As Integer)
Dim sTemplate As String
sTemplate = "!@#%&*()_+-.,<>?^|[]{};:":""
If InStr(1, sTemplate, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
End Sub

Private Sub txtCPassword_KeyPress(KeyAscii As Integer)
Dim sTemplate As String
sTemplate = "!@#%&*()_+-.,<>?^|[]{};:":""
If InStr(1, sTemplate, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
End Sub

Private Sub cmbAkses_KeyPress(KeyAscii As Integer)
If KeyAscii >= 0 Then KeyAscii = 0
End Sub

```

6. User Menu

```

Dim menuname(100) As String * 50
Dim menucaption(100) As String * 50
Dim menuindeks(100) As Byte
Dim nMenu As Long

Private Sub Form_Load() 'olah data hak akses
Dim nodX As Node
Dim S As String
Dim i As Long
nMenu = 0

For Each obj In MainForm.Controls
If UCase(Mid(obj.Name, 1, 3)) = "MNU" Then
    menuname(nMenu) = obj.Name
    menucaption(nMenu) = obj.Caption
    menuindeks(nMenu) = obj.Index
    nMenu = nMenu + 1
End If
Next

For i = 0 To nMenu - 1
If Len(Trim$(Str$(menuindeks(i)))) = 1 Then
    S = Chr$(Asc(Trim$(Str$(menuindeks(i)))) + 16)
    Set nodX = TreeView1.Nodes.Add(, S, menucaption(i))
    nodX.Checked = True
    nodX.Expanded = True
Else
    S = Chr$(Asc(Trim$(Str$(menuindeks(i)))) + 16)
    s2 = S & Mid$(Trim$(Str$(menuindeks(i))), 2, 1)
    Set nodX = TreeView1.Nodes.Add(S, tvwChild, s2, menucaption(i))
    nodX.Checked = True
End If
Next
nodX.EnsureVisible

TreeView1.BorderStyle = vbFixedSingle
End Sub

Private Sub cmdReload_Click()
    ambil_data_level_user
End Sub

Private Sub cmdSimpan_Click()
    kelola_level_user
End Sub

Private Sub cmdExit_Click()
    Unload Me
End Sub

Sub kelola_level_user() 'mengatur hak akses user
Dim rs As New ADODB.Recordset
Dim vaField As Variant
Dim vaValue As Variant

If txtLevel.Text = "1" Then
    MsgBox "Level 1 untuk supervisor!", vbOKOnly, "Informasi!"
    txtLevel.SetFocus
    Exit Sub
End If

vaField = Array("level", "nourut", "menuname", "visible")
Set rs = ExecSQL(GetDSN, "DELETE FROM menu WHERE level='" & txtLevel.Text & "'")
Dim i As Long

```

```

For i = 0 To nMenu - 1
    If TreeView1.Nodes(i + 1).Checked Then
        vaValue = Array(txtLevel.Text, i, menuname(i), "1")
    Else
        vaValue = Array(txtLevel.Text, i, menuname(i), "0")
    End If
    Success = Insert(GetDSN, "menu", vaField, vaValue)
Next
End Sub

```

```

Sub ambil_data_level_user() 'mengoreksi hak akses user
Dim rs As New ADODB.Recordset
Dim i As Long

```

```

If txtLevel.Text = "1" Then
    For i = 1 To nMenu
        TreeView1.Nodes(i).Checked = True
    Next i
    Exit Sub
End If

```

```

Set rs = ExecSQL(GetDSN, "SELECT * FROM menu WHERE level = '" & txtLevel.Text & "' ORDER BY
nourut")

```

```

If rs.RecordCount > 0 Then
    For i = 1 To rs.RecordCount
        If rs!Visible = "0" Then
            TreeView1.Nodes(i).Checked = False
        Else
            TreeView1.Nodes(i).Checked = True
        End If
        rs.MoveNext
    Next i
End If
End Sub

```

7. Set Data Cuaca

Option Explicit

```

Dim conn As New ADODB.Connection
Dim rs As New ADODB.Recordset
Dim rclone As New ADODB.Recordset

```

```

Dim cOrder As String
Dim vaKode As Variant
Dim vaEntry As Variant
Dim vaGrid As Variant
Dim VaUnit As Variant

```

```

Public Col As TrueOleDBGrid70.Column
Public Cols As TrueOleDBGrid70.Columns

```

```

Dim IP As String
Dim S As Integer
Dim m As Integer

```

```

Private Sub Form_Load() 'olah data cuaca
    InitForm Me
    InitValue
    GetUser

```

```

    vaKode = Array(txtDate, txtTime, txtDirection, txtSpeed, txtVisibility, cmbWeather, cmbClouds, txtKeterangan)
    vaEntry = Array(txtDate, txtTime, txtDirection, txtSpeed, txtVisibility, cmbWeather, cmbClouds, txtClouds,
txtWeather, txtClouds, txtFeet, txtTemperature, txtDew, txtQnh, txtKeterangan, txtData, txtActive)
    VaUnit = Array(txtDate, txtTime, txtKeterangan, txtData, txtActive)
    vaGrid = Array(datagrid)

```

```

TabIndex Array(txtDate, txtTime, txtDirection, txtSpeed, _
txtVisibility, cmbWeather, cmbClouds, txtWeather, txtFeet, txtTemperature, txtDew, txtQnh,
txtKeterangan, txtData, txtActive, cmdOk, _
cmdCancel, cmdExit)

setStatus 6, Me, vaKode, vaEntry, vaGrid, VaUnit
cOrder = "date"
RefreshGrid

StatusBar1.Panels(1).Text = "HOST MODE"
cmdSendText.Enabled = False
txtSendText.Enabled = False
FrameCounter.Visible = False
txtNick.Enabled = False
txtIP.Enabled = False
txtPort.Enabled = False
StatusBar1.Panels(2).Text = "TIDAK TERKONEKSI"
IP = Winsock1.LocalIP
txtIP.Text = IP
End Sub

Private Sub cmdAdd_Click() 'input data cuaca
    tambah_data
End Sub

Private Sub cmdOk_Click()
    simpan_data_cuaca
End Sub

Private Sub cmdCancel_Click()
    setStatus 7, Me, vaKode, vaEntry, vaGrid, VaUnit
    GetList ValidNull(datagrid.Columns(0).Value)
End Sub

Private Sub cmdExit_Click()
    keluar
End Sub

Private Sub cmdConnect_Click()
    koneksi
End Sub

Private Sub cmdSendText_Click()
    kirim_pesan
End Sub

Private Sub cmbClouds_Click()
    clouds
End Sub

Private Sub cmbWeather_Click()
    weather
End Sub

Sub tambah_data()
Dim ubData As New ADODB.Recordset
Set ubData = ExecSQL(GetDSN, "SELECT * FROM cuaca WHERE data = 'NEW'")
If ubData.RecordCount > 0 Then
    If ubData!Data = "NEW" Then
        Dim upData As New ADODB.Recordset
        Set upData = ExecSQL(GetDSN, "UPDATE cuaca SET data = 'OLD' WHERE cuacaid='" & ubData!CuacaId &
        """)
    End If
End If
setStatus 0, Me, vaKode, vaEntry, vaGrid, VaUnit
End Sub

Sub simpan_data_cuaca()
Dim vaField As Variant
Dim vaValue As Variant

```

Dim Success As Boolean

```

vaField = Array("date", "time", _
               "direction", "speed", "visibility", "weather", "clouds", "feet", "temperature", "dewpoint", "qnh", "userid",
               "data", "active")
vaValue = Array(Format(txtDate.Text, "dd-mm-yyyy"), Format(TIME, "hh:mm"), _
               txtDirection.Text, txtSpeed.Text, txtVisibility.Text, cmbWeather.Text, cmbClouds.Text, txtFeet.Text,
               txtTemperature.Text, txtDew.Text, txtQnh.Text, txtKeterangan.Text, txtData.Text, txtActive.Text)

```

If ValidSaving Then **'validasi data**

```

If YesNo(Array("Simpan Data ?", "Date: " + txtDate.Text, "Time: " + txtTime.Text, "Direction: " +
txtDirection.Text, "Speed: " + txtSpeed.Text, "Visibility: " + txtVisibility.Text, "Weather: " + cmbWeather.Text,
"Clouds: " + cmbClouds.Text, "Feet: " + txtFeet.Text, "Temperature: " + txtTemperature.Text, "DEW Point: " +
txtDew.Text, "QNH: " + txtQnh.Text)) Then

```

```

'If isUpdate Then

```

```

'Success = Update(GetDSN, "cuaca", "date= " & txtDate.Text & "", vaField, vaValue)

```

```

'Else

```

```

Success = Insert(GetDSN, "cuaca", vaField, vaValue)

```

```

End If

```

```

If Not Success Then

```

```

MsgBox "Proses penyimpanan data gagal", , "Informasi"

```

```

End If

```

```

RefreshGrid

```

```

setStatus 8, Me, vaKode, vaEntry, vaGrid, VaUnit

```

```

cmdAdd.SetFocus

```

```

End If

```

End Sub

Sub keluar()

```

Dim ubActive As New ADODB.Recordset

```

```

Set ubActive = ExecSQL(GetDSN, "SELECT * FROM cuaca WHERE active = '1'")

```

```

If ubActive.RecordCount > 0 Then

```

```

If ubActive!Active = "1" Then

```

```

Dim upActive As New ADODB.Recordset

```

```

Set upActive = ExecSQL(GetDSN, "UPDATE cuaca SET Active = '0' WHERE cuacaId=" & ubActive!CuacaId &
& """)

```

```

End If

```

```

End If

```

```

Dim clData As New ADODB.Recordset

```

```

Set clData = ExecSQL(GetDSN, "SELECT * FROM cuaca WHERE data = 'NEW'")

```

```

If clData.RecordCount > 0 Then

```

```

If clData!Data = "NEW" Then

```

```

Dim tuData As New ADODB.Recordset

```

```

Set tuData = ExecSQL(GetDSN, "UPDATE cuaca SET data = 'OLD' WHERE cuacaId=" & clData!CuacaId &
""")

```

```

End If

```

```

End If

```

Unload Me

End Sub

Sub koneksi()

```

If cmdConnect.Caption = "Connect" Or cmdConnect.Caption = "Listen" Then 'mendeteksi status komunikasi unit
Call Online

```

```

Else

```

```

Call Offline

```

```

If optHostGuest(0).Value = True Then

```

```

cmdConnect.Caption = "Listen"

```

```

Call Listening

```

```

Else

```

```

cmdConnect.Caption = "Connect"

```

```

End If

```

```

Call Offline

```

```

Exit Sub

```

```

End If

```

```

Select Case optHostGuest(0).Value

```

```

Case True:

```

```

Call Listening

```

```

Case False:

```

```

Call Connecting

```

```

    End Select
End Sub

Sub kirim_pesan()
    Winsock1.SendData txtNick.Text & ": " & txtSendText.Text
    txtIncomingData.Text = txtIncomingData.Text + vbCrLf + txtNick.Text + ": " + txtSendText.Text
    txtSendText.Text = ""
End Sub

Private Sub RefreshGrid()
On Error Resume Next
    Set rs = ExecSQL(GetDSN, "SELECT * FROM cuaca WHERE active = '1' ORDER by " & cOrder)
    Set rsclone = rs.Clone
    DoRefreshGrid datagrid, rsclone, Array(7, 5, 6, 5, 5, 5, 5, 8, 5, 5, 15)
End Sub

Sub InitValue()
    txtDirection.Text = "": txtDirection.MaxLength = 3
    txtSpeed.Text = "": txtSpeed.MaxLength = 2
    txtVisibility.Text = "": txtVisibility.MaxLength = 5
    cmbWeather.Text = "--Choose Weather--"
    cmbClouds.Text = "--Choose Clouds--"
    txtFeet.Text = "": txtFeet.MaxLength = 6
    txtTemperature.Text = "": txtTemperature.MaxLength = 2
    txtDew.Text = "": txtDew.MaxLength = 2
    txtQnh.Text = "": txtQnh.MaxLength = 4
    txtWeather.Text = "": txtWeather.MaxLength = 2
    txtClouds.Text = "": txtClouds.MaxLength = 2
End Sub

Private Function GetList(pDate As String)
    Dim rs As New ADODB.Recordset

    Set rs = ExecSQL(GetDSN, "SELECT * FROM cuaca WHERE date=" & pDate & """)
    With rs
        If .RecordCount > 0 Then
            txtDate.Text = ValidNull(!Date)
            txtTime.Text = ValidNull(!TIME)
            txtDirection.Text = ValidNull(!Direction)
            txtSpeed.Text = ValidNull(!SPEED)
            txtVisibility.Text = ValidNull(!VISIBILITY)
            cmbWeather.Text = ValidNull(!weather)
            cmbClouds.Text = ValidNull(!clouds)
            txtFeet.Text = ValidNull(!FEET)
            txtTemperature.Text = ValidNull(!TEMPERATURE)
            txtDew.Text = ValidNull(!dewpoint)
            txtQnh.Text = ValidNull(!QNH)
            txtData.Text = ValidNull(!Data)
            txtActive.Text = ValidNull(!Active)
            txtKeterangan.Text = ValidNull(!UserID)
            On Error Resume Next

            GetList = True
        Else
            InitValue
            GetList = False
        End If
    End With
End Function

Private Function ValidSaving()
On Error Resume Next
    ValidSaving = True
    If Not CheckData(txtDate, "Isi Field Date") Then
        ValidSaving = False
        txtDate.SetFocus
        Exit Function
    End If
    If Not CheckData(txtTime, "Isi Field Time") Then
        ValidSaving = False
    End If
End Function

```

```

        txtTime.SetFocus
    Exit Function
End If
If Not CheckData(txtDirection.Text, "Isi Field Direction") Then
    ValidSaving = False
    txtDirection.SetFocus
    Exit Function
End If
If Not CheckData(txtSpeed.Text, "Isi Field Speed") Then
    ValidSaving = False
    txtSpeed.SetFocus
    Exit Function
End If
If Not CheckData(txtVisibility, "Isi Field Visibility") Then
    ValidSaving = False
    txtVisibility.SetFocus
    Exit Function
End If
If Not CheckData(txtWeather, "Sorot Pilihan Weather") Then
    ValidSaving = False
    cmbWeather.SetFocus
    Exit Function
End If
If Not CheckData(txtClouds, "Sorot Pilihan Clouds") Then
    ValidSaving = False
    cmbClouds.SetFocus
    Exit Function
End If
If Not CheckData(txtFeet, "Isi Field Feet") Then
    ValidSaving = False
    txtFeet.SetFocus
    Exit Function
End If
If Not CheckData(txtTemperature, "Isi Field Temperature") Then
    ValidSaving = False
    txtTemperature.SetFocus
    Exit Function
End If
If Not CheckData(txtDew, "Isi Field DEW Point") Then
    ValidSaving = False
    txtDew.SetFocus
    Exit Function
End If
If Not CheckData(txtQnh, "Isi Field QNH") Then
    ValidSaving = False
    txtQnh.SetFocus
    Exit Function
End If
End Function

Private Sub datagrid_HeadClick(ByVal ColIndex As Integer)
    cOrder = GetGridOrder(datagrid, ColIndex, False)
    RefreshGrid
End Sub

Private Sub datagrid_FilterChange()
    GoFilter datagrid, rsclone
End Sub

Private Sub datagrid_RowColChange>LastRow As Variant, ByVal LastCol As Integer)
    If Not IsNull(datagrid.Columns(0).Value) Then
        GetList datagrid.Columns(0).Value
    End If
End Sub

Private Sub Tmr4Clock_Timer()
    txtDate.Text = Date
    txtTime.Text = Format(TIME, "hh:mm")
End Sub

```



```

Sub weather()
  If cmbWeather.ListIndex = 0 Then
    txtWeather.Text = "OK"
  End If
  If cmbWeather.ListIndex = 1 Then
    txtWeather.Text = "OK"
  End If
  If cmbWeather.ListIndex = 2 Then
    txtWeather.Text = "OK"
  End If
  If cmbWeather.ListIndex = 3 Then
    txtWeather.Text = "OK"
  End If
End Sub

Sub clouds()
  If cmbClouds.ListIndex = 0 Then
    txtClouds.Text = "OK"
  End If
  If cmbClouds.ListIndex = 1 Then
    txtClouds.Text = "OK"
  End If
  If cmbClouds.ListIndex = 2 Then
    txtClouds.Text = "OK"
  End If
  If cmbClouds.ListIndex = 3 Then
    txtClouds.Text = "OK"
  End If
  If cmbClouds.ListIndex = 4 Then
    txtClouds.Text = "OK"
  End If
End Sub

Private Sub txtDirection_KeyPress(KeyAscii As Integer)
  Dim sTemplate As String
  Dim sChar As String
  Dim sBig As String
  sTemplate = "!@#%&*()_+.,=<>?/[[]{};:;'"
  If InStr(1, sTemplate, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
  End If
  sChar = "abcdefghijklmnopqrstuvwxyz"
  If InStr(1, sChar, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
  End If
  sBig = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
  If InStr(1, sBig, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
  End If
End Sub

Private Sub txtSpeed_KeyPress(KeyAscii As Integer)
  Dim sTemplate As String
  Dim sChar As String
  Dim sBig As String
  sTemplate = "!@#%&*()_+.,=<>?/[[]{};:;'"
  If InStr(1, sTemplate, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
  End If
  sChar = "abcdefghijklmnopqrstuvwxyz"
  If InStr(1, sChar, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
  End If
  sBig = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
  If InStr(1, sBig, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
  End If
End Sub

Private Sub txtVisibility_KeyPress(KeyAscii As Integer)

```

```

Dim sTemplate As String
Dim sChar As String
Dim sBig As String
sTemplate = "!@#%&*( )_+.,=<>?/[ ]{ } ; : ; """"
If InStr(1, sTemplate, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
sChar = "abcdefghijklmnopqrstuvwxy"
If InStr(1, sChar, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
sBig = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
If InStr(1, sBig, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
End Sub

Private Sub txtFeet_KeyPress(KeyAscii As Integer)
    Dim sTemplate As String
    Dim sChar As String
    Dim sBig As String
    sTemplate = "!@#%&*( )_+.,=<>?/[ ]{ } ; : ; """"
    If InStr(1, sTemplate, Chr(KeyAscii)) > 0 Then
        KeyAscii = 0
    End If
    sChar = "abcdefghijklmnopqrstuvwxy"
    If InStr(1, sChar, Chr(KeyAscii)) > 0 Then
        KeyAscii = 0
    End If
    sBig = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
    If InStr(1, sBig, Chr(KeyAscii)) > 0 Then
        KeyAscii = 0
    End If
End Sub

Private Sub txtTemperature_KeyPress(KeyAscii As Integer)
    Dim sTemplate As String
    Dim sChar As String
    Dim sBig As String
    sTemplate = "!@#%&*( )_+.,=<>?/[ ]{ } ; : ; """"
    If InStr(1, sTemplate, Chr(KeyAscii)) > 0 Then
        KeyAscii = 0
    End If
    sChar = "abcdefghijklmnopqrstuvwxy"
    If InStr(1, sChar, Chr(KeyAscii)) > 0 Then
        KeyAscii = 0
    End If
    sBig = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
    If InStr(1, sBig, Chr(KeyAscii)) > 0 Then
        KeyAscii = 0
    End If
End Sub

Private Sub txtDew_KeyPress(KeyAscii As Integer)
    Dim sTemplate As String
    Dim sChar As String
    Dim sBig As String
    sTemplate = "!@#%&*( )_+.,=<>?/[ ]{ } ; : ; """"
    If InStr(1, sTemplate, Chr(KeyAscii)) > 0 Then
        KeyAscii = 0
    End If
    sChar = "abcdefghijklmnopqrstuvwxy"
    If InStr(1, sChar, Chr(KeyAscii)) > 0 Then
        KeyAscii = 0
    End If
    sBig = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
    If InStr(1, sBig, Chr(KeyAscii)) > 0 Then
        KeyAscii = 0
    End If
End Sub

```

```

Private Sub txtQnh_KeyPress(KeyAscii As Integer)
    Dim sTemplate As String
    Dim sChar As String
    Dim sBig As String
    sTemplate = "!@#%$%^&*()_+.,=<>?/[|]{ } ; : ; """"
    If InStr(1, sTemplate, Chr(KeyAscii)) > 0 Then
        KeyAscii = 0
    End If
    sChar = "abcdefghijklmnopqrstuvwxy"
    If InStr(1, sChar, Chr(KeyAscii)) > 0 Then
        KeyAscii = 0
    End If
    sBig = "ABCDEFGHIJKLMNPOQRSTUVWXYZ"
    If InStr(1, sBig, Chr(KeyAscii)) > 0 Then
        KeyAscii = 0
    End If
End Sub

Private Sub cmbWeather_KeyPress(KeyAscii As Integer)
    If KeyAscii >= 0 Then KeyAscii = 0
End Sub

Private Sub cmbClouds_KeyPress(KeyAscii As Integer)
    If KeyAscii >= 0 Then KeyAscii = 0
End Sub

Sub GetUser()
    Dim rs As New ADODB.Recordset
    Set rs = ExecSQL(GetDSN, "SELECT * FROM user WHERE unit = 'BMG' AND status = 'Online'")
    If rs.RecordCount > 0 Then
        txtKeterangan.Text = rs!nama
    Else
        InitValue
    End If
End Sub

Public Sub Offline()
    imgOffline.Visible = True
    imgOnline.Visible = False
    imgConnecting.Visible = False
    txtPort.Enabled = True
    txtNick.Enabled = True
    txtIP.Enabled = True
    txtNick.Enabled = False
    cmdSendText.Enabled = False
    txtSendText.Enabled = False
    optHostGuest(0).Enabled = True
    optHostGuest(1).Enabled = True
    StatusBar1.Panels(2).Text = "TIDAK TERKONEKSI"
    Winsock1.Close
    tmrClock.Enabled = False
    S = 0
    m = 0
    FrameCounter.Visible = False
    lblSeconds.Caption = ""
    lblMinutes.Caption = ""
End Sub

Public Sub Online()
    imgOnline.Visible = True
    imgOffline.Visible = False
    imgConnecting.Visible = False
    txtIP.Enabled = False
    txtPort.Enabled = False
    txtNick.Enabled = False
    cmdSendText.Enabled = True
    txtSendText.Enabled = True
    txtIncomingData.Text = ""
    optHostGuest(0).Enabled = False

```

```

    optHostGuest(1).Enabled = False
    cmdConnect.Caption = "Disconnect"
End Sub

Public Sub Connecting()
    imgConnecting.Visible = True
    imgOffline.Visible = False
    imgOnline.Visible = False
    cmdSendText.Enabled = False
    txtSendText.Enabled = False
    StatusBar1.Panels(2).Text = "CONTACTING HOST..."
    If Winsock1.State <> sckClosed Then
        Winsock1.Close
    End If
    IP = txtIP.Text
    If LCase$(IP) = "localhost" Then IP = Winsock1.LocalIP
    Winsock1.Connect txtIP.Text, txtPort.Text
End Sub

Public Sub Listening() 'menampilkan pesan status koneksi unit
    imgConnecting.Visible = True
    imgOffline.Visible = False
    imgOnline.Visible = False
    cmdSendText.Enabled = False
    txtSendText.Enabled = False
    Winsock1.Close
    Winsock1.LocalPort = txtPort.Text
    Winsock1.Listen
    StatusBar1.Panels(2).Text = "LISTENING ON PORT: " & txtPort.Text
End Sub

Private Sub optHostGuest_Click(Index As Integer)
    Select Case Index
        Case 0: '
            IP = Winsock1.LocalIP
            txtIP.Text = IP
            txtIP.Locked = True
            txtIP.Enabled = False
            cmdConnect.Caption = "Listen"
            StatusBar1.Panels(1).Text = "HOST MODE"
        Case 1:
            txtIP.Text = "localhost"
            cmdConnect.Caption = "Connect"
            txtIP.Locked = False
            txtIP.Enabled = True
            StatusBar1.Panels(1).Text = "GUEST MODE"
    End Select
End Sub

Private Sub tmrClock_Timer()
    S = S + 1
    If S = 60 Then
        S = 0
        m = m + 1
    End If
    If m > 0 Then
        lblMinutes.Caption = m
    End If
    lblSeconds.Caption = S
    DoEvents
End Sub

Private Sub txtIncomingData_Change()
    txtIncomingData.SelStart = Len(txtIncomingData)
End Sub

Private Sub txtSendText_KeyPress(KeyAscii As Integer) 'mengirim pesan komunikasi
    If KeyAscii = 13 Then
        cmdSendText_Click
        KeyAscii = 0
    End If
End Sub

```

```

End If
End Sub

Private Sub Winsock1_Close()
    Select Case optHostGuest(0).Value
        Case True:
            cmdConnect_Click
            cmdConnect_Click
            MsgBox "Connection terminated by Guest. Server has been reset and awaiting a new client..."
        Case False:
            MsgBox "Connection terminated by Host..."
            cmdConnect_Click
    End Select
End Sub

Private Sub Winsock1_Connect()
    Select Case optHostGuest(0).Value
        Case True:
        Case False:
            Call Online
            StatusBar1.Panels(2).Text = "CONNECTED TO HOST"
    End Select
    FrameCounter.Visible = True
    tmrClock.Enabled = True
End Sub

Private Sub Winsock1_ConnectionRequest(ByVal requestID As Long)
    If Winsock1.State <> sckClosed Then
        Winsock1.Close
        Winsock1.LocalPort = txtPort.Text
        Winsock1.Accept requestID
        Online
        StatusBar1.Panels(2).Text = "CLIENT CONNECTED"
        FrameCounter.Visible = True
        tmrClock.Enabled = True
    End If
End Sub

Private Sub Winsock1_DataArrival(ByVal bytesTotal As Long) 'membuka jalur komunikasi
    Dim Data As String
    Winsock1.GetData Data
    txtIncomingData.Text = txtIncomingData.Text + vbCrLf & Data
End Sub

Private Sub Winsock1_Error(ByVal Number As Integer, Description As String, ByVal Scode As Long, ByVal Source As String, ByVal HelpFile As String, ByVal HelpContext As Long, CancelDisplay As Boolean)
    StatusBar1.Panels(2).Text = "WINSOCK ERROR: " & Err
    cmdConnect_Click
    MsgBox "UNABLE TO CONNECT..."
End Sub

```

8. Set Data METAR

```

Option Explicit

Dim conn As New ADODB.Connection
Dim rs As New ADODB.Recordset
Dim rsclose As New ADODB.Recordset

Dim cOrder As String
Dim vaKode As Variant
Dim vaEntry As Variant
Dim vaGrid As Variant
Dim VaUnit As Variant

Public Col As TrueOLEDBGrid70.Column
Public Cols As TrueOLEDBGrid70.Columns

Dim IP As String

```

```

Dim S As Integer
Dim m As Integer

Dim WithEvents baca As SpeechLib.SpVoice
Dim bacavoice As SpeechLib.ISpeechObjectToken
Dim z, n As Long
Dim RecdTime As Boolean
Dim strVoice As String
Dim settings As String
Private m_cancel As Boolean

Private Sub Form_Load() 'olah data METAR

    InitForm Me
    InitValue
    GetUser

    vaKode = Array(txtDate, txtTime, txtDirection, txtSpeed, txtVisibility, txtWeather, txtClouds, txtFeet,
txtTemperature, txtDew, txtQnh, txtKeterangan, cmbMetar, txtMetar, txtRunway1, txtRunway2, Check1, Check2)
    vaEntry = Array(txtKeterangan, cmbMetar, txtMetar)
    VaUnit = Array(txtDate, txtTime, txtDirection, txtSpeed, txtVisibility, txtWeather, txtClouds, txtFeet, txtClouds,
txtTemperature, txtDew, txtQnh, txtKeterangan, txtRunway1, txtRunway2)
    vaGrid = Array(datagrid)

    TabIndex Array(txtDate, txtTime, txtDirection, txtSpeed, _
txtVisibility, txtWeather, txtClouds, txtFeet, txtTemperature, txtDew, txtQnh, txtKeterangan, cmbMetar, txtMetar,
cmdOk, _
cmdCancel, cmdExit)

    SetStatus 9, Me, vaKode, vaEntry, vaGrid, VaUnit
    cOrder = "date"
    RefreshGrid

    StatusBar1.Panels(1).Text = "GUEST MODE"
    cmdSendText.Enabled = False
    txtSendText.Enabled = False
    FrameCounter.Visible = False
    txtNick.Enabled = False
    StatusBar1.Panels(2).Text = "TIDAK TERKONEKSI"

    cmdActivated.Enabled = False
    cmdSpeak.Enabled = False
    cmdStop.Enabled = False
    shpWait.Visible = False
    shpGo.Visible = False
    shpError.Visible = False
    Label18.Visible = False
    Label19.Visible = False
    Label21.Visible = False

    Set baca = New SpeechLib.SpVoice
    For Each bacavoice In baca.GetVoices
        strVoice = bacavoice.GetDescription
        LstVoice.AddItem strVoice
    Next

    LstVoice.Text = LstVoice.List(0)
    Slider1.Value = 14
    Slider2.Value = 20

End Sub

Private Sub cmdAdd_Click() 'input data METAR
    SetStatus 0, Me, vaKode, vaEntry, vaGrid, VaUnit
End Sub

Private Sub cmdRefresh_Click()
    RefreshGrid
End Sub

```

```

Private Sub cmdOk_Click()
    simpan_data_METAR
End Sub

Private Sub cmbClouds_Click()
    clouds
End Sub

Private Sub cmbWeather_Click()
    weather
End Sub

Private Sub cmdCancel_Click()
    SetStatus 10, Me, vaKode, vaEntry, vaGrid, VaUnit
    GetList ValidNull(datagrid.Columns(0).Value)
End Sub

Private Sub cmdActivated_Click()
    aktifasi_ATIS
End Sub

Private Sub cmdSpeak_Click()
    aktifasi_suara_ATIS
End Sub

Private Sub cmdStop_Click()
    stop_suara_ATIS
End Sub

Private Sub cmdConnect_Click()
    koneksi
End Sub

Private Sub cmdSendText_Click()
    kirim_pesan
End Sub

Private Sub cmdExit_Click()
    Unload Me
End Sub

Private Sub RefreshGrid()
    On Error Resume Next
    Set rs = ExecSQL(GetDSN, "SELECT * FROM cuaca WHERE data = 'NEW' ORDER by " & cOrder)
    Set rsclone = rs.Clone
    DoRefreshGrid datagrid, rsclone, Array(7, 5, 6)
End Sub

Sub simpan_data_METAR()
    Dim vaField As Variant
    Dim vaValue As Variant
    Dim Success As Boolean

    vaField = Array("date", "time", _
        "metar", "userid")
    vaValue = Array(Format(txtDate.Text, "dd-mm-yyyy"), Format(txtTime.Text, "HH:mm"), _
        cmbMetar.Text, txtKeterangan.Text)

    If ValidSaving Then 'validasi data

        If YesNo(Array("Simpan Data ?", "Date: " + txtDate.Text, "Time: " + txtTime.Text, "METAR: " +
            cmbMetar.Text)) Then
            Success = Insert(GetDSN, "metar", vaField, vaValue)
            cmdActivated.Enabled = True
            cmdSpeak.Enabled = False
            cmdStop.Enabled = False
            shpWait.Visible = True
            shpGo.Visible = False
            shpError.Visible = False
        End If
    End If
End Sub

```

```

        Label18.Visible = False
        Label19.Visible = True
        Label21.Visible = False
    End If
    If Not Success Then
        MsgBox "Proses penyimpanan data gagal", , "Informasi"
    End If
    RefreshGrid
    SetStatus 11, Me, vaKode, vaEntry, vaGrid, VaUnit
    cmdAdd.SetFocus
    End If
cmdExit.Enabled = False
End Sub

Sub aktifasi_ATIS()
    cmdSpeak.Enabled = True
    cmdStop.Enabled = False
    shpWait.Visible = False
    shpGo.Visible = True
    shpError.Visible = False
    Label18.Visible = True
    Label19.Visible = False
    Label21.Visible = False
    txtMetar2.Text = txtMetar3.Text
    ATIS
End Sub

Sub aktifasi_suara_ATIS()      'aktifasi suara
Dim i As Long
    cmdActivated.Enabled = False
    cmdStop.Enabled = True
    shpWait.Visible = False
    shpGo.Visible = True
    shpError.Visible = False
    Label18.Visible = True
    Label19.Visible = False
    Label21.Visible = False

    On Error Resume Next
    Set baca = New SpeechLib.SpVoice
    Set baca.Voice = baca.GetVoices().Item(LstVoice.ListIndex)
    baca.Rate = Slider1.Value - 10
    baca.Volume = 5 * Slider2.Value

    m_cancel = False
    For i = 1 To 20
        z = baca.Speak(" Radin Inten 2 airport information " + txtMetar2, SVSFlagsAsync)
        TIME
        WIND
        SPEED
        VISIBILITY
        weather
        clouds
        FEET
        TEMPERATURE
        DEW
        QNH
        z = baca.Speak(" Runway " + " One Four ", SVSFlagsAsync)
        z = baca.Speak(" Runway " + " Three Two ", SVSFlagsAsync)
        z = baca.Speak(" End Of Information " + txtMetar2, SVSFlagsAsync)
        z = i
        DoEvents
        If m_cancel Then Exit For
    Next
End Sub

Sub stop_suara_ATIS()
    cmdActivated.Enabled = False
    cmdSpeak.Enabled = False
    shpWait.Visible = False

```



```

shpGo.Visible = False
shpError.Visible = True
Label18.Visible = False
Label19.Visible = False
Label21.Visible = True
m_cancel = True
baca.Speak vbNullString, SVSFPurgeBeforeSpeak
Check1.Value = "0"
Check2.Value = "0"
cmdExit.Enabled = True
End Sub

Sub koneksi()
If cmdConnect.Caption = "Connect" Or cmdConnect.Caption = "Listen" Then
Call Online
Else
Call Offline
If optHostGuest(0).Value = True Then
cmdConnect.Caption = "Listen"
Call Listening
Else
cmdConnect.Caption = "Connect"
End If
Call Offline
Exit Sub
End If
Select Case optHostGuest(0).Value
Case True:
Call Listening
Case False:
Call Connecting
End Select
End Sub

Sub kirim_pesan()
Winsock1.SendData txtNick.Text & ": " & txtSendText.Text
txtIncomingData.Text = txtIncomingData.Text + vbCrLf + txtNick.Text + ": " + txtSendText.Text
txtSendText.Text = ""
End Sub

Sub InitValue()
cmbMetar.Text = "--Choose METAR--"
End Sub

Private Function GetList(pDate As String)
Dim rs As New ADODB.Recordset

Set rs = ExecSQL(GetDSN, "SELECT * FROM cuaca WHERE date=" & pDate & """)
With rs
If .RecordCount > 0 Then
txtDate.Text = ValidNull(!Date)
txtTime.Text = ValidNull(Format(!TIME, "hh:mm"))
txtDirection.Text = ValidNull(!Direction)
txtSpeed.Text = ValidNull(!SPEED)
txtVisibility.Text = ValidNull(!VISIBILITY)
txtWeather.Text = ValidNull(!weather)
txtClouds.Text = ValidNull(!clouds)
txtFeet.Text = ValidNull(!FEET)
txtTemperature.Text = ValidNull(!TEMPERATURE)
txtDew.Text = ValidNull(!dewpoint)
txtQnh.Text = ValidNull(!QNH)
On Error Resume Next
GetList = True
Else
InitValue
GetList = False
End If
End With
End Function

```

```
Private Sub cmbMetar_Click()  
    If cmbMetar.ListIndex = 0 Then  
        txtMetar3.Text = "Alfa"  
    End If  
    If cmbMetar.ListIndex = 1 Then  
        txtMetar3.Text = "Bravo"  
    End If  
    If cmbMetar.ListIndex = 2 Then  
        txtMetar3.Text = "Charlie"  
    End If  
    If cmbMetar.ListIndex = 3 Then  
        txtMetar3.Text = "Delta"  
    End If  
    If cmbMetar.ListIndex = 4 Then  
        txtMetar3.Text = "Echo"  
    End If  
    If cmbMetar.ListIndex = 5 Then  
        txtMetar3.Text = "Foxtrot"  
    End If  
    If cmbMetar.ListIndex = 6 Then  
        txtMetar3.Text = "Golf"  
    End If  
    If cmbMetar.ListIndex = 7 Then  
        txtMetar3.Text = "Hotel"  
    End If  
    If cmbMetar.ListIndex = 8 Then  
        txtMetar3.Text = "India"  
    End If  
    If cmbMetar.ListIndex = 9 Then  
        txtMetar3.Text = "Juliett"  
    End If  
    If cmbMetar.ListIndex = 10 Then  
        txtMetar3.Text = "Kilo"  
    End If  
    If cmbMetar.ListIndex = 11 Then  
        txtMetar3.Text = "Lima"  
    End If  
    If cmbMetar.ListIndex = 12 Then  
        txtMetar3.Text = "Mike"  
    End If  
    If cmbMetar.ListIndex = 13 Then  
        txtMetar3.Text = "November"  
    End If  
    If cmbMetar.ListIndex = 14 Then  
        txtMetar3.Text = "Oscar"  
    End If  
    If cmbMetar.ListIndex = 15 Then  
        txtMetar3.Text = "Papa"  
    End If  
    If cmbMetar.ListIndex = 16 Then  
        txtMetar3.Text = "Quebec"  
    End If  
    If cmbMetar.ListIndex = 17 Then  
        txtMetar3.Text = "Romeo"  
    End If  
    If cmbMetar.ListIndex = 18 Then  
        txtMetar3.Text = "Sierra"  
    End If  
    If cmbMetar.ListIndex = 19 Then  
        txtMetar3.Text = "Tango"  
    End If  
    If cmbMetar.ListIndex = 20 Then  
        txtMetar3.Text = "Uniform"  
    End If  
    If cmbMetar.ListIndex = 21 Then  
        txtMetar3.Text = "Victor"  
    End If  
    If cmbMetar.ListIndex = 22 Then  
        txtMetar3.Text = "Whiskey"  
    End If
```

```

If cmbMetar.ListIndex = 23 Then
    txtMetar3.Text = "X-ray"
End If
If cmbMetar.ListIndex = 24 Then
    txtMetar3.Text = "Yankee"
End If
If cmbMetar.ListIndex = 25 Then
    txtMetar3.Text = "zulu"
End If
    metar
End Sub

```

```

Private Sub ATIS()
    txtDate2.Text = txtDate.Text
    txtTime2.Text = txtTime.Text
    txtDirection2.Text = txtDirection.Text
    txtSpeed2.Text = txtSpeed.Text
    txtVisibility2.Text = txtVisibility.Text
    txtWeather2.Text = txtWeather.Text
    txtClouds2.Text = txtClouds.Text
    txtFeet2.Text = txtFeet.Text
    txtTemperature2.Text = txtTemperature.Text
    txtDew2.Text = txtDew.Text
    txtQnh2.Text = txtQnh.Text
End Sub

```

```

Private Function ValidSaving()
On Error Resume Next
    ValidSaving = True
    If Not CheckData(txtDate, "Isi Field Date") Then
        ValidSaving = False
        txtDate.SetFocus
        Exit Function
    End If
    If Not CheckData(txtTime, "Isi Field Time") Then
        ValidSaving = False
        txtTime.SetFocus
        Exit Function
    End If
    If Not CheckData(txtMetar, "Sorot Pilihan Weather !") Then
        ValidSaving = False
        cmbMetar.SetFocus
        Exit Function
    End If
End Function

```

```

Private Sub TIME()
If txtTime2.Text = "08:00" Then
    z = baca.Speak(" Weather, " + " At " + " Zero Eight Zero Zero " + " Zulu ", SVSFlagsAsync)
End If
If txtTime2.Text = "09:53" Then
    z = baca.Speak(" Weather, " + " At " + " Zero Niner Five Three " + " Zulu ", SVSFlagsAsync)
End If
If txtTime2.Text = "12:17" Then
    z = baca.Speak(" Weather, " + " At " + " One Two One Seven " + " Zulu ", SVSFlagsAsync)
End If
If txtTime2.Text = "19:56" Then
    z = baca.Speak(" Weather, " + " At " + " One Niner Five Six " + " Zulu ", SVSFlagsAsync)
End If
End Sub

```

```

Private Sub WIND()
If txtDirection2.Text = "0" Then
    z = baca.Speak("wind, " + " Zero Zero Zero " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "1" Then
    z = baca.Speak("wind, " + " Zero Zero One " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "2" Then
    z = baca.Speak("wind, " + " Zero Zero Two " + " Degrees ", SVSFlagsAsync)

```

```
End If
If txtDirection2.Text = "3" Then
    z = baca.Speak("wind, " + " Zero Zero Three " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "4" Then
    z = baca.Speak("wind, " + " Zero Zero Four " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "5" Then
    z = baca.Speak("wind, " + " Zero Zero Five " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "6" Then
    z = baca.Speak("wind, " + " Zero Zero Six " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "7" Then
    z = baca.Speak("wind, " + " Zero Zero Seven " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "8" Then
    z = baca.Speak("wind, " + " Zero Zero Eight " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "9" Then
    z = baca.Speak("wind, " + " Zero Zero Niner " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "10" Then
    z = baca.Speak("wind, " + " Zero One Zero " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "11" Then
    z = baca.Speak("wind, " + " Zero One One " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "12" Then
    z = baca.Speak("wind, " + " Zero One Two " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "13" Then
    z = baca.Speak("wind, " + " Zero One Three " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "14" Then
    z = baca.Speak("wind, " + " Zero One Four " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "15" Then
    z = baca.Speak("wind, " + " Zero One Five " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "16" Then
    z = baca.Speak("wind, " + " Zero One Six " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "17" Then
    z = baca.Speak("wind, " + " Zero One Seven " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "18" Then
    z = baca.Speak("wind, " + " Zero One Eight " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "19" Then
    z = baca.Speak("wind, " + " Zero One Niner " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "20" Then
    z = baca.Speak("wind, " + " Zero Two Zero " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "21" Then
    z = baca.Speak("wind, " + " Zero Two One " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "22" Then
    z = baca.Speak("wind, " + " Zero Two Two " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "23" Then
    z = baca.Speak("wind, " + " Zero Two Three " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "24" Then
    z = baca.Speak("wind, " + " Zero Two Four " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "25" Then
    z = baca.Speak("wind, " + " Zero Two Five " + " Degrees ", SVSFlagsAsync)
End If
```

```
If txtDirection2.Text = "26" Then
    z = baca.Speak("wind, " + " Zero Two Six " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "27" Then
    z = baca.Speak("wind, " + " Zero Two Seven " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "28" Then
    z = baca.Speak("wind, " + " Zero Two Eight " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "29" Then
    z = baca.Speak("wind, " + " Zero Two Niner " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "30" Then
    z = baca.Speak("wind, " + " Zero Three Zero " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "31" Then
    z = baca.Speak("wind, " + " Zero Three One " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "32" Then
    z = baca.Speak("wind, " + " Zero Three Two " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "33" Then
    z = baca.Speak("wind, " + " Zero Three Three " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "34" Then
    z = baca.Speak("wind, " + " Zero Three Four " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "35" Then
    z = baca.Speak("wind, " + " Zero Three Five " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "36" Then
    z = baca.Speak("wind, " + " Zero Three Six " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "37" Then
    z = baca.Speak("wind, " + " Zero Three Seven " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "38" Then
    z = baca.Speak("wind, " + " Zero Three Eight " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "39" Then
    z = baca.Speak("wind, " + " Zero Three Niner " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "40" Then
    z = baca.Speak("wind, " + " Zero Four Zero " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "41" Then
    z = baca.Speak("wind, " + " Zero Four One " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "42" Then
    z = baca.Speak("wind, " + " Zero Four Two " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "43" Then
    z = baca.Speak("wind, " + " Zero Four Three " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "44" Then
    z = baca.Speak("wind, " + " Zero Four Four " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "45" Then
    z = baca.Speak("wind, " + " Zero Four Five " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "46" Then
    z = baca.Speak("wind, " + " Zero Four Six " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "47" Then
    z = baca.Speak("wind, " + " Zero Four Seven " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "48" Then
    z = baca.Speak("wind, " + " Zero Four Eight " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "49" Then
```

```
z = baca.Speak("wind, " + " Zero Four Niner " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "50" Then
z = baca.Speak("wind, " + " Zero Five Zero " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "51" Then
z = baca.Speak("wind, " + " Zero Five One " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "52" Then
z = baca.Speak("wind, " + " Zero Five Two " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "53" Then
z = baca.Speak("wind, " + " Zero Five Three " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "54" Then
z = baca.Speak("wind, " + " Zero Five Four " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "55" Then
z = baca.Speak("wind, " + " Zero Five Five " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "56" Then
z = baca.Speak("wind, " + " Zero Five Six " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "57" Then
z = baca.Speak("wind, " + " Zero Five Seven " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "58" Then
z = baca.Speak("wind, " + " Zero Five Eight " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "59" Then
z = baca.Speak("wind, " + " Zero Five Niner " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "60" Then
z = baca.Speak("wind, " + " Zero Six Zero " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "61" Then
z = baca.Speak("wind, " + " Zero Six One " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "62" Then
z = baca.Speak("wind, " + " Zero Six Two " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "63" Then
z = baca.Speak("wind, " + " Zero Six Three " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "64" Then
z = baca.Speak("wind, " + " Zero Six Four " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "65" Then
z = baca.Speak("wind, " + " Zero Six Five " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "66" Then
z = baca.Speak("wind, " + " Zero Six Six " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "67" Then
z = baca.Speak("wind, " + " Zero Six Seven " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "68" Then
z = baca.Speak("wind, " + " Zero Six Eight " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "69" Then
z = baca.Speak("wind, " + " Zero Six Niner " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "70" Then
z = baca.Speak("wind, " + " Zero Seven Zero " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "71" Then
z = baca.Speak("wind, " + " Zero Seven One " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "72" Then
z = baca.Speak("wind, " + " Zero Seven Two " + " Degrees ", SVSFlagsAsync)
```

```
End If
If txtDirection2.Text = "73" Then
    z = baca.Speak("wind, " + " Zero Seven Three " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "74" Then
    z = baca.Speak("wind, " + " Zero Seven Four " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "75" Then
    z = baca.Speak("wind, " + " Zero Seven Five " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "76" Then
    z = baca.Speak("wind, " + " Zero Seven Six " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "77" Then
    z = baca.Speak("wind, " + " Zero Seven Seven " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "78" Then
    z = baca.Speak("wind, " + " Zero Seven Eight " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "79" Then
    z = baca.Speak("wind, " + " Zero Seven Niner " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "80" Then
    z = baca.Speak("wind, " + " Zero Eight Zero " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "81" Then
    z = baca.Speak("wind, " + " Zero Eight One " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "82" Then
    z = baca.Speak("wind, " + " Zero Eight Two " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "83" Then
    z = baca.Speak("wind, " + " Zero Eight Three " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "84" Then
    z = baca.Speak("wind, " + " Zero Eight Four " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "85" Then
    z = baca.Speak("wind, " + " Zero Eight Five " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "86" Then
    z = baca.Speak("wind, " + " Zero Eight Six " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "87" Then
    z = baca.Speak("wind, " + " Zero Eight Seven " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "88" Then
    z = baca.Speak("wind, " + " Zero Eight Eight " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "89" Then
    z = baca.Speak("wind, " + " Zero Eight Niner " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "90" Then
    z = baca.Speak("wind, " + " Zero Niner Zero " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "91" Then
    z = baca.Speak("wind, " + " Zero Niner One " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "92" Then
    z = baca.Speak("wind, " + " Zero Niner Two " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "93" Then
    z = baca.Speak("wind, " + " Zero Niner Three " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "94" Then
    z = baca.Speak("wind, " + " Zero Niner Four " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "95" Then
    z = baca.Speak("wind, " + " Zero Niner Five " + " Degrees ", SVSFlagsAsync)
End If
```

```
If txtDirection2.Text = "96" Then
    z = baca.Speak("wind, " + " Zero Niner Six " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "97" Then
    z = baca.Speak("wind, " + " Zero Niner Seven " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "98" Then
    z = baca.Speak("wind, " + " Zero Niner Eight " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "99" Then
    z = baca.Speak("wind, " + " Zero Niner Niner " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "100" Then
    z = baca.Speak("wind, " + " One Zero Zero " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "101" Then
    z = baca.Speak("wind, " + " One Zero One " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "102" Then
    z = baca.Speak("wind, " + " One Zero Two " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "103" Then
    z = baca.Speak("wind, " + " One Zero Three " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "104" Then
    z = baca.Speak("wind, " + " One Zero Four " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "105" Then
    z = baca.Speak("wind, " + " One Zero Five " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "106" Then
    z = baca.Speak("wind, " + " One Zero Six " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "107" Then
    z = baca.Speak("wind, " + " One Zero Seven " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "108" Then
    z = baca.Speak("wind, " + " One Zero Eight " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "109" Then
    z = baca.Speak("wind, " + " One Zero Niner " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "110" Then
    z = baca.Speak("wind, " + " One One Zero " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "111" Then
    z = baca.Speak("wind, " + " One One One " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "112" Then
    z = baca.Speak("wind, " + " One One Two " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "113" Then
    z = baca.Speak("wind, " + " One One Three " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "114" Then
    z = baca.Speak("wind, " + " One One Four " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "115" Then
    z = baca.Speak("wind, " + " One One Five " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "116" Then
    z = baca.Speak("wind, " + " One One Six " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "117" Then
    z = baca.Speak("wind, " + " One One Seven " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "118" Then
    z = baca.Speak("wind, " + " One One Eight " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "119" Then
```



```
z = baca.Speak("wind, " + " One One Niner " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "120" Then
z = baca.Speak("wind, " + " One Two Zero " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "121" Then
z = baca.Speak("wind, " + " One Two One " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "122" Then
z = baca.Speak("wind, " + " One Two Two " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "123" Then
z = baca.Speak("wind, " + " One Two Three " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "124" Then
z = baca.Speak("wind, " + " One Two Four " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "125" Then
z = baca.Speak("wind, " + " One Two Five " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "126" Then
z = baca.Speak("wind, " + " One Two Six " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "127" Then
z = baca.Speak("wind, " + " One Two Seven " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "128" Then
z = baca.Speak("wind, " + " One Two Eight " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "129" Then
z = baca.Speak("wind, " + " One Two Niner " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "130" Then
z = baca.Speak("wind, " + " One Three Zero " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "131" Then
z = baca.Speak("wind, " + " One Three One " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "132" Then
z = baca.Speak("wind, " + " One Three Two " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "133" Then
z = baca.Speak("wind, " + " One Three Three " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "134" Then
z = baca.Speak("wind, " + " One Three Four " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "135" Then
z = baca.Speak("wind, " + " One Three Five " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "136" Then
z = baca.Speak("wind, " + " One Three Six " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "137" Then
z = baca.Speak("wind, " + " One Three Seven " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "138" Then
z = baca.Speak("wind, " + " One Three Eight " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "139" Then
z = baca.Speak("wind, " + " One Three Niner " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "140" Then
z = baca.Speak("wind, " + " One Four Zero " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "141" Then
z = baca.Speak("wind, " + " One Four One " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "142" Then
z = baca.Speak("wind, " + " One Four Two " + " Degrees ", SVSFlagsAsync)
```

```

End If
If txtDirection2.Text = "143" Then
    z = baca.Speak("wind, " + " One Four Three " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "144" Then
    z = baca.Speak("wind, " + " One Four Four " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "145" Then
    z = baca.Speak("wind, " + " One Four Five " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "146" Then
    z = baca.Speak("wind, " + " One Four Six " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "147" Then
    z = baca.Speak("wind, " + " One Four Seven " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "148" Then
    z = baca.Speak("wind, " + " One Four Eight " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "149" Then
    z = baca.Speak("wind, " + " One Four Niner " + " Degrees ", SVSFlagsAsync)
End If
If txtDirection2.Text = "150" Then
    z = baca.Speak("wind, " + " One Five Zero " + " Degrees ", SVSFlagsAsync)
End If
End sub

```

```

Private Sub SPEED()
If txtSpeed2.Text = "1" Then
    z = baca.Speak(" one " + " Knots ", SVSFlagsAsync)
End If
If txtSpeed2.Text = "2" Then
    z = baca.Speak(" two " + " Knots ", SVSFlagsAsync)
End If
If txtSpeed2.Text = "3" Then
    z = baca.Speak(" three " + " Knots ", SVSFlagsAsync)
End If
If txtSpeed2.Text = "4" Then
    z = baca.Speak(" four " + " Knots ", SVSFlagsAsync)
End If
If txtSpeed2.Text = "5" Then
    z = baca.Speak(" five " + " Knots ", SVSFlagsAsync)
End If
If txtSpeed2.Text = "6" Then
    z = baca.Speak(" six " + " Knots ", SVSFlagsAsync)
End If
If txtSpeed2.Text = "7" Then
    z = baca.Speak(" seven " + " Knots ", SVSFlagsAsync)
End If
If txtSpeed2.Text = "8" Then
    z = baca.Speak(" eight " + " Knots ", SVSFlagsAsync)
End If
If txtSpeed2.Text = "9" Then
    z = baca.Speak(" niner " + " Knots ", SVSFlagsAsync)
End If
If txtSpeed2.Text = "10" Then
    z = baca.Speak(" ten " + " Knots ", SVSFlagsAsync)
End If
If txtSpeed2.Text = "11" Then
    z = baca.Speak(" one one " + " Knots ", SVSFlagsAsync)
End If
If txtSpeed2.Text = "12" Then
    z = baca.Speak(" one two " + " Knots ", SVSFlagsAsync)
End If
If txtSpeed2.Text = "13" Then
    z = baca.Speak(" one three " + " Knots ", SVSFlagsAsync)
End If
If txtSpeed2.Text = "14" Then
    z = baca.Speak(" one four " + " Knots ", SVSFlagsAsync)
End If

```

```

If txtSpeed2.Text = "15" Then
    z = baca.Speak(" one five " + " Knots ", SVSFlagsAsync)
End If
If txtSpeed2.Text = "16" Then
    z = baca.Speak(" one six " + " Knots ", SVSFlagsAsync)
End If
If txtSpeed2.Text = "17" Then
    z = baca.Speak(" one seven " + " Knots ", SVSFlagsAsync)
End If
If txtSpeed2.Text = "18" Then
    z = baca.Speak(" one eight " + " Knots ", SVSFlagsAsync)
End If
If txtSpeed2.Text = "19" Then
    z = baca.Speak(" one niner " + " Knots ", SVSFlagsAsync)
End If
If txtSpeed2.Text = "20" Then
    z = baca.Speak(" two zero " + " Knots ", SVSFlagsAsync)
End If
If txtSpeed2.Text = "21" Then
    z = baca.Speak(" two one " + " Knots ", SVSFlagsAsync)
End If
If txtSpeed2.Text = "22" Then
    z = baca.Speak(" two two " + " Knots ", SVSFlagsAsync)
End If
If txtSpeed2.Text = "23" Then
    z = baca.Speak(" two three " + " Knots ", SVSFlagsAsync)
End If
If txtSpeed2.Text = "24" Then
    z = baca.Speak(" Two Four " + " Knots ", SVSFlagsAsync)
End If
If txtSpeed2.Text = "25" Then
    z = baca.Speak(" two five " + " Knots ", SVSFlagsAsync)
End If
End Sub

Private Sub VISIBILITY()
If txtVisibility2.Text = "1" Then
    z = baca.Speak(" Visibility " + " One ", SVSFlagsAsync)
End If
If txtVisibility2.Text = "2" Then
    z = baca.Speak(" Visibility " + " Two ", SVSFlagsAsync)
End If
If txtVisibility2.Text = "3" Then
    z = baca.Speak(" Visibility " + " Three ", SVSFlagsAsync)
End If
If txtVisibility2.Text = "4" Then
    z = baca.Speak(" Visibility " + " Four ", SVSFlagsAsync)
End If
If txtVisibility2.Text = "5" Then
    z = baca.Speak(" Visibility " + " Five ", SVSFlagsAsync)
End If
End Sub

Private Sub weather()
If txtWeather2.Text = "RAIN" Then
    z = baca.Speak(" Present Weather " + " Rain ", SVSFlagsAsync)
End If
If txtWeather2.Text = "FOG" Then
    z = baca.Speak(" Present Weather " + " Fog ", SVSFlagsAsync)
End If
If txtWeather2.Text = "HAZE" Then
    z = baca.Speak(" Present Weather " + " Haze ", SVSFlagsAsync)
End If
If txtWeather2.Text = "NILL" Then
    z = baca.Speak(" Present Weather " + " Nill ", SVSFlagsAsync)
End If
End Sub

Private Sub clouds()
If txtClouds2.Text = "BKN" Then

```

```

    z = baca.Speak(" Clouds " + " Broken ", SVSFlagsAsync)
End If
If txtClouds2.Text = "SKC" Then
    z = baca.Speak(" Clouds " + " Sky Clear ", SVSFlagsAsync)
End If
If txtClouds2.Text = "FEW" Then
    z = baca.Speak(" Clouds " + " Few ", SVSFlagsAsync)
End If
End Sub

Private Sub FEET()
If txtFeet2.Text = "2000" Then
    z = baca.Speak(" Two Thousand " + " Feet ", SVSFlagsAsync)
End If
If txtFeet2.Text = "2100" Then
    z = baca.Speak(" Two Thousand One Hundred " + " Feet ", SVSFlagsAsync)
End If
If txtFeet2.Text = "2200" Then
    z = baca.Speak(" Two Thousand Two Hundred " + " Feet ", SVSFlagsAsync)
End If
If txtFeet2.Text = "2300" Then
    z = baca.Speak(" Two Thousand Three Hundred " + " Feet ", SVSFlagsAsync)
End If
If txtFeet2.Text = "2400" Then
    z = baca.Speak(" Two Thousand Four Hundred " + " Feet ", SVSFlagsAsync)
End If
If txtFeet2.Text = "2500" Then
    z = baca.Speak(" Two Thousand Five Hundred " + " Feet ", SVSFlagsAsync)
End If
End Sub

Private Sub TEMPERATURE()
If txtTemperature2.Text = "12" Then
    z = baca.Speak(" Temperature " + " One Two ", SVSFlagsAsync)
End If
If txtTemperature2.Text = "13" Then
    z = baca.Speak(" Temperature " + " One Three ", SVSFlagsAsync)
End If
If txtTemperature2.Text = "14" Then
    z = baca.Speak(" Temperature " + " One Four ", SVSFlagsAsync)
End If
If txtTemperature2.Text = "15" Then
    z = baca.Speak(" Temperature " + " One Five ", SVSFlagsAsync)
End If
If txtTemperature2.Text = "16" Then
    z = baca.Speak(" Temperature " + " One Six ", SVSFlagsAsync)
End If
If txtTemperature2.Text = "17" Then
    z = baca.Speak(" Temperature " + " One Seven ", SVSFlagsAsync)
End If
If txtTemperature2.Text = "18" Then
    z = baca.Speak(" Temperature " + " One Eight ", SVSFlagsAsync)
End If
If txtTemperature2.Text = "19" Then
    z = baca.Speak(" Temperature " + " One Niner ", SVSFlagsAsync)
End If
If txtTemperature2.Text = "20" Then
    z = baca.Speak(" Temperature " + " Two Zero ", SVSFlagsAsync)
End If
If txtTemperature2.Text = "21" Then
    z = baca.Speak(" Temperature " + " Two One ", SVSFlagsAsync)
End If
If txtTemperature2.Text = "22" Then
    z = baca.Speak(" Temperature " + " Two Two ", SVSFlagsAsync)
End If
End Sub

Private Sub DEW()
If txtDew2.Text = "12" Then
    z = baca.Speak(" Dew Point " + " One Two ", SVSFlagsAsync)

```

```

End If
If txtDew2.Text = "13" Then
    z = baca.Speak(" Dew Point " + " One Three ", SVSFlagsAsync)
End If
If txtDew2.Text = "14" Then
    z = baca.Speak(" Dew Point " + " One Four ", SVSFlagsAsync)
End If
If txtDew2.Text = "15" Then
    z = baca.Speak(" Dew Point " + " One Five ", SVSFlagsAsync)
End If
If txtDew2.Text = "16" Then
    z = baca.Speak(" Dew Point " + " One Six ", SVSFlagsAsync)
End If
If txtDew2.Text = "17" Then
    z = baca.Speak(" Dew Point " + " One Seven ", SVSFlagsAsync)
End If
If txtDew2.Text = "18" Then
    z = baca.Speak(" Dew Point " + " One Eight ", SVSFlagsAsync)
End If
If txtDew2.Text = "19" Then
    z = baca.Speak(" Dew Point " + " One Niner ", SVSFlagsAsync)
End If
If txtDew2.Text = "20" Then
    z = baca.Speak(" Dew Point " + " Two Zero ", SVSFlagsAsync)
End If
If txtDew2.Text = "21" Then
    z = baca.Speak(" Dew Point " + " Two One ", SVSFlagsAsync)
End If
If txtDew2.Text = "22" Then
    z = baca.Speak(" Dew Point " + " Two two ", SVSFlagsAsync)
End If
If txtDew2.Text = "23" Then
    z = baca.Speak(" Dew Point " + " Two Three ", SVSFlagsAsync)
End If
If txtDew2.Text = "24" Then
    z = baca.Speak(" Dew Point " + " Two Four ", SVSFlagsAsync)
End If
End Sub

Private Sub QNH()
If txtQnh2.Text = "1001" Then
    z = baca.Speak(" Altimeter " + " One Zero Zero One ", SVSFlagsAsync)
End If
If txtQnh2.Text = "1002" Then
    z = baca.Speak(" Altimeter " + " One Zero Zero Two ", SVSFlagsAsync)
End If
If txtQnh2.Text = "1003" Then
    z = baca.Speak(" Altimeter " + " One Zero Zero Three ", SVSFlagsAsync)
End If
If txtQnh2.Text = "1004" Then
    z = baca.Speak(" Altimeter " + " One Zero Zero Four ", SVSFlagsAsync)
End If
If txtQnh2.Text = "1005" Then
    z = baca.Speak(" Altimeter " + " One Zero Zero Five ", SVSFlagsAsync)
End If
If txtQnh2.Text = "1006" Then
    z = baca.Speak(" Altimeter " + " One Zero Zero Six ", SVSFlagsAsync)
End If
If txtQnh2.Text = "1007" Then
    z = baca.Speak(" Altimeter " + " One Zero Zero Seven ", SVSFlagsAsync)
End If
If txtQnh2.Text = "1008" Then
    z = baca.Speak(" Altimeter " + " One Zero Zero Eight ", SVSFlagsAsync)
End If
If txtQnh2.Text = "1009" Then
    z = baca.Speak(" Altimeter " + " One Zero Zero Niner ", SVSFlagsAsync)
End If
If txtQnh2.Text = "1010" Then
    z = baca.Speak(" Altimeter " + " One Zero One Zero ", SVSFlagsAsync)
End If

```

```

If txtQnh2.Text = "1011" Then
    z = baca.Speak(" Altimeter " + " One Zero One One ", SVSFlagsAsync)
End If
If txtQnh2.Text = "1012" Then
    z = baca.Speak(" Altimeter " + " One Zero One Two ", SVSFlagsAsync)
End If
If txtQnh2.Text = "1013" Then
    z = baca.Speak(" Altimeter " + " One Zero One Three ", SVSFlagsAsync)
End If
If txtQnh2.Text = "1014" Then
    z = baca.Speak(" Altimeter " + " One Zero One Four ", SVSFlagsAsync)
End If
If txtQnh2.Text = "1015" Then
    z = baca.Speak(" Altimeter " + " One Zero One Five ", SVSFlagsAsync)
End If
End Sub

Private Sub datagrid_HeadClick(ByVal ColIndex As Integer)
    cOrder = GetGridOrder(datagrid, ColIndex, False)
    RefreshGrid
End Sub

Private Sub datagrid_FilterChange()
    GoFilter datagrid, rsclone
End Sub

Private Sub datagrid_RowColChange>LastRow As Variant, ByVal LastCol As Integer)
    If Not IsNull(datagrid.Columns(0).Value) Then
        GetList datagrid.Columns(0).Value
    End If
End Sub

Private Sub Tmr4Clock_Timer()
    txtDate.Text = Date
    txtTime.Text = Format(TIME, "hh:mm")
End Sub

Sub metar()
    If cmbMetar.ListIndex = 0 Then
        txtMetar.Text = "OK"
    End If
    If cmbMetar.ListIndex = 1 Then
        txtMetar.Text = "OK"
    End If
    If cmbMetar.ListIndex = 2 Then
        txtMetar.Text = "OK"
    End If
    If cmbMetar.ListIndex = 3 Then
        txtMetar.Text = "OK"
    End If
    If cmbMetar.ListIndex = 4 Then
        txtMetar.Text = "OK"
    End If
    If cmbMetar.ListIndex = 5 Then
        txtMetar.Text = "OK"
    End If
    If cmbMetar.ListIndex = 6 Then
        txtMetar.Text = "OK"
    End If
    If cmbMetar.ListIndex = 7 Then
        txtMetar.Text = "OK"
    End If
    If cmbMetar.ListIndex = 8 Then
        txtMetar.Text = "OK"
    End If
    If cmbMetar.ListIndex = 9 Then
        txtMetar.Text = "OK"
    End If
    If cmbMetar.ListIndex = 10 Then
        txtMetar.Text = "OK"
    End If

```

```

End If
If cmbMetar.ListIndex = 11 Then
    txtMetar.Text = "OK"
End If
If cmbMetar.ListIndex = 12 Then
    txtMetar.Text = "OK"
End If
If cmbMetar.ListIndex = 13 Then
    txtMetar.Text = "OK"
End If
If cmbMetar.ListIndex = 14 Then
    txtMetar.Text = "OK"
End If
If cmbMetar.ListIndex = 15 Then
    txtMetar.Text = "OK"
End If
If cmbMetar.ListIndex = 16 Then
    txtMetar.Text = "OK"
End If
If cmbMetar.ListIndex = 17 Then
    txtMetar.Text = "OK"
End If
If cmbMetar.ListIndex = 18 Then
    txtMetar.Text = "OK"
End If
If cmbMetar.ListIndex = 19 Then
    txtMetar.Text = "OK"
End If
If cmbMetar.ListIndex = 20 Then
    txtMetar.Text = "OK"
End If
If cmbMetar.ListIndex = 21 Then
    txtMetar.Text = "OK"
End If
If cmbMetar.ListIndex = 22 Then
    txtMetar.Text = "OK"
End If
If cmbMetar.ListIndex = 23 Then
    txtMetar.Text = "OK"
End If
If cmbMetar.ListIndex = 24 Then
    txtMetar.Text = "OK"
End If
If cmbMetar.ListIndex = 25 Then
    txtMetar.Text = "OK"
End If
End Sub

Private Sub txtDirection_KeyPress(KeyAscii As Integer)
Dim sTemplate As String
Dim sChar As String
Dim sBig As String
sTemplate = "!@#$%^&*()_+-.=<>?/[|]{ };"
If InStr(1, sTemplate, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
sChar = "abcdefghijklmnopqrstuvwxy"
If InStr(1, sChar, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
sBig = "ABCDEFGHIJKLMNPOQRSTUVWXYZ"
If InStr(1, sBig, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
End Sub

Private Sub txtSpeed_KeyPress(KeyAscii As Integer)
Dim sTemplate As String
Dim sChar As String
Dim sBig As String

```

```

sTemplate = "!@#$$%^&*()_+-.=<>?/[|]{ };"
If InStr(1, sTemplate, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
sChar = "abcdefghijklmnopqrstuvwxyz"
If InStr(1, sChar, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
sBig = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
If InStr(1, sBig, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
End Sub

Private Sub txtVisibility_KeyPress(KeyAscii As Integer)
Dim sTemplate As String
Dim sChar As String
Dim sBig As String
sTemplate = "!@#$$%^&*()_+-.=<>?/[|]{ };"
If InStr(1, sTemplate, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
sChar = "abcdefghijklmnopqrstuvwxyz"
If InStr(1, sChar, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
sBig = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
If InStr(1, sBig, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
End Sub

Private Sub txtFeet_KeyPress(KeyAscii As Integer)
Dim sTemplate As String
Dim sChar As String
Dim sBig As String
sTemplate = "!@#$$%^&*()_+-.=<>?/[|]{ };"
If InStr(1, sTemplate, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
sChar = "abcdefghijklmnopqrstuvwxyz"
If InStr(1, sChar, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
sBig = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
If InStr(1, sBig, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
End Sub

Private Sub txtTemperature_KeyPress(KeyAscii As Integer)
Dim sTemplate As String
Dim sChar As String
Dim sBig As String
sTemplate = "!@#$$%^&*()_+-.=<>?/[|]{ };"
If InStr(1, sTemplate, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
sChar = "abcdefghijklmnopqrstuvwxyz"
If InStr(1, sChar, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
sBig = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
If InStr(1, sBig, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
End Sub

Private Sub txtDew_KeyPress(KeyAscii As Integer)
Dim sTemplate As String

```



```

Dim sChar As String
Dim sBig As String
sTemplate = "!@#%$%^&*()_+-.=<>?/[|]{:;""'"
If InStr(1, sTemplate, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
sChar = "abcdefghijklmnopqrstuvwxyz"
If InStr(1, sChar, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
sBig = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
If InStr(1, sBig, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
End Sub

Private Sub txtQnh_KeyPress(KeyAscii As Integer)
Dim sTemplate As String
Dim sChar As String
Dim sBig As String
sTemplate = "!@#%$%^&*()_+-.=<>?/[|]{:;""'"
If InStr(1, sTemplate, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
sChar = "abcdefghijklmnopqrstuvwxyz"
If InStr(1, sChar, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
sBig = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
If InStr(1, sBig, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
End Sub

Private Sub cmbMetar_KeyPress(KeyAscii As Integer)
If KeyAscii >= 0 Then KeyAscii = 0
End Sub

Sub GetUser()
Dim rs As New ADODB.Recordset

Set rs = ExecSQL(GetDSN, "SELECT * FROM user WHERE unit = 'BO' AND status = 'Online'")
If rs.RecordCount > 0 Then
    txtKeterangan.Text = rs!nama
Else
    InitValue
End If
End Sub

Public Sub Offline()
imgOffline.Visible = True
imgOnline.Visible = False
imgConnecting.Visible = False
txtPort.Enabled = True
txtNick.Enabled = False
txtIP.Enabled = True
cmdSendText.Enabled = False
txtSendText.Enabled = False
optHostGuest(0).Enabled = True
optHostGuest(1).Enabled = True
StatusBar1.Panels(2).Text = "TIDAK TERKONEKSI"
Winsock1.Close
tmrClock.Enabled = False
S = 0
m = 0
FrameCounter.Visible = False
lblSeconds.Caption = ""
lblMinutes.Caption = ""
End Sub

```

```

Public Sub Online()
    imgOnline.Visible = True
    imgOffline.Visible = False
    imgConnecting.Visible = False
    txtIP.Enabled = False
    txtPort.Enabled = False
    txtNick.Enabled = False
    cmdSendText.Enabled = True
    txtSendText.Enabled = True
    txtIncomingData.Text = ""
    optHostGuest(0).Enabled = False
    optHostGuest(1).Enabled = False
    cmdConnect.Caption = "Disconnect"
End Sub

Public Sub Connecting()
    imgConnecting.Visible = True
    imgOffline.Visible = False
    imgOnline.Visible = False
    cmdSendText.Enabled = False
    txtSendText.Enabled = False
    StatusBar1.Panels(2).Text = "CONTACTING HOST..."
    If Winsock1.State <> sckClosed Then
        Winsock1.Close
    End If
        IP = txtIP.Text
    If LCase$(IP) = "localhost" Then IP = Winsock1.LocalIP
    Winsock1.Connect txtIP.Text, txtPort.Text
End Sub

Public Sub Listening()
    imgConnecting.Visible = True
    imgOffline.Visible = False
    imgOnline.Visible = False
    cmdSendText.Enabled = False
    txtSendText.Enabled = False
        Winsock1.Close
        Winsock1.LocalPort = txtPort.Text 'set the port
        Winsock1.Listen 'tell it to listen
        StatusBar1.Panels(2).Text = "LISTENING ON PORT: " & txtPort.Text
End Sub

Private Sub optHostGuest_Click(Index As Integer)
    Select Case Index
        Case 0:
            IP = Winsock1.LocalIP
            txtIP.Text = IP
            txtIP.Locked = True
            txtIP.Enabled = False
            cmdConnect.Caption = "Listen"
            StatusBar1.Panels(1).Text = "HOST MODE"
        Case 1:
            txtIP.Text = "localhost"
            cmdConnect.Caption = "Connect"
            txtIP.Locked = False
            txtIP.Enabled = True
            StatusBar1.Panels(1).Text = "GUEST MODE"
    End Select
End Sub

Private Sub tmrClock_Timer()
    S = S + 1
    If S = 60 Then
        S = 0
        m = m + 1
    End If
    If m > 0 Then
        lblMinutes.Caption = m
    End If
    lblSeconds.Caption = S
End Sub

```

```

    DoEvents
End Sub

Private Sub txtIncomingData_Change()
    txtIncomingData.SelStart = Len(txtIncomingData)
End Sub

Private Sub txtSendText_KeyPress(KeyAscii As Integer) 'mengirim pesan komunikasi
    If KeyAscii = 13 Then
        cmdSendText_Click
        KeyAscii = 0
    End If
End Sub

Private Sub Winsock1_Close() 'mendeteksi status komunikasi unit
    Select Case optHostGuest(0).Value
        Case True:
            cmdConnect_Click
            cmdConnect_Click
            MsgBox "Connection terminated by Guest. Server has been reset and awaiting a new client..."
'menampilkan pesan status komunikasi unit
        Case False:
            MsgBox "Connection terminated by Host..."
            cmdConnect_Click
    End Select
End Sub

Private Sub Winsock1_Connect()
    Select Case optHostGuest(0).Value
        Case True:

        Case False:
            Call Online
            StatusBar1.Panels(2).Text = "CONNECTED TO HOST"
    End Select
    FrameCounter.Visible = True
    tmrClock.Enabled = True
End Sub

Private Sub Winsock1_ConnectionRequest(ByVal requestID As Long)
    If Winsock1.State <> sckClosed Then
        Winsock1.Close
        Winsock1.LocalPort = txtPort.Text
        Winsock1.Accept requestID
        Online
        StatusBar1.Panels(2).Text = "CLIENT CONNECTED"
        FrameCounter.Visible = True
        tmrClock.Enabled = True
    End If
End Sub

Private Sub Winsock1_DataArrival(ByVal bytesTotal As Long) 'membuka jalur komunikasi
    Dim Data As String
    Winsock1.GetData Data
    txtIncomingData.Text = txtIncomingData.Text + vbCrLf & Data
End Sub

Private Sub Winsock1_Error(ByVal Number As Integer, Description As String, ByVal Scode As Long, ByVal Source As String, ByVal HelpFile As String, ByVal HelpContext As Long, CancelDisplay As Boolean)
    StatusBar1.Panels(2).Text = "WINSOCK ERROR: " & Err
    cmdConnect_Click
    MsgBox "UNABLE TO CONNECT..."
End Sub

```

9. Browse Pengguna BMG

Option Explicit

```
Dim conn As New ADODB.Connection
Dim rs As New ADODB.Recordset
Dim rsclose As New ADODB.Recordset
Dim cOrder As String
```

```
Private Sub Form_Load()
    cOrder = "userid"
    RefreshGrid
    InitGrid datagrid
End Sub
```

```
Private Sub cmdExit_Click()
    Unload Me
End Sub
```

```
Private Sub cmdRefresh_Click()
    RefreshGrid
End Sub
```

```
Private Sub RefreshGrid()
    On Error Resume Next
    unit online 'mendeteksi pengguna
    Set rs = ExecSQL(GetDSN, "SELECT * FROM user WHERE unit = 'BMG' AND qualify = 'USER' AND status = 'Online' ORDER by " & cOrder)
    Set rsclose = rs.Clone
    DoRefreshGrid datagrid, rsclose, Array(6, 3, 3)
End Sub
```

```
Private Sub datagrid_HeadClick(ByVal ColIndex As Integer)
    cOrder = GetGridOrder(datagrid, ColIndex, False)
    RefreshGrid
End Sub
```

```
Private Sub datagrid_FilterChange()
    GoFilter datagrid, rsclose
End Sub
```

10. Browse Pengguna BO

Option Explicit

```
Dim conn As New ADODB.Connection
Dim rs As New ADODB.Recordset
Dim rsclose As New ADODB.Recordset
Dim cOrder As String
```

```
Private Sub Form_Load()
    cOrder = "userid"
    RefreshGrid
    InitGrid datagrid
End Sub
```

```
Private Sub cmdExit_Click()
    Unload Me
End Sub
```

```
Private Sub cmdRefresh_Click()
    RefreshGrid
End Sub
```

```
Private Sub RefreshGrid()
    On Error Resume Next
    unit online 'mendeteksi pengguna
```

```

    Set rs = ExecSQL(GetDSN, "SELECT * FROM user WHERE unit = 'BO' AND qualify = 'USER' AND status =
'Online' ORDER by " & cOrder)
    Set rsclone = rs.Clone
    DoRefreshGrid datagrid, rsclone, Array(6, 3, 3)
End Sub

Private Sub datagrid_HeadClick(ByVal ColIndex As Integer)
    cOrder = GetGridOrder(datagrid, ColIndex, False)
    RefreshGrid
End Sub

Private Sub datagrid_FilterChange()
    GoFilter datagrid, rsclone
End Sub

```

11. Lihat Data Cuaca

Option Explicit

```

Dim conn As New ADODB.Connection
Dim rs As New ADODB.Recordset
Dim rsclone As New ADODB.Recordset
Dim cOrder As String

```

```

Private Sub Form_Load()
    cOrder = "date"
    RefreshGrid
    InitGrid datagrid
End Sub

```

```

Private Sub cmdExit_Click()
    Unload Me
End Sub

```

```

Private Sub RefreshGrid()
On Error Resume Next
    Set rs = ExecSQL(GetDSN, "SELECT * FROM cuaca ORDER by " & cOrder) 'menampilkan data
    Set rsclone = rs.Clone
    DoRefreshGrid datagrid, rsclone, Array(7, 5, 6, 5, 5, 5, 5, 8, 5, 5, 15)
End Sub

```

```

Private Sub datagrid_HeadClick(ByVal ColIndex As Integer)
    cOrder = GetGridOrder(datagrid, ColIndex, False)
    RefreshGrid
End Sub

```

```

Private Sub datagrid_FilterChange()
    GoFilter datagrid, rsclone
End Sub

```

12. Lihat Data METAR

Option Explicit

```

Dim conn As New ADODB.Connection
Dim rs As New ADODB.Recordset
Dim rsclone As New ADODB.Recordset
Dim cOrder As String

```

```

Private Sub Form_Load()
    cOrder = "date"
    RefreshGrid
    InitGrid datagrid
End Sub

```

```
Private Sub cmdExit_Click()
    Unload Me
End Sub
```

```
Private Sub RefreshGrid()
On Error Resume Next
    Set rs = ExecSQL(GetDSN, "SELECT * FROM metar ORDER by " & cOrder) 'menampilkan data
    Set rsclone = rs.Clone
    DoRefreshGrid datagrid, rsclone, Array(3, 3, 3, 3)
End Sub
```

```
Private Sub datagrid_HeadClick(ByVal ColIndex As Integer)
    cOrder = GetGridOrder(datagrid, ColIndex, False)
    RefreshGrid
End Sub
```

```
Private Sub datagrid_FilterChange()
    GoFilter datagrid, rsclone
End Sub
```

13. Lihat Data Pengguna

Option Explicit

```
Dim conn As New ADODB.Connection
Dim rs As New ADODB.Recordset
Dim rsclone As New ADODB.Recordset
Dim cOrder As String
```

```
Private Sub Form_Load()
    cOrder = "userid"
    RefreshGrid
    InitGrid datagrid
End Sub
```

```
Private Sub cmdExit_Click()
    Unload Me
End Sub
```

```
Private Sub RefreshGrid()
On Error Resume Next
    Set rs = ExecSQL(GetDSN, "SELECT * FROM user ORDER by " & cOrder)
    Set rsclone = rs.Clone
    DoRefreshGrid datagrid, rsclone, Array(2, 6, 5, 2, 2, 3)
End Sub
```

```
Private Sub datagrid_HeadClick(ByVal ColIndex As Integer)
    cOrder = GetGridOrder(datagrid, ColIndex, False)
    RefreshGrid
End Sub
```

```
Private Sub datagrid_FilterChange()
    GoFilter datagrid, rsclone
End Sub
```

14. Set Global

Option Explicit

```
Private Sub Form_Load() 'input data global bandara
    InitForm Me
    GetList
    GetUser
```

```
    TabIndex Array(txtAirport, txtSign, txtNegara, _
        txtProvinsi, txtAlamat, txtKeterangan, _
        txtFrekuensi, cmdSimpan, cmdExit)
End Sub
```

```
Private Sub cmdSimpan_Click()
    kelola_global
End Sub
```

```
Private Sub cmdExit_Click()
    Unload Me
End Sub
```

```
Sub InitValue()
    txtAirport.Text = ""
    txtAirport.MaxLength = 40
    txtSign.Text = ""
    txtSign.MaxLength = 5
    txtNegara.Text = ""
    txtNegara.MaxLength = 20
    txtProvinsi.Text = ""
    txtProvinsi.MaxLength = 20
    txtAlamat.Text = ""
    txtAlamat.MaxLength = 40
    txtFrekuensi.Text = ""
    txtFrekuensi.MaxLength = 10
End Sub
```

```
Sub GetList()
    Dim rs As New ADODB.Recordset
    Set rs = ExecSQL(GetDSN, "SELECT * FROM global")
    If rs.RecordCount > 0 Then
        txtAirport.Text = rs!Airport
        txtSign.Text = rs!sign
        txtNegara.Text = rs!negara
        txtProvinsi.Text = rs!provinsi
        txtAlamat.Text = rs!alamat
        txtFrekuensi.Text = rs!frekuensi
        txtKeterangan.Text = rs!UserID
    Else
        InitValue
    End If
End Sub
```

```
Sub GetUser()
    Dim rs As New ADODB.Recordset
    Set rs = ExecSQL(GetDSN, "SELECT * FROM user WHERE unit = 'admin' AND status = 'Online'")
    If rs.RecordCount > 0 Then
        txtKeterangan.Text = rs!nama
    Else
        InitValue
    End If
End Sub
```

```
Function ValidSaving()
    ValidSaving = True
    If Not CheckData(txtAirport.Text, "Isi Nama Airport") Then
        txtAirport.SetFocus
        ValidSaving = False
    End If
```

```

If Not CheckData(txtSign.Text, "Isi Sign Airport") Then
    txtSign.SetFocus
    ValidSaving = False
End If
If Not CheckData(txtNegara.Text, "Isi Nama Negara") Then
    txtNegara.SetFocus
    ValidSaving = False
End If
If Not CheckData(txtProvinsi.Text, "Isi Nama Provinsi") Then
    txtProvinsi.SetFocus
    ValidSaving = False
End If
If Not CheckData(txtAlamat.Text, "Isi Alamat") Then
    txtAlamat.SetFocus
    ValidSaving = False
End If
If Not CheckData(txtFrekuensi.Text, "Isi Nama Airport") Then
    txtFrekuensi.SetFocus
    ValidSaving = False
End If
End Function

Private Sub Form_KeyDown(KeyCode As Integer, Shift As Integer)
    If KeyCode = 13 Then SendKeys "{TAB}"
End Sub

Private Sub txtAirport_KeyPress(KeyAscii As Integer)
    Dim sTemplate As String

    sTemplate = "!@#$$%^&*()_+-.,<>?/\[\]{};:'''\""
    If InStr(1, sTemplate, Chr(KeyAscii)) > 0 Then
        KeyAscii = 0
    End If
End Sub

Private Sub txtSign_KeyPress(KeyAscii As Integer)
    Dim sTemplate As String
    sTemplate = "!@#$$%^&*()_+-.,<>?/\[\]{};:'''\""
    If InStr(1, sTemplate, Chr(KeyAscii)) > 0 Then
        KeyAscii = 0
    End If
    If KeyAscii > 47 And KeyAscii < 58 Then
        KeyAscii = 0
    End If
End Sub

Private Sub txtNegara_KeyPress(KeyAscii As Integer)
    Dim sTemplate As String
    sTemplate = "!@#$$%^&*()_+-.,<>?/\[\]{};:'''\""
    If InStr(1, sTemplate, Chr(KeyAscii)) > 0 Then
        KeyAscii = 0
    End If
    If KeyAscii > 47 And KeyAscii < 58 Then
        KeyAscii = 0
    End If
End Sub

Private Sub txtProvinsi_KeyPress(KeyAscii As Integer)
    Dim sTemplate As String
    sTemplate = "!@#$$%^&*()_+-.,<>?/\[\]{};:'''\""
    If InStr(1, sTemplate, Chr(KeyAscii)) > 0 Then
        KeyAscii = 0
    End If
    If KeyAscii > 47 And KeyAscii < 58 Then
        KeyAscii = 0
    End If
End Sub

Private Sub txtAlamat_KeyPress(KeyAscii As Integer)
    Dim sTemplate As String

```



```

sTemplate = "!@#$$%^&*()_+<=>?/[ \ ] { } ; : ; """"
If InStr(1, sTemplate, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
End Sub

Private Sub txtFrekuensi_KeyPress(KeyAscii As Integer)
Dim sTemplate As String
Dim sChar As String
Dim sBig As String
sTemplate = "!@#$$%^&*()_+<=>?/[ \ ] { } ; : ; """"
If InStr(1, sTemplate, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
sChar = "abcdefghijklmnopqrstuvwxyz"
If InStr(1, sChar, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
sBig = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
If InStr(1, sBig, Chr(KeyAscii)) > 0 Then
    KeyAscii = 0
End If
End Sub

Sub kelola_global()
Dim vaField As Variant
Dim vaValue As Variant
Dim Success As Boolean
Dim rs As New ADODB.Recordset

If ValidSaving Then 'validasi data

    vaField = Array("airport", "sign", "negara", "provinsi", _
        "alamat", "frekuensi", "userid")
    vaValue = Array(txtAirport.Text, txtSign.Text, txtNegara.Text, _
        txtProvinsi.Text, txtAlamat.Text, txtFrekuensi.Text, txtKeterangan.Text)

    Set rs = ExecSQL(GetDSN, "DELETE FROM Global")
    Success = Insert(GetDSN, "global", vaField, vaValue)
    If Not Success Then
        MsgBox "Ada kesalahan penyimpanan data!", , "Informasi"
    Else
        MsgBox "Data tersimpan"
    End If
End If
End Sub

```

15. Laporan Harian Data Cuaca

Option Explicit

```
Private Sub Form_Load()
    dtAwal.Value = Date
End Sub
```

```
Private Sub cmdCetak_Click()
    cetak_laporan_harian_cuaca
End Sub
```

```
Private Sub cmdExit_Click()
    Unload Me
End Sub
```

```
Sub cetak_laporan_harian_cuaca() 'membuat laporan data cuaca
```

```
Dim rsglobal As New ADODB.Recordset
Dim rsdetail As New ADODB.Recordset
Dim rstemp As New ADODB.Recordset
Dim PosY As Long
Dim i As Long
Dim j As Long
```

```
Set rsglobal = ExecSQL(GetDSN, "SELECT * FROM cuaca " & _
"WHERE date >=" & Format(dtAwal.Value, "dd-mm-yyyy") & " AND " & _
"date <=" & Format(dtAwal.Value, "dd-mm-yyyy") & """)
```

```
If rsglobal.RecordCount > 0 Then
    OpenExcel True
    OpenWorkBook App.Path & "\LaporanHarianDataCuaca.xls"
    Set rstemp = ExecSQL(GetDSN, "SELECT * FROM global")
    If rstemp.RecordCount > 0 Then
        PutExcelValue 1, 1, rstemp!Airport
        PutExcelValue 2, 1, rstemp!alamat & " - " & rstemp!provinsi
        PutExcelValue 3, 1, rstemp!negara
    End If
    PutExcelValue 5, 1, dtAwal.Value
    PosY = 8
    For j = 0 To rsglobal.RecordCount - 1
        PutExcelValue PosY, 1, Trim(j + 1) & ". " & Format(rsglobal!Date, "dd/mm/yyyy")
        PutExcelValue PosY, 2, " " & Format(rsglobal!TIME, "hh:mm")
        PutExcelValue PosY, 3, "" & rsglobal!Direction
        PutExcelValue PosY, 4, rsglobal!SPEED
        PutExcelValue PosY, 5, rsglobal!VISIBILITY
        PutExcelValue PosY, 6, rsglobal!weather
        PutExcelValue PosY, 7, rsglobal!clouds
        PutExcelValue PosY, 8, rsglobal!FEET
        PutExcelValue PosY, 9, rsglobal!TEMPERATURE
        PutExcelValue PosY, 10, rsglobal!dewpoint
        PutExcelValue PosY, 11, rsglobal!QNH
        PutExcelValue PosY, 12, rsglobal!UserID
    For i = 1 To 12
        SetCellFontSize PosY, i, 12
    If i >= 12 And i <= 11 Then SetHorizontalAlignment PosY, i, xlRight
    If j = rsglobal.RecordCount - 1 Then
        SetBorderLine PosY, i, LBottom, Inormal
    End If
    Next i
    PosY = PosY + 1
    rsglobal.MoveNext
    Next j
    For i = 1 To 12
        SetCellFontSize PosY, i, 12
        SetHorizontalAlignment PosY, i, xlRight
        SetCellFontBold PosY, i
    Next i
    'ExcelPrintPreview
    'CloseWorkBook
```

```

        'CloseExcel
    Else
        MsgBox "Tidak Ada Data Cuaca !", vbInformation, "Informasi"
    End If
End Sub

```

16. Laporan Periodik Data Cuaca

Option Explicit

```

Private Sub Form_Load()
    dtAwal.Value = Date
    dtAkhir.Value = Date
End Sub

```

```

Private Sub cmdCetak_Click()
    cetak_laporan_periodik_cuaca
End Sub

```

```

Private Sub cmdExit_Click()
    Unload Me
End Sub

```

Sub cetak_laporan_periodik_cuaca() **membuat laporan data cuaca**

```

Dim rsglobal As New ADODB.Recordset
Dim rsdetail As New ADODB.Recordset
Dim rstemp As New ADODB.Recordset
Dim PosY As Long
Dim i As Long
Dim j As Long

```

```

Set rsglobal = ExecSQL(GetDSN, "SELECT * FROM cuaca " & _
"WHERE date >=" & Format(dtAwal.Value, "dd-mm-yyyy") & " AND " & _
"date <=" & Format(dtAkhir.Value, "dd-mm-yyyy") & "")

```

```

If rsglobal.RecordCount > 0 Then

```

```

    OpenExcel True
    OpenWorkBook App.Path & "\LaporanPeriodikDataCuaca.xls"
    Set rstemp = ExecSQL(GetDSN, "SELECT * FROM global")

```

```

If rstemp.RecordCount > 0 Then

```

```

    PutExcelValue 1, 1, rstemp!Airport
    PutExcelValue 2, 1, rstemp!alamat & " - " & rstemp!provinsi
    PutExcelValue 3, 1, rstemp!negara

```

```

End If

```

```

PutExcelValue 5, 1, dtAwal.Value & " s.d. " & _
& dtAkhir.Value
PosY = 8

```

```

For j = 0 To rsglobal.RecordCount - 1

```

```

    PutExcelValue PosY, 1, Trim(j + 1) & ". " & Format(rsglobal!Date, "dd/mm/yyyy")
    PutExcelValue PosY, 2, " " & Format(rsglobal!TIME, "hh:mm")
    PutExcelValue PosY, 3, "" & rsglobal!Direction
    PutExcelValue PosY, 4, rsglobal!SPEED
    PutExcelValue PosY, 5, rsglobal!VISIBILITY
    PutExcelValue PosY, 6, rsglobal!weather
    PutExcelValue PosY, 7, rsglobal!clouds
    PutExcelValue PosY, 8, rsglobal!FEET
    PutExcelValue PosY, 9, rsglobal!TEMPERATURE
    PutExcelValue PosY, 10, rsglobal!dewpoint
    PutExcelValue PosY, 11, rsglobal!QNH
    PutExcelValue PosY, 12, rsglobal!UserID

```

```

For i = 1 To 12

```

```

    SetCellFontSize PosY, i, 12
    If i >= 12 And i <= 11 Then SetHorizontalAlignment PosY, i, xlRight
    If j = rsglobal.RecordCount - 1 Then
        SetBorderLine PosY, i, LBottom, Inormal
    End If

```

```

Next i
PosY = PosY + 1
rsglobal.MoveNext

```

```

Next j
  For i = 1 To 12
    SetCellFontSize PosY, i, 12
    SetHorizontalAlignment PosY, i, xlRight
    SetCellFontBold PosY, i
  Next i
  'ExcelPrintPreview
  'CloseWorkBook
  'CloseExcel
  Else
    MsgBox "Tidak Ada Data Cuaca !", vbInformation, "Informasi"
  End If
End Sub

```

17. Laporan Harian Data METAR

Option Explicit

```

Private Sub Form_Load()
  dtAwal.Value = Date
End Sub

```

```

Private Sub cmdCetak_Click()
  cetak_laporan_harian_METAR
End Sub

```

```

Private Sub cmdExit_Click()
  Unload Me
End Sub

```

```

Sub cetak_laporan_harian_METAR() 'membuat laporan data METAR

```

```

  Dim rsglobal As New ADODB.Recordset
  Dim rsdetail As New ADODB.Recordset
  Dim rstemp As New ADODB.Recordset
  Dim PosY As Long
  Dim i As Long
  Dim j As Long

```

```

  Set rsglobal = ExecSQL(GetDSN, "SELECT * FROM metar " & _
    "WHERE date >=" & Format(dtAwal.Value, "dd-mm-yyyy") & " AND " & _
    "date <=" & Format(dtAwal.Value, "dd-mm-yyyy") & """)

```

```

  If rsglobal.RecordCount > 0 Then
    OpenExcel True
    OpenWorkBook App.Path & "\LaporanHarianDataMetar.xls"
    Set rstemp = ExecSQL(GetDSN, "SELECT * FROM global")
    If rstemp.RecordCount > 0 Then
      PutExcelValue 1, 1, rstemp!Airport
      PutExcelValue 2, 1, rstemp!alamat & " - " & rstemp!provinsi
      PutExcelValue 3, 1, rstemp!negara
    End If
    PutExcelValue 5, 1, dtAwal.Value
    PutExcelValue 6, 1, " Frekuensi ATIS " & " : " & rstemp!frekuensi & " MHz "
    PosY = 8
    For j = 0 To rsglobal.RecordCount - 1
      PutExcelValue PosY, 1, Trim(j + 1) & ". " & Format(rsglobal!Date, "dd/mm/yyyy")
      PutExcelValue PosY, 2, "" & Format(rsglobal!TIME, "hh:mm")
      PutExcelValue PosY, 3, "" & rsglobal!metar
      PutExcelValue PosY, 4, "" & rsglobal!UserID
    Next j
    For i = 1 To 8
      SetCellFontSize PosY, i, 8
      If i >= 4 And i <= 7 Then SetHorizontalAlignment PosY, i, xlRight
      If j = rsglobal.RecordCount - 1 Then
        SetBorderLine PosY, i, LBottom, Inormal
      End If
    Next i
    PosY = PosY + 1
    rsglobal.MoveNext
  End If

```

```

Next j
  For i = 4 To 7
    SetCellFontSize PosY, i, 8
    SetHorisontalAlignment PosY, i, xlRight
    SetCellFontBold PosY, i
  Next i
  'ExcelPrintPreview
  'CloseWorkBook
  'CloseExcel
  Else
    MsgBox "Tidak Ada Data Cuaca!", vbInformation, "Informasi"
  End If
End Sub

```

18. Laporan Periodik Data METAR

Option Explicit

```

Private Sub Form_Load()
  dtAwal.Value = Date
  dtAkhir.Value = Date
End Sub

```

```

Private Sub cmdCetak_Click()
  cetak_laporan_periodik_METAR
End Sub

```

```

Private Sub cmdExit_Click()
  Unload Me
End Sub

```

```

Sub cetak_laporan_periodik_METAR() 'membuat laporan data METAR
  Dim rsglobal As New ADODB.Recordset
  Dim rsdetail As New ADODB.Recordset
  Dim rstemp As New ADODB.Recordset
  Dim PosY As Long
  Dim i As Long
  Dim j As Long

  Set rsglobal = ExecSQL(GetDSN, "SELECT * FROM metar " & _
    "WHERE date >=" & Format(dtAwal.Value, "dd-mm-yyyy") & " AND " & _
    "date <=" & Format(dtAkhir.Value, "dd-mm-yyyy") & """)
  If rsglobal.RecordCount > 0 Then
    OpenExcel True
    OpenWorkBook App.Path & "\LaporanPeriodikDataMetar.xls"
    Set rstemp = ExecSQL(GetDSN, "SELECT * FROM global")
    If rstemp.RecordCount > 0 Then
      PutExcelValue 1, 1, rstemp!Airport
      PutExcelValue 2, 1, rstemp!alamat & " - " & rstemp!provinsi
      PutExcelValue 3, 1, rstemp!negara
    End If
    PutExcelValue 5, 1, dtAwal.Value & " s.d. " & _
      & dtAkhir.Value
    PutExcelValue 6, 1, " Frekuensi ATIS " & " : " & rstemp!frekuensi & " MHz "
    PosY = 8
    For j = 0 To rsglobal.RecordCount - 1
      PutExcelValue PosY, 1, Trim(j + 1) & ". " & Format(rsglobal!Date, "dd/mm/yyyy")
      PutExcelValue PosY, 2, "" & Format(rsglobal!TIME, "hh:mm")
      PutExcelValue PosY, 3, "" & rsglobal!metar
      PutExcelValue PosY, 4, "" & rsglobal!UserID
    For i = 1 To 8
      SetCellFontSize PosY, i, 8
      If i >= 4 And i <= 7 Then SetHorisontalAlignment PosY, i, xlRight
      If j = rsglobal.RecordCount - 1 Then
        SetBorderLine PosY, i, LBottom, Inormal
      End If
    Next i
    PosY = PosY + 1
  End If
End Sub

```

```

    rsglobal.MoveNext
Next j
    For i = 4 To 7
        SetCellFontSize PosY, i, 8
        SetHorizontalAlignment PosY, i, xlRight
        SetCellFontBold PosY, i
    Next i
    'ExcelPrintPreview
    'CloseWorkBook
    'CloseExcel
    Else
        MsgBox "Tidak Ada Data Cuaca !", vbInformation, "Informasi"
    End If
End Sub

```

19. Fasilitas Record

```

Private Declare Function mciRekam _
    Lib "winmm.dll" Alias "mciSendStringA" ( _
    ByVal lpstrCommand As String, _
    ByVal lpstrReturnString As Any, _
    ByVal uReturnLength As Long, _
    ByVal hwndCallback As Long) As Long

Private Sub Form_Load()
    Dim i As Integer
    Dim intLoop As Integer
    i = mciRekam("close all", 0&, 0, 0)

    i = mciRekam("open new type waveaudio " _
        & "alias capture", 0&, 0, 0)

    With Me
        .AutoRedraw = True
        .DrawStyle = vbInsideSolid
        .DrawMode = vbCopyPen
        .ScaleMode = vbPixels
        .DrawWidth = 2
        .ScaleHeight = 256
    End With

    txtFile.Text = ""
    Timer2.Interval = 10

    With MMControl1
        .Notify = False
        .Wait = True
        .Shareable = False
        .DeviceType = "WaveAudio"
        .Visible = False
    End With

    cmdBroadcast.Enabled = False
    cmdOpen.Enabled = False

End Sub

Private Sub cmdRecord_Click()
    rekam_suara
End Sub

Private Sub cmdStop_Click()
    stop_rekam_suara
End Sub

Private Sub cmdPlay_Click()
    tes_rekaman_suara
End Sub

```

```

Private Sub cmdBroadcast_Click()
    broadcast_suara
End Sub

Private Sub cmdExit_Click()
    Unload Me
End Sub

Private Sub cmdSaveAs_Click()
    simpan_rekaman
End Sub

Private Sub cmdOpen_Click()
    buka_penyimpanan_rekaman
End Sub

Private Sub Timer1_Timer()
    Dim mssg As String * 255
    Dim msg As String
    Dim i As Integer

    i = mciRekam("status capture channels", mssg, 255, 0)
    If Str(mssg) = 1 Then
        msg = msg & "Channels = 1 (mono)" & vbCrLf
    ElseIf Str(mssg) = 2 Then
        msg = msg & "Channels = 2 (stereo)" & vbCrLf
    End If

    txtStatus.Text = msg
End Sub

Private Sub Timer2_Timer()
    If MMControl1.Mode = mciModeStop Then
        MMControl1.Wait = True
        MMControl1.Command = "Prev"
        MMControl1.Command = "Play"
    End If
End Sub

Sub rekam_suara() 'merekam suara
    i = mciRekam("close all", 0&, 0, 0)
    i = mciRekam("open new type waveaudio alias capture", 0&, 0, 0)
    i = mciRekam("set capture channels 2", 0&, 0, 0)
    i = mciRekam("seek capture to start", 0&, 0, 0)
    i = mciRekam("set capture samplespersec 44100", 0&, 0, 0)
    i = mciRekam("set capture bitspersample 16", 0&, 0, 0)
    i = mciRekam("record capture", 0&, 0, 0)
    cmdStop.Enabled = True
    cmdPlay.Enabled = False
    cmdSaveAs.Enabled = False
    cmdRecord.Caption = "Recording..."
    cmdRecord.Enabled = False
    cmdBroadcast.Enabled = False
    cmdOpen.Enabled = False
    Timer1_Timer
    Timer1.Enabled = True
End Sub

Sub stop_rekam_suara()
    Dim i As Integer
    i = mciRekam("stop capture", 0&, 0, 0)
    cmdSaveAs.Enabled = True
    cmdPlay.Enabled = True
    cmdRecord.Enabled = True
    cmdRecord.Caption = "Start Recording"
    cmdStop.Enabled = False
    Timer1.Enabled = True
End Sub

```

```

Sub tes_rekaman_suara()
Dim i As Integer
    i = mciRekam("play capture from 0", 0&, 0, 0)
End Sub

Sub simpan_rekaman()
Dim i As Integer
    DlgBrowse.CancelError = True
    On Error GoTo ErrHandler1
    DlgBrowse.Filter = "WAV file (*.wav)*.wav"
    DlgBrowse.Flags = &H2 Or &H400
    DlgBrowse.InitDir = GetSetting(App.EXENAME, _
        App.EXENAME, "Default Path")
    DlgBrowse.FileName = GetSetting(App.EXENAME, _
        App.EXENAME, "Default Filename")
    DlgBrowse.ShowSave
    If Len(Dir$(DlgBrowse.FileName)) > 0 Then
        Kill DlgBrowse.FileName
    End If
    i = mciRekam("save capture " & """" _
        & DlgBrowse.FileName & """"", 0&, 0, 0)
ErrHandler1:

cmdOpen.Enabled = True
End Sub

Sub buka_penyimpanan_rekaman()
Dim strFilter As String
    strFilter = "WAV (*.wav)*.wav"
    strFilter = strFilter & "|WAV (*.wav)*.wav"
    CommonDialog1.Filter = strFilter
    CommonDialog1.ShowOpen
    CommonDialog1.DialogTitle = "Buka Data Rekam Suara"
    txtFile.Text = CommonDialog1.FileName
    cmdBroadcast.Enabled = True
End Sub

Sub broadcast_suara()
If Timer2.Enabled Then
    Timer2.Enabled = False
    cmdBroadcast.Caption = "Play"
    MMControl1.Command = "Close"
Else
    Timer2.Enabled = True
    cmdBroadcast.Caption = "Stop"
    MMControl1.FileName = txtFile.Text
    MMControl1.Wait = True
    MMControl1.Command = "Open"
End If

If cmdBroadcast.Caption = "Play" Then
    txtFile.Text = ""
End If
cmdOpen.Enabled = False
cmdSaveAs.Enabled = False
cmdPlay.Enabled = False
cmdRecord.Enabled = True
End Sub

```


20. Modul Database

```

Option Explicit
Public Col As TrueOleDbGrid70.Column
Public Cols As TrueOleDbGrid70.Columns

Function GetServerTime() As Date
    Dim conn As New ADODB.Connection
    Dim rs As New ADODB.Recordset

    Set rs = ExecSQL(GetDSN, "Select Current_TimeStamp as time")
    GetServerTime = rs!TIME
End Function

Function GetServerDate() As Date
    Dim dbData As New ADODB.Recordset
    Set dbData = ExecSQL(GetDSN, "SELECT curdate() as MyDate")
    Date = dbData!Mydate
End Function

Function GetFilter() As String
    Dim tmp As String
    Dim n As Integer
    For Each Col In Cols
        If Trim(Col.FilterText) <> "" Then
            n = n + 1
            If n > 1 Then
                tmp = tmp & " AND "
            End If
            tmp = tmp & Col.DataField & " LIKE '%" & Col.FilterText & "%"
        End If
    Next Col
    GetFilter = tmp
End Function

Function GoFilterClear(objGrid As TDBGrid, pdbData As ADODB.Recordset) As String
    For Each Col In objGrid.Columns
        Col.FilterText = ""
    Next Col
    pdbData.Filter = adFilterNone
    GoFilterClear = ""
End Function

Function GoFilter(objGrid As TDBGrid, pdbData As ADODB.Recordset) As String
On Error GoTo errHandler
    Dim c As Integer
    Dim cFilter As String
    With objGrid
        Set Cols = .Columns
        c = .Col
        .Update
        .HoldFields
        cFilter = GetFilter()
        pdbData.Filter = cFilter
        .Col = c
        .EditActive = True
        .Columns(0).FooterText = GetInfoRecNo(objGrid)
        GoFilter = cFilter
    End With
    Exit Function
errHandler:
    GoFilter = ""
    GoFilterClear objGrid, pdbData
End Function

Public Function GetDSN() As String
    GetDSN = "master"
End Function

```

```

Private Function CheckDataSql(tmpStr As Variant) As String
Dim tStr As String
tStr = Replace(tmpStr, "\", "\\")
tStr = Replace(tStr, "'", "\'")
CheckDataSql = tStr
End Function

```

```

Public Function Insert(DSN As String, tablename As String, ArField As Variant, ArValue As Variant) As Boolean
Dim i As Integer
Dim SQL As String

```

```

SQL = "INSERT INTO " & tablename & " ("
For i = 0 To UBound(ArField)
If i = UBound(ArField) Then
SQL = SQL & ArField(i) & ")"
Else
SQL = SQL & ArField(i) & ","
End If
Next i

```

```

SQL = SQL & " Values ("

```

```

For i = 0 To UBound(ArValue)
If i = UBound(ArValue) Then
SQL = SQL & "'" & CheckDataSql(ArValue(i)) & "'"
Else
SQL = SQL & "'" & CheckDataSql(ArValue(i)) & "',"
End If
Next i

```

```

Dim objData As New ADODB.Connection
objData.CursorLocation = adUseClient
objData.Open "DSN=" & DSN

```

```

On Error GoTo Salah
objData.Execute SQL
objData.Close
Set objData = Nothing
Insert = True

```

```

Exit Function

```

```

Salah:
Insert = False
End Function

```

```

Public Function Update(DSN As String, tablename As String, kondisi As String, ArField As Variant, ArValue As Variant) As Boolean

```

```

Dim i As Integer
Dim SQL As String
On Error GoTo err1
SQL = "UPDATE " & tablename & " SET "
For i = 0 To UBound(ArField)
If i = UBound(ArField) Then
SQL = SQL & ArField(i) & "=" & CheckDataSql(ArValue(i)) & ""
Else
SQL = SQL & ArField(i) & "=" & CheckDataSql(ArValue(i)) & "" & ","
End If
Next i
SQL = SQL + " Where " + kondisi

```

```

Dim objData As New ADODB.Connection
objData.CursorLocation = adUseClient
objData.Open "DSN=" & DSN

```

```

objData.Execute SQL

```

```

objData.Close
Set objData = Nothing
Update = True
Exit Function

```

```

err1:
    Update = False
End Function

Public Function ExecSQL(DSN As String, SQL As String) As ADODB.Recordset
    Dim objData As New ADODB.Connection
    objData.CursorLocation = adUseClient
    objData.Open "DSN=" & DSN

    Set ExecSQL = objData.Execute(SQL)
End Function

Function GetKeterangan(cDSN As String, nmTabel As String, Optional pFieldUserId As String = "ID", Optional
pFieldNama As String = "userid", Optional pValueNama As String = "", Optional cWhere As String = "") As Variant
    Dim rs As New ADODB.Recordset

    Set rs = ExecSQL(GetDSN, "SELECT " & pFieldUserId & _
        " AS hasil FROM " & nmTabel & _
        " WHERE " & pFieldNama & " = " & pValueNama & "")

    If rs.RecordCount > 0 Then
        GetKeterangan = ValidNull(rs!hasil)
    Else
        GetKeterangan = ""
    End If
End Function

```

21. Modul DataGrid

```

Public nFlag As Byte 'display record data

Public Function GetGridOrder(pObject As TDBGrid, ColIndex As Integer, Optional pWithOrderBy As Boolean =
True) As String
    If nFlag = 1 Then
        GetGridOrder = IIf(pWithOrderBy, "order by ", "") + pObject.Columns(ColIndex).DataField
        nFlag = 0
    Else
        GetGridOrder = IIf(pWithOrderBy, "order by ", "") + pObject.Columns(ColIndex).DataField + " desc"
        nFlag = 1
    End If
End Function

Function InitGrid(oName As TDBGrid)
    Dim n
    Dim nHighlight As Long

    With oName
        .MarqueeStyle = dbgHighlightRow
        .HighlightRowStyle.BackColor = RGB(255, 200, 100)
        .HighlightRowStyle.ForeColor = vbBlack
        .ScrollBars = dbgVertical
        .ScrollTrack = True
        .Appearance = dbgTrack3D

        .CaptionStyle.BackColor = RGB(128, 128, 255)
        .CaptionStyle.ForeColor = vbWhite
        .CaptionStyle.Font.Bold = True
        .CaptionStyle.Font.size = 9

        .HeadFont.size = 8
        .HeadFont.Bold = True
        .HeadBackColor = RGB(64, 64, 255)
        .HeadForeColor = vbYellow
        .HeadingStyle.Alignment = dbgCenter

        .ColumnFooters = True
        .FooterBackColor = .HeadBackColor
        .FooterForeColor = .HeadForeColor
        .FooterFont.Bold = True
    End With
End Function

```

```

.FooterFont.size = 8

.Font.size = 8
.EvenRowStyle.BackColor = RGB(212, 212, 255)
.AlternatingRowStyle = True
.RecordSelectors = False

.PrintInfo.PageHeader = "Print Date : " + Format(GetServertime, "dddd - mmmm dd, yyyy")
.PrintInfo.PageFooterFont.Italic = True
.PrintInfo.PageHeaderFont.size = 8
.DeadAreaBackColor = .OddRowStyle.BackColor
For n = 0 To .Columns.Count - 1
    .Columns(n).ButtonHeader = True
    .Columns(n).FooterStyle.ForeColor = vbYellow
    .Columns(n).Font.size = 8
Next
End With
End Function

Function GetInfoRecNo(pDgGrid As TDBGrid) As String
    GetInfoRecNo = "#" + Trim(Str(pDgGrid.ApproxCount))
End Function

Public Function DoRefreshGrid(pGrid As TDBGrid, pData As ADODB.Recordset, vaRatio As Variant)
Dim nBookmark As Single
On Error Resume Next
If Val(ValidNull(pGrid.Bookmark)) = 0 Then
    If pGrid.DataMode = dbgUnboundSt Then
        nBookmark = 0
    Else
        nBookmark = 1
    End If
Else
    nBookmark = pGrid.Bookmark
End If

With pGrid
    .DataSource = pData
    .Refresh
    .MoveFirst
    If pGrid.ApproxCount > 0 And nBookmark - 1 <= pGrid.ApproxCount Then
        .MoveRelative nBookmark - 1
    End If
    .Columns(0).FooterText = GetInfoRecNo(pGrid)
End With

Dim nTotal, n As Integer
With pGrid
If UBound(vaRatio) = 0 Then
    .Columns(0).Width = .Width
Else
    For n = 0 To UBound(vaRatio)
        nTotal = nTotal + vaRatio(n)
    Next
    For n = 0 To UBound(vaRatio)
        .Columns(n).Width = (vaRatio(n) / nTotal) * (.Width - 300)
    Next
End If
End With
End Function

```

22. Modul Utama

Option Explicit

Public Enum ButtonStatus

```

    btAdd = 0
    btUpdate = 1
    btCancel = 2
    btInit = 3
    btDisable = 4
    btDel = 5

```

End Enum

Function ValidNull(ChkValue As Variant) As Variant

```

    If VarType(ChkValue) = vbByte Or VarType(ChkValue) = vbCurrency Or VarType(ChkValue) = vbDecimal Or
    VarType(ChkValue) = vbDouble Or VarType(ChkValue) = vbInteger Or VarType(ChkValue) = vbLong Or
    VarType(ChkValue) = 4 Then
        ValidNull = IIf(IsNull(ChkValue), 0, ChkValue)
    ElseIf VarType(ChkValue) = 19 Then
        ValidNull = IIf(IsNull(ChkValue), 0, ChkValue)
    ElseIf VarType(ChkValue) = vbDate Then
        ValidNull = IIf(ChkValue = CDate("1900/01/01"), "", ChkValue)
    ElseIf VarType(ChkValue) = vbString Then
        ValidNull = IIf(IsNull(ChkValue), " ", ChkValue)
    ElseIf VarType(ChkValue) = vbBoolean Then
        ValidNull = IIf(IsNull(ChkValue), 0, IIf(ChkValue, 1, 0))
    End If
End Function

```

Function InitForm(pForm As Object)

```

Dim objForm As Object
On Error Resume Next
For Each objForm In pForm.Controls
    If (TypeOf objForm Is TDBGrid) Then
        InitGrid objForm
        objForm.Font.Name = "tahoma"
        objForm.Font.size = "8"
    End If
    If (TypeOf objForm Is TextBox) Then
        objForm.Font.Name = "tahoma"
        objForm.Font.size = "8"
        objForm.Appearance = 0
    End If
    If (TypeOf objForm Is CommandButton) Then
        objForm.Font.Name = "MS Sans Serif"
        objForm.Font.Bold = True
        objForm.Font.size = 8
    End If
Next
End Function

```

Function SetStatus(pButtonStatus As Byte, pObjForm As Form, vaObjKode As Variant, vaEntry As Variant, vaGrid As Variant, VaUnit As Variant)

```

On Error Resume Next
With pObjForm
    Select Case pButtonStatus
    Case 0 'add
        GetNavigator Array(.cmdOk, .cmdCancel), True
        GetNavigator Array(.cmdAdd, .cmdChange, .cmdDel), False
        GetNavigator vaEntry, True
        GetNavigator vaObjKode, True
        GetNavigator VaUnit, False
        GetNavigator vaGrid, False
        vaObjKode(0).SetFocus
        .InitValue
    Case 1 'update
        GetNavigator Array(.cmdOk, .cmdCancel, .cmdDel), True
        GetNavigator Array(.cmdAdd, .cmdChange), False
        GetNavigator vaEntry, True
    
```

GetNavigator vaObjKode, False
 GetNavigator VaUnit, False
 GetNavigator vaGrid, False

Case 2, 3 'cancel / init

GetNavigator Array(.cmdOk, .cmdCancel, .cmdDel), False
 GetNavigator Array(.cmdAdd, .cmdChange), True
 GetNavigator vaObjKode, False
 GetNavigator vaEntry, False
 GetNavigator vaGrid, True

Case 4 'disable

GetNavigator Array(.cmdOk, .cmdCancel, .cmdDel), False
 GetNavigator Array(.cmdAdd, .cmdChange), False
 GetNavigator vaObjKode, False
 GetNavigator VaUnit, False
 GetNavigator vaEntry, False
 GetNavigator vaGrid, True

Case 5 'del

GetNavigator Array(.cmdOk, .cmdCancel, .cmdDel), False
 GetNavigator Array(.cmdAdd, .cmdChange), True
 GetNavigator vaEntry, False
 GetNavigator vaObjKode, False
 GetNavigator VaUnit, False
 GetNavigator vaGrid, True
 vaObjKode(0).SetFocus
 .InitDelete

Case 6 'Load Cuaca

GetNavigator Array(.cmdOk, .cmdCancel), False
 GetNavigator Array(.cmdAdd), True
 GetNavigator vaObjKode, False
 GetNavigator vaEntry, False
 GetNavigator vaGrid, True

Case 7 'cancel cuaca

GetNavigator Array(.cmdOk, .cmdCancel), False
 GetNavigator Array(.cmdAdd), True
 GetNavigator vaObjKode, False
 GetNavigator vaEntry, False
 GetNavigator vaGrid, True

Case 8 'cmdOk cuaca

GetNavigator Array(.cmdAdd), True
 GetNavigator Array(.cmdOk, .cmdCancel), False
 GetNavigator vaEntry, True
 GetNavigator vaObjKode, True
 GetNavigator VaUnit, False
 GetNavigator vaGrid, False

Case 9 'Load METAR

GetNavigator Array(.cmdOk, .cmdCancel), False
 GetNavigator Array(.cmdAdd, .cmdRefresh), True
 GetNavigator vaObjKode, False
 GetNavigator vaEntry, False
 GetNavigator vaGrid, True

Case 10 'cancel METAR

GetNavigator Array(.cmdOk, .cmdCancel), False
 GetNavigator Array(.cmdAdd, .cmdRefresh), True
 GetNavigator vaObjKode, False
 GetNavigator vaEntry, False
 GetNavigator vaGrid, True

Case 11 'cmdOk METAR

GetNavigator Array(.cmdAdd, .cmdRefresh), True
 GetNavigator Array(.cmdOk, .cmdCancel), False
 GetNavigator vaEntry, False
 GetNavigator vaObjKode, False

```

        GetNavigator VaUnit, False
        GetNavigator vaGrid, False

    End Select

End With
End Function

Public Sub GetNavigator(vaObject As Variant, pBoolean As Boolean)
    Dim n As Integer
    On Error Resume Next
        For n = 0 To UBound(vaObject)
            vaObject(n).Enabled = pBoolean
        Next
    End Sub

Function FindValueInArray(pString As String, pArray As Variant) As Boolean
    Dim n As Integer
    FindValueInArray = False
    For n = 0 To UBound(pArray)
        If Trim(pString) = Trim(pArray(n)) Then
            FindValueInArray = True
            Exit Function
        Else
            FindValueInArray = False
        End If
    Next
End Function

Function CheckData(ChkValue, cMsg) As Boolean
    CheckData = True
    If Len(Trim(ChkValue)) = 0 Then
        CheckData = False
        Pesan Array(cMsg)
    End If
    If ChkValue = 0 Then
        CheckData = False
        Pesan Array(cMsg)
    End If
End Function

Sub TabIndex(vaObject As Variant)
    Dim n As Single
    Dim obj As Object
    On Error Resume Next
        For n = 0 To UBound(vaObject) - 1
            Set obj = vaObject(n)
            obj.TabIndex = n
        Next
    End Sub

Function YesNo(vaTanya As Variant) As Boolean
    Dim n As Integer
    Dim cTanya As String
    For n = 0 To UBound(vaTanya)
        cTanya = IIf(n = 0, vaTanya(n), cTanya + vbCr + vaTanya(n))
    Next
    If MsgBox(cTanya, vbOKCancel + vbQuestion + vbDefaultButton1, "Question") = vbOK Then
        YesNo = True
    Else
        YesNo = False
    End If
End Function

Function Pesan(vaPesan As Variant)
    Dim n As Integer
    Dim cPesan As String
    For n = 0 To UBound(vaPesan)
        cPesan = IIf(n = 0, vaPesan(n), cPesan + vbCr + vaPesan(n))
    Next

```

```

    MsgBox cPesan, vbInformation, "Infomation"
End Function

Sub SetMenu(frm As Object, level As Long)
    Dim rs As New ADODB.Recordset
    Dim obj As Object

    'level 1 = supervisor
    If level = 1 Then Exit Sub

    For Each obj In frm.Controls
        If UCase(Mid(obj.Name, 1, 3)) = "MNU" Then
            'membaca level user
            Set rs = ExecSQL(GetDSN, "SELECT * FROM menu WHERE level=" & level & " AND menuname=" &
            obj.Name & "") 'mencari form menu
            If rs.RecordCount > 0 Then
                On Error Resume Next
                If rs!Visible = 1 Then obj.Visible = True Else obj.Visible = False 'menampilkan field form pengelolaan
                informasi
            End If
        End If
    Next

End Sub

Function LeadingZero(S As Long, length As Byte) As String
    Dim st As String
    st = Trim(S)
    While Len(st) < length
        st = "0" + st
    Wend
    LeadingZero = st
End Function

```

23. Modul Print

Option Explicit

```

Dim MyExcel As Excel.Application
Dim MyExcelWorkBook As Excel.Workbook
Dim MyExcelWorksheet As Excel.Worksheet

```

```

Public Enum XllineType
    Inormal = 1
    LDash = -4115
    LDashDot = 4
    LDashDotDot = 5
    LDot = -4118
    LDouble = -4119
    LSlantDashDot = 13
    LNone = -4142
End Enum

```

```

Public Enum XlLinePos
    LDiagonalDown = 5
    LLeft = 7
    LTop = 8
    LBottom = 9
    LRight = 10
End Enum

```

```

Public Enum XlAlignment
    xlCenter = -4108
    xlLeft = -4131
    xlRight = -4152
    xlTop = -4160
    xlBottom = -4107
End Enum

```

```

Public Sub OpenExcel(Optional Show As Boolean = True)

```



```

Set MyExcel = CreateObject("Excel.Application")
MyExcel.Visible = Show
End Sub

```

```

Public Sub OpenWorkBook(Optional PathAndNameXls As String = "", Optional Psw1 As String = "", Optional
Psw2 As String = "")
If PathAndNameXls = "" Then
Set MyExcelWorkBook = MyExcel.Workbooks.Add
Set MyExcelWorksheet = MyExcelWorkBook.Sheets(1)
Else
Set MyExcelWorkBook = MyExcel.Workbooks.Open(PathAndNameXls, , , Psw1, Psw2)
Set MyExcelWorksheet = MyExcelWorkBook.Sheets(1)
End If
End Sub

```

```

Public Sub CloseWorkBook(Optional NoConfirm As Boolean = True)
MyExcelWorkBook.Saved = NoConfirm
MyExcelWorkBook.Close
End Sub

```

```

Public Sub CloseExcel()
MyExcel.Quit
Set MyExcel = Nothing
End Sub

```

```

'Public Sub ExcelPrintPreview()
'MyExcel.Visible = True
'MyExcelWorksheet.PrintPreview
'MyExcel.Visible = False
'End Sub

```

```

Public Sub PutExcelValue(y As Long, X As Long, S As String)
MyExcelWorksheet.Cells(y, X).Value = S
End Sub

```

```

Public Sub SetCellFontSize(y As Long, X As Long, size As Long)
MyExcelWorksheet.Cells(y, X).Font.size = size
End Sub

```

```

Public Sub SetCellFontBold(y As Long, X As Long)
MyExcelWorksheet.Cells(y, X).Font.Bold = True
End Sub

```

```

Public Sub SetCellFontItalic(y As Long, X As Long)
MyExcelWorksheet.Cells(y, X).Font.Italic = True
End Sub

```

```

Public Sub SetHorizontalAlignment(y As Long, X As Long, alg As XlAlignment)
MyExcelWorksheet.Cells(y, X).HorizontalAlignment = alg
End Sub

```

```

Public Sub SetBorderLine(y As Long, X As Long, lpos As XlLinePos, ltype As XlLineStyle)
MyExcelWorksheet.Cells(y, X).Borders(lpos).LineStyle = ltype
End Sub

```

LAMPIRAN B (Standar Komunikasi Penerbangan)

International Standards
and Recommended Practices and
Procedures for Air Navigation Services



Annex 10
to the Convention on
International Civil Aviation

Aeronautical Telecommunications

Volume II
Communication Procedures
including those with PANS status

This edition incorporates all amendments adopted by the Council prior to 13 March 2001 and supersedes, on 1 November 2001, all previous editions of Annex 10, Volume II.

For information regarding the applicability of the Standards and Recommended Practices and the Procedures for Air Navigation Services, see Foreword.

Sixth Edition
October 2001

International Civil Aviation Organization

Letter	Word	Approximate pronunciation	
		International Phonetic Convention	Latin alphabet representation
A	Alfa	'ɑː fɑː	<u>AL</u> FAH
B	Bravo	'brɑː vɔː	<u>BRAH</u> VOH
C	Charlie	'tʃɑː li or 'tʃɑː li	<u>CHAR</u> LEE or <u>SHAR</u> LEE
D	Delta	'deltɑː	<u>DELL</u> TAH
E	Echo	'ekɔː	<u>ECK</u> OH
F	Foxtrot	'fɔːkstrɔːt	<u>FOKS</u> TROT
G	Golf	gɔːlf	GOLF
H	Hotel	hoː'tel	HO <u>TELL</u>
I	India	'ɪndiɑː	<u>IN</u> DEE AH
J	Juliett	'dʒuːliɑːt	<u>JEW</u> LEE <u>ETT</u>
K	Kilo	'kiːlɔː	<u>KEY</u> LOH
L	Lima	'liːmɑː	<u>LEE</u> MAH
M	Mike	mɑːk	MIKE
N	November	noʊ'vembɜː	NO <u>VEM</u> BER
O	Oscar	'ɒskɑː	<u>OSS</u> CAH
P	Papa	pɑːpɑː	PAH PAH
Q	Quebec	ke'bek	<u>KEH</u> BECK
R	Romeo	'rɔːmiʊ	<u>ROW</u> ME OH
S	Sierra	si'ɛrɑː	SEE <u>AIR</u> RAH
T	Tango	'tɑːŋɡɔː	<u>TANG</u> GO
U	Uniform	'juːnɪfɔːrm or 'uːnɪfɔːrm	<u>YOU</u> NEE FORM or <u>OO</u> NEE FORM
V	Victor	'vɪktɑː	<u>VIK</u> TAH
W	Whiskey	'wɪski	<u>WISS</u> KEY
X	X-ray	'eks'reɪ	<u>ECKS</u> RAY
Y	Yankee	'jɑːŋki	<u>YANG</u> KEY
Z	Zulu	'zuːluː	<u>ZOO</u> LOO

Note.— In the approximate representation using the Latin alphabet, syllables to be emphasized are underlined.

Note 1.— The pronunciation of the words in the alphabet may vary according to the language habits of the speakers. In order to eliminate wide variations in pronunciation, posters illustrating the desired pronunciation are available from ICAO.

Note 2.— The Spelling Alphabet specified in 5.2.1.3 is also prescribed for use in the Maritime Mobile Service (ITU Radio Regulations, Appendix S14).

<i>aircraft call signs</i>	<i>transmitted as</i>
CCA 238	Air China two three eight
OAL 242	Olympic two four two
<i>flight levels</i>	<i>transmitted as</i>
FL 180	flight level one eight zero
FL 200	flight level two zero zero
<i>headings</i>	<i>transmitted as</i>
100 degrees	heading one zero zero
080 degrees	heading zero eight zero
<i>wind direction and speed</i>	<i>transmitted as</i>
200 degrees 70 knots	wind two zero zero degrees seven zero knots
160 degrees 18 knots gusting 30 knots	wind one six zero degrees one eight knots gusting three zero knots
<i>transponder codes</i>	<i>transmitted as</i>
2 400	squawk two four zero zero
4 203	squawk four two zero three
<i>runway</i>	<i>transmitted as</i>
27	runway two seven
30	runway three zero
<i>altimeter setting</i>	<i>transmitted as</i>
1 010	QNH one zero one zero
1 000	QNH one zero zero zero

5.2.1.4.1.2 All numbers used in the transmission of altitude, cloud height, visibility and runway visual range (RVR) information, which contain whole hundreds and whole thousands, shall be transmitted by pronouncing each digit in the number of hundreds or thousands followed by the word HUNDRED or THOUSAND as appropriate. Combinations of thousands and whole hundreds shall be transmitted by pronouncing each digit in the number of thousands followed by the word THOUSAND followed by the number of hundreds followed by the word HUNDRED.

<i>altitude</i>	<i>transmitted as</i>	<i>Time</i>	<i>Statement</i>
800	eight hundred	0920 (9:20 A.M.)	TOO ZE-RO or ZE-RO NIN-er TOO ZE-RO
3 400	three thousand four hundred	1643 (4:43 P.M.)	FOW-er TREE or WUN SIX FOW-er TREE
12 000	one two thousand		

<i>cloud height</i>	<i>transmitted as</i>
2 200	two thousand two hundred
4 300	four thousand three hundred

<i>visibility</i>	<i>transmitted as</i>
1 000	visibility one thousand
700	visibility seven hundred

<i>runway visual range</i>	<i>transmitted as</i>
600	RVR six hundred
1 700	RVR one thousand seven hundred

5.2.1.4.1.3 Numbers containing a decimal point shall be transmitted as prescribed in 5.2.1.4.1.1 with the decimal point in appropriate sequence being indicated by the word DECIMAL.

Note 1.— The following examples illustrate the application of this procedure:

<i>Number</i>	<i>Transmitted as</i>
100.3	ONE ZERO ZERO DECIMAL THREE
38 143.9	THREE EIGHT ONE FOUR THREE DECIMAL NINE

Note 2.— For identification of VHF frequencies the number of digits used after the decimal point are determined on the basis of the channel spacing (5.2.1.7.3.4.3 refers to frequencies separated by 25 kHz, 5.2.1.7.3.4.4 refers to frequencies separated by 8.33 kHz).

Note 3.— The channelling/frequency pairing relationship for 8.33 kHz and 25 kHz is found in Table 4-1 (bis), Volume V.

5.2.1.4.1.4 PANS.— *When transmitting time, only the minutes of the hour should normally be required. Each digit should be pronounced separately. However, the hour should be included when any possibility of confusion is likely to result.*

Note.— The following example illustrates the application of this procedure when applying the provisions of 5.2.1.2.2:

5.2.1.4.2 Verification of numbers

5.2.1.4.2.1 When it is desired to verify the accurate reception of numbers the person transmitting the message shall request the person receiving the message to read back the numbers.

5.2.1.4.3 Pronunciation of numbers

5.2.1.4.3.1 When the language used for communication is English, numbers shall be transmitted using the following pronunciation:

<i>Numeral or numeral element</i>	<i>Pronunciation</i>
0	ZE-RO
1	WUN
2	TOO
3	TREE
4	FOW-er
5	FIFE
6	SIX
7	SEV-en
8	AIT
9	NIN-er
Decimal	DAY-SEE-MAL
Hundred	HUN-dred
Thousand	TOU-SAND

Note.— The syllables printed in capital letters in the above list are to be stressed; for example, the two syllables in ZE-RO are given equal emphasis, whereas the first syllable of FOW-er is given primary emphasis.

5.2.1.5 Transmitting technique

5.2.1.5.1 PANS.— *Each written message should be read prior to commencement of transmission in order to eliminate unnecessary delays in communications.*

5.2.1.5.2 Transmissions shall be conducted concisely in a normal conversational tone.

Note.— See the language proficiency requirements in the Appendix to Annex 1.

5.2.1.5.3 PANS.— *Speech transmitting technique should be such that the highest possible intelligibility is incorporated in each transmission. Fulfilment of this aim requires that air crew and ground personnel should:*