



Learning By Sample

Sistem Informasi menggunakan Microsoft Access

Soetam Rizky Wicaksono

Learning By Sample

Sistem Informasi dengan Microsoft Access

Soetam Rizky Wicaksono

Diterbitkan oleh : CV. Seribu Bintang



Learning By Sample

Sistem Informasi dengan Microsoft Access

Penulis : Soetam Rizky Wicaksono
Editor / Tata Letak : Seribu Bintang Team
Desain Sampul : Seribu Bintang Team

Publisher / Penerbit :

CV. Seribu Bintang

Malang – Jawa Timur - Indonesia

website: www.SeribuBintang.co.id

email : info@seribubintang.co.id

FB : www.fb.com/cv.seribu.bintang

ISBN: 978-602-52194-2-9

Agustus 2018

Copyright @ Seribu Bintang

〔 Untuk istri dan anak-anakku, pelipur jiwa di kala lara dan duka 〕

〔 Kekayaan adalah merasa bahagia dan bersyukur dengan apa yang telah kita miliki
Kemiskinan adalah merasa sedih dan mengeluh dengan apa yang telah kita miliki 〕

Daftar Isi

Daftar Isi	i
Daftar Gambar	iii
Pendahuluan.....	1
Pengantar Microsoft Access	5
Microsoft Office ?	6
Microsoft Access	9
Interface Microsoft Access	12
Pengenalan Database.....	15
Database Relasional	19
Studi Kasus: Sistem Informasi Katering.....	25
Pengantar	26
Pembuatan Database dan Tabel	27
Pembuatan Form	46
Form Setting	48
Form Jenis Menu.....	57
Form Paket Menu.....	70
Form Transaksi Pemesanan	87
Form Pelunasan	116

Pembuatan Report.....	127
Laporan Daftar Menu	129
Laporan Transaksi Per Periode.....	135
Navigasi Aplikasi.....	147
Daftar Pustaka	157
Glosarium	159

Daftar Gambar

Gambar 1. Tampilan Awal Microsoft Access	13
Gambar 2. Menu Awal	13
Gambar 3. Menu Pembuatan Tabel.....	14
Gambar 4. Tampilan Menu Form.....	14
Gambar 5. Level Abstraksi Data Database	20
Gambar 6. Hirarki Database	21
Gambar 7. Contoh Struktur Tabel.....	22
Gambar 8. Contoh Relasi Antar Tabel	24
Gambar 9. Template Database	28
Gambar 10. Penamaan Database	29
Gambar 11. Tabel Kosong	29
Gambar 12. Table Design	30
Gambar 13. Struktur Tabel Setting	31
Gambar 14. Icon Primary Key.....	31
Gambar 15. Icon Table View.....	32
Gambar 16. Isian Data Tabel Setting.....	32
Gambar 17. Struktur Tabel Jenis.....	33
Gambar 18. Struktur Tabel Customer	34
Gambar 19. Struktur Tabel Menu	34
Gambar 20. Tipe Data Lookup Wizard	35
Gambar 21. Opsi Wizard	36
Gambar 22. Pilihan Field Lookup	36
Gambar 23. Struktur Tabel Paket	38

Gambar 24. Struktur Tabel Detail Paket	38
Gambar 25. Struktur Tabel Transaksi.....	40
Gambar 26. Opsi Default Value	41
Gambar 27. Opsi Validation Rule.....	42
Gambar 28. Opsi Default Value 0	42
Gambar 29. Struktur Tabel Detail Transaksi.....	43
Gambar 30. Icon Relationship	44
Gambar 31. Relasi Tabel.....	45
Gambar 32. Opsi Form Wizard	48
Gambar 33. Pemilihan Tabel Untuk Form.....	49
Gambar 34. Judul Form Setting	50
Gambar 35. Icon Button	51
Gambar 36. Opsi Tombol Save	52
Gambar 37. Pengisian Button Save.....	52
Gambar 38. Pembuatan Button Batal.....	53
Gambar 39. Tampilan Desain Awal Form Setting	54
Gambar 40. Icon View.....	55
Gambar 41. Hasil Form Setting.....	56
Gambar 42. Setting Tabel Form Menu.....	58
Gambar 43. Modifikasi Form Menu.....	58
Gambar 44. Pilihan Navigasi Record	59
Gambar 45. Opsi First Record.....	60
Gambar 46. Opsi Previous Record	60
Gambar 47. Opsi Next Record	61
Gambar 48. Opsi Last Record	61
Gambar 49. Button Navigasi.....	62
Gambar 50. Opsi Align Button	62
Gambar 51. Opsi Ukuran Button	63

Gambar 52. Button Add	64
Gambar 53. Judul Button Add	64
Gambar 54. Opsi Button Save.....	65
Gambar 55. Desain Akhir Form Menu	66
Gambar 56. Hasil Form Menu.....	66
Gambar 57. Uji Coba Form Menu	67
Gambar 58. Uji Coba Checkbox	69
Gambar 59. Desain Form Paket Menu	73
Gambar 60. Tombol Form Paket.....	73
Gambar 61. Navigasi Button.....	74
Gambar 62. Icon Combobox.....	74
Gambar 63. Layout Combobox	75
Gambar 64. Opsi Combobox	75
Gambar 65. Pilihan Field Combobox.....	76
Gambar 66. Opsi Row Source	77
Gambar 67. Embedded Macro	77
Gambar 68. Icon Subform	79
Gambar 69. Opsi Subform	80
Gambar 70. Field Subform	80
Gambar 71. List Subform.....	81
Gambar 72. Icon Textbox	83
Gambar 73. Opsi Caption	83
Gambar 74. Opsi Control Source	84
Gambar 75. Opsi Locked	85
Gambar 76. Hasil Form Paket Menu	86
Gambar 77. Pemilihan Field Form Transaksi	89
Gambar 78. Desain Header Form Transaksi	89
Gambar 79. Opsi Enabled dan Locked	90

Gambar 80. Default Value dengan tipe Date	91
Gambar 81. Tombol Atas Form Transaksi	92
Gambar 82. Icon Listbox.....	93
Gambar 83. Opsi Listbox	94
Gambar 84. Field Listbox.....	95
Gambar 85. Tampilan Listbox.....	96
Gambar 86. Opsi Nama Listbox	96
Gambar 87. Opsi Isian Listbox Manual.....	97
Gambar 88. Hasil Listbox Kedua	98
Gambar 89. Opsi Nama Listbox Kedua.....	98
Gambar 90. Button Tambah	99
Gambar 91. Opsi On Click.....	99
Gambar 92. Opsi Code Builder	100
Gambar 93. Button Hapus.....	102
Gambar 94. Label Total	104
Gambar 95. Opsi Nama Label	104
Gambar 96. Desain Form Transaksi	105
Gambar 97. Icon View Code	106
Gambar 98. Icon Button	107
Gambar 99. Opsi Nama Button.....	108
Gambar 100. Penempatan Button Simpan	108
Gambar 101. Opsi Code Builder	109
Gambar 102. Opsi Add New	113
Gambar 103. Opsi Teks New	113
Gambar 104. Desain Akhir Form Transaksi	114
Gambar 105. Uji Coba Form Transaksi.....	115
Gambar 106. Icon Query Design.....	116
Gambar 107. Icon Show Table.....	117

Gambar 108. Relasi Transaksi.....	118
Gambar 109. Opsi Field Query.....	119
Gambar 110. Field Form.....	120
Gambar 111. Opsi Combo Form	121
Gambar 112. Pemilihan Query	121
Gambar 113. Pemilihan Field	122
Gambar 114. Desain Form Pelunasan.....	124
Gambar 115. Penempatan Button	125
Gambar 116. Teks Button Simpan	125
Gambar 117. Uji Coba Form Pelunasan	126
Gambar 118. Query Design Report.....	130
Gambar 119. Pemilihan Field Query	131
Gambar 120. Pilihan Field Query	131
Gambar 121. Report Wizard.....	132
Gambar 122. Label SUM.....	133
Gambar 123. Preview Report	134
Gambar 124. Query Design Report Transaksi	136
Gambar 125. Field Report	136
Gambar 126. Opsi Field Report	138
Gambar 127. Desain Form untuk Report	139
Gambar 128. Isian Format Tanggal	139
Gambar 129. Opsi Subreport.....	140
Gambar 130. Opsi Penggunaan Query.....	141
Gambar 131. Field Tabel Setting.....	142
Gambar 132. Pengaturan Awal.....	143
Gambar 133. Isian Tanggal	143
Gambar 134. Format Short Date	144
Gambar 135. Opsi Nama Textbox Report	144

Gambar 136. Button Preview	145
Gambar 137. Desain Laporan Transaksi	146
Gambar 138. Uji Coba Laporan Transaksi	146
Gambar 139. Menu Navigasi	147
Gambar 140. Form Navigasi Awal.....	148
Gambar 141. Desain Menu.....	150
Gambar 142. Desain Akhir Form Navigasi	150
Gambar 143. Setting Aplikasi	151
Gambar 144. Opsi Navigasi	153
Gambar 145. Opsi Display	154
Gambar 146. Menu Ekspor Package	155
Gambar 147. Kompilasi Database.....	156

Pendahuluan

Masih banyak orang yang mengabaikan Microsoft Access sebagai perangkat lunak database yang dapat dipergunakan untuk keperluan komersial. Khususnya di Indonesia yang lebih banyak memilih database komersial dengan kapasitas “raksasa” seperti SQL Server ataupun Oracle (meskipun diperoleh dengan cara yang ilegal).

Padahal untuk kepentingan pemrosesan data yang sederhana ataupun pembelajaran database dasar, Microsoft Access merupakan “senjata” yang harus diperhitungkan. Selain keberadaannya yang umumnya melekat pada Microsoft Office, Microsoft Access juga cukup ringan dijalankan, sehingga dapat diaplikasikan di komputer ataupun laptop yang memiliki spesifikasi hardware “sederhana”.

Bagi para praktisi UMKM, buku ini dapat dijadikan acuan untuk dapat membuat sistem informasi sederhana. Contoh yang dilampirkan disusun sedemikian rupa dengan model *learning by sample*, sehingga dapat dipahami dan langsung dipraktekkan secara cepat, bahkan oleh orang awam IT sekalipun.

Sedangkan bagi akademisi, khususnya mahasiswa, buku ini dapat menjadi sarana pembelajaran dasar yang sederhana. Yang pertama

untuk pembelajaran awal teori sekaligus praktek basis data atau database. Penggunaan Microsoft Access yang ringan dan mudah dapat menjadi alternatif praktikum bagi mahasiswa ataupun kampus yang memiliki spesifikasi perangkat keras minimal.

Yang kedua, dapat pula dijadikan sarana pembelajaran untuk Sistem Informasi Manajemen (atau mahasiswa non TI yang didalamnya terdapat mata kuliah pengantar sistem informasi). Sebab dengan penggunaan Microsoft Access secara langsung (dalam bentuk studi kasus) dapat mempercepat pemahaman praktek bagi mahasiswa.

Sedangkan yang terakhir dapat pula dijadikan perangkat pembelajaran bagi mahasiswa non TI yang melaksanakan kerja praktek ataupun PKL. Sebab contoh yang terdapat dalam buku ini dapat sekaligus dimodifikasi dengan mudah sesuai dengan kebutuhan yang ada.



Kamu tidak perlu menjadi luar biasa untuk memulai, tapi kamu harus memulai untuk menjadi luar biasa.



Pengantar Microsoft Access

Tujuan:

- Memahami dasar dari Microsoft Office
- Memahami dasar dari Microsoft Access, baik dari segi tampilan maupun penggunaan awal

Microsoft Office ?

.....

Microsoft Office hingga saat ini diklaim sebagai aplikasi perkantoran dengan pengguna terbanyak di dunia. Pengguna resmi (bukan bajakan), pada tahun 2016 diklaim telah mencapai 12 milyar oleh Microsoft (sumber: <https://www.windowscentral.com/there-are-now-12-billion-office-users-60-million-office-365-commercial-customers>, diakses pada tanggal 2 Juli 2018).

Hal ini tentu berpengaruh terhadap keilmuan dan kepakaran tentang penggunaan Microsoft Office di kalangan masyarakat TI maupun non TI (kerap disebut sebagai pengguna awam). Bahkan di beragam jenjang sekolah, mata pelajaran TI yang mengikutsertakan materi Microsoft Office telah jamak diajarkan.

Microsoft Office yang konon muncul pertama kali pada tahun 1990 dengan nama Office 1.0, baru mengemuka secara masif pada saat diperkenalkan hampir bersamaan dengan kemunculan Windows 95 dengan nama yang sama yakni Office 95. Semenjak kehadiran Office 95, maka para pesaing terdekatnya saat itu seperti Wordstar, Chi Writer dan lainnya mulai terdesak keberadaannya.

Microsoft Office sendiri pada dasarnya memiliki fungsi untuk mempermudah pekerjaan umum yang terdapat di perkantoran, misal: pengetikan, pembuatan tabel, perhitungan, kalkulasi hingga ke pembuatan sistem informasi sederhana. Namun demikian, perkembangan Microsoft Office tidak hanya sampai ke sana, di versi terbaru Microsoft Office, paket yang disertakan telah merambah ke fungsi lain yang lebih kompleks seperti adanya Microsoft Project, Visio hingga Sharepoint.

Microsoft Office juga memiliki keunggulan yang sulit disaingi oleh pesaingnya yakni integritas antar aplikasi serta pemrograman yang *menempel* di dalam aplikasi yang disebut sebagai VBA (Visual Basic for Application) atau kerap disebut sebagai macro. VBA yang cenderung lebih mudah dan lebih "bersahabat" bagi pemula maupun kalangan nonTI. Hal ini dikarenakan VBA lebih bersifat struktural dan bahkan dapat disimulasikan menggunakan macro recorder sebelum memulai pengetikan listing.

VBA inilah yang kemudian berkembang menjadi "senjata" yang sakti dalam pengembangan dokumen Microsoft Office menjadi sebuah sistem informasi level sederhana. Bahkan untuk Microsoft Excel, VBA dapat

menjadikan dokumen Excel sebagai perangkat analisis data yang sangat kompleks, mulai dari penyelesaian masalah statistik hingga ke analisis tingkat lanjut.

Begitu pula dengan keluarga Microsoft Office lainnya seperti Microsoft Word ataupun Power Point, bahkan di anggota keluarga lain seperti Sharepoint hingga Microsoft Access. Di dalam buku ini, pembahasan Microsoft Access hingga menjadi sebuah sistem informasi juga menggunakan VBA. Namun demikian, penggunaan VBA tersebut tidaklah serumit yang dibayangkan.

Bahkan bagi para pemula sekalipun, penjelasan yang ada di dalam buku ini diharapkan dapat dengan mudah dipahami serta dipraktekkan. Sehingga pada akhirnya benar-benar menjadi sebuah sistem informasi yang diharapkan.

Microsoft Access

Microsoft Access merupakan keluarga dari Microsoft Office yang memiliki fungsi sebagai DBMS (*Database Management System*). Sebagian kalangan menyebut bahwa Microsoft Access tidak layak disebut sebagai DBMS (penjelasan tentang database dapat ditelaah di bab berikutnya). Namun demikian, fungsi yang ditawarkan oleh Microsoft Access telah mampu memenuhi syarat dari DBMS.

Microsoft Access pertama kali dirilis pada tahun 1990 yang kemudian diperbaiki kembali pada tahun 1993. Rilis pertama yang disebut sangat penuh dengan bug ini, telah menggunakan Access Basic yang pada akhirnya berevolusi menjadi VBA.

Evolusi menjadi VBA terjadi pada saat rilis Microsoft Office 95 yang bersamaan dengan rilis Windows 95 yang fenomenal. Didalamnya termasuk Microsoft Access yang juga pada akhirnya menggunakan VBA didalamnya sebagai perangkat bantu dalam membuat sebuah sistem informasi.

Pada saat Microsoft Office 2000 dirilis, Microsoft Access yang termasuk dalam paket didalamnya

mengalami beberapa perubahan. Salah satu yang paling signifikan adalah adanya fasilitas untuk melakukan konversi database ke dalam format HTML sehingga dapat diakses melalui browser. Selain itu, Microsoft Access pada saat itu mulai dilengkapi dengan beberapa fasilitas security dan backup sederhana.



Evolusi menjadi VBA terjadi pada saat rilis Microsoft Office 95 yang bersamaan dengan rilis Windows 95 yang fenomenal.

Fitur yang sangat dinanti pada versi 2000 adalah penggunaan SQL yang lebih baik. Sehingga bagi para pengembang perangkat lunak yang ingin membuat sistem informasi sederhana dapat menerapkan SQL secara lebih kompleks mulai versi 2000. Bahkan pada versi berikutnya terdapat fasilitas untuk melakukan ekspor ke SQL Server jika data yang dibutuhkan semakin besar.

Pada versi ini pula dimulai era *client server* pada Microsoft Access sehingga mendukung penggunaan multi user. Hal ini juga berpengaruh terhadap perkembangan Microsoft Access di versi berikutnya,

sehingga sistem informasi yang dihasilkan oleh Microsoft Access dapat "naik kelas" ke level menengah.

Berikutnya pada saat rilis Microsoft Office 2007, terjadi perubahan format file pada Microsoft Access, yang sebelumnya memiliki format MDB, menjadi ACCDB. Perubahan ini juga mencakup adanya fasilitas report viewer hingga penyimpanan ke dalam format PDF untuk laporan yang dihasilkan serta keamanan directory.

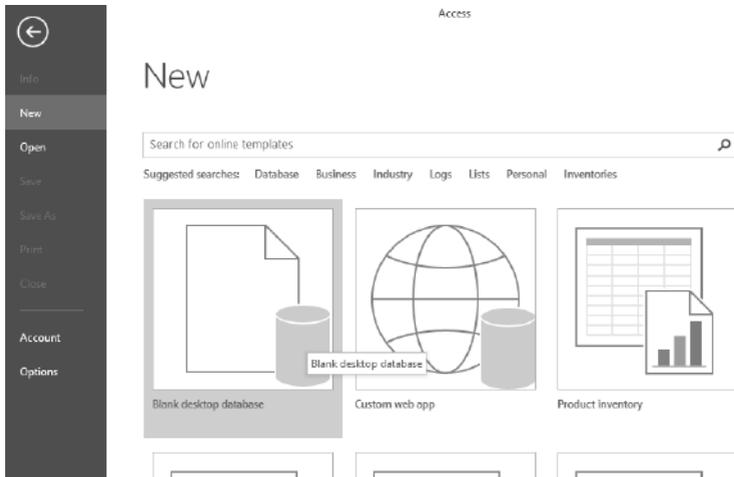
Beberapa perubahan lainnya termasuk ditiadakannya *switchboard* serta penambahan fasilitas untuk melakukan impor dan ekspor data dari database berbeda. Sehingga memudahkan pengembang sistem untuk melakukan *upgrade* ataupun *downgrade* dari sistem yang sudah ada.

Interface Microsoft Access

Bagi Anda yang telah familiar dengan keluarga aplikasi Microsoft Office lain seperti Microsoft Word atau Microsoft Excel, maka membuka Microsoft Access menjadi sebuah hal yang "sama". Baik dalam tampilan awal ataupun menu secara umum (File, Exit, Edit dan lainnya).

Namun demikian, dikarenakan tujuan dari Microsoft Access yang berbeda dengan aplikasi Office lainnya, maka tampilan awal untuk pembuatan dokumen/database berbeda.

Berikut adalah contoh tampilan pada saat Microsoft Access pertama kali dibuka. Sebagai catatan, bahwa tampilan ini bisa berbeda untuk tiap versi Microsoft Access yang dimiliki.



Gambar 1. Tampilan Awal Microsoft Access

Kemudian pada saat awal pembuatan database, maka menu yang tersedia adalah seperti pada gambar 2.



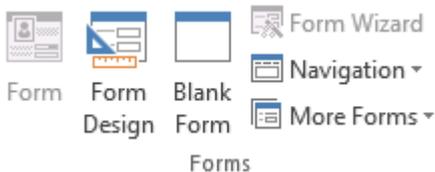
Gambar 2. Menu Awal

Langkah pertama di dalam pembuatan database adalah dengan memulai pembuatan tabel. Pada saat tabel pertama dibuat, maka kita fokus



Gambar 3. Menu Pembuatan Tabel

Menu lain yang wajib diketahui sebelum lebih lanjut mempelajari buku ini adalah menu pembuatan form. Di dalam rangkaian pembuatan form, terdapat mode design yang mensyaratkan pembuatan form dengan menggunakan template yang sudah tersedia, atau dengan menggunakan mode blank yang berarti bahwa pembuatan form diawali dengan tampilan kosong.



Gambar 4. Tampilan Menu Form

Pengenalan Database

Tujuan:

- Memahami konsep dasar database
- Memahami tentang tabel dan field sebagai bagian dari database

Konsep database (atau seringkali diterjemahkan dalam bahasa Indonesia menjadi basis data, tetapi untuk kepentingan kemudahan dalam buku ini tetap dipertahankan istilah database) yang ada dalam buku ini tidak menjelaskan secara rinci tentang asal muasal serta teori didalamnya. Tetapi lebih dipentingkan mengenai pemahaman serta penggunaan dari konsep yang diimplementasikan.

Database diasumsikan tersedia sejak awal tahun 60-an, saat Charles Bachman dari General Electric menciptakan *Information Data Store*. Lebih seKemudian dikembangkan oleh IBM menjadi sistem informasi dengan menggunakan DBMS pada akhir dekade 1960, lalu dibuat menjadi sistem yang dinamakan SABRE untuk pemesanan tiket pesawat bersama American Airlines. Selanjutnya pada era 1970-an, Edgar Codd dari IBM mengembangkan sebuah kerangka baru yang dinamakan *relational data model* dan menjadi awal dari model database saat ini.

Khususnya untuk bahasa SQL, pada awalnya dimulai pada tahun 1970-an saat IBM mengembangkan *Sequel* sebagai bagian dari proyek yang dinamakan *System R*. Sedangkan SQL sendiri pada perkembangannya menjadi sebuah standard

internasional yang ditetapkan oleh organisasi ANSI (*American National Standards Institute*) dan ISO (*International Organization for Standardization*) sejak tahun 1986. Meski pada perkembangannya tiap vendor memiliki standard sendiri-sendiri untuk SQL yang digunakan di dalam produknya, tetapi secara umum masih mengacu kepada standard yang ada pada ANSI dan ISO.

Sistem database (atau di beberapa buku teori terjemahan disebut sebagai sistem basis data) merupakan koleksi dari file yang saling berelasi dan sebuah set aplikasi yang memungkinkan untuk memodifikasi file tersebut. Tujuan utama dari sistem database adalah menyediakan *view* secara abstrak terhadap pengguna sehingga pengguna tidak lagi perlu untuk mengetahui bagaimana data itu disimpan dan diatur. Sedangkan DBMS (*DataBase Management System*) merupakan sebuah perangkat lunak yang bertujuan untuk mengatur dan mengorganisasi sistem database.

Database sendiri memiliki keuntungan yaitu:

1. Data yang independen, sehingga memisahkan antara data dengan aplikasi dalam penyimpanannya

2. Akses data yang efisien, dengan berbagai teknik yang tersedia dalam perangkat lunak pengolahan database, maka data yang besar dapat diolah dan diakses dengan cepat.
3. Integritas dan keamanan data yang lebih kuat, dengan memanfaatkan akses kontrol dari database itu sendiri.
4. Pemulihan saat terjadi kerusakan yang saat ini sudah dimiliki oleh hampir semua perangkat lunak DBMS, dengan adanya fasilitas backup dan recovery



Sistem database merupakan koleksi dari file yang saling berelasi dan sebuah set aplikasi yang memungkinkan untuk memodifikasi file tersebut.

Yang terakhir, dengan adanya penggunaan database (yang seakan sudah menjadi kewajiban bagi para pengembang aplikasi sistem informasi saat ini), waktu yang dibutuhkan untuk mengembangkan aplikasi sistem informasi tentu saja menjadi lebih cepat.

Database Relasional

.....

Data model atau juga sering disebut pemodelan data bertujuan untuk memenuhi kebutuhan pelanggan dengan mendefinisikan struktur dari data yang merepresentasikan tujuan tertentu dalam bentuk yang logis. Dalam pemodelan ini terdapat tiga level abstraksi data yaitu :

1. *Physical level*

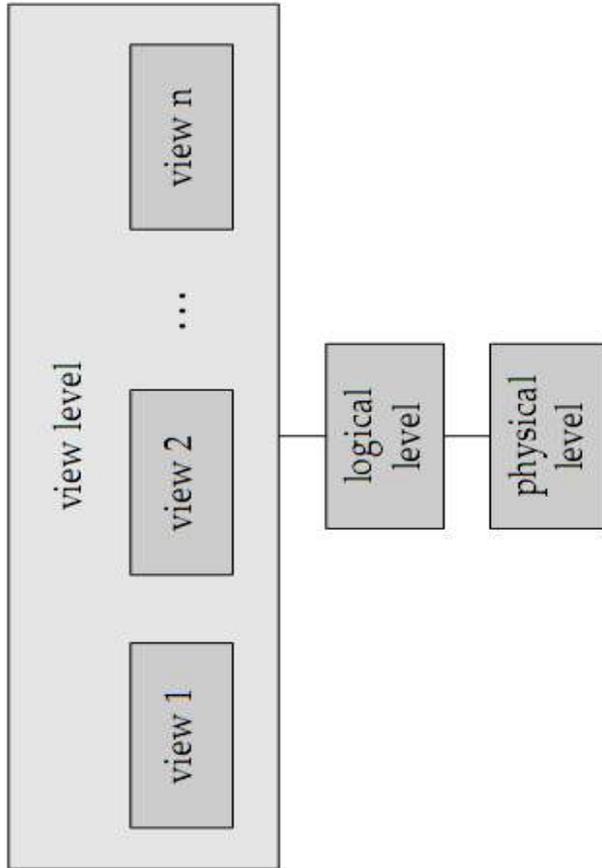
Merupakan level terendah yang menyatakan bagaimana data itu benar-benar disimpan dalam media penyimpanan.

2. *Logical level*

Level ini menyatakan data apa yang disimpan dalam database dan relasi antar data yang disimpan tersebut.

3. *View level*

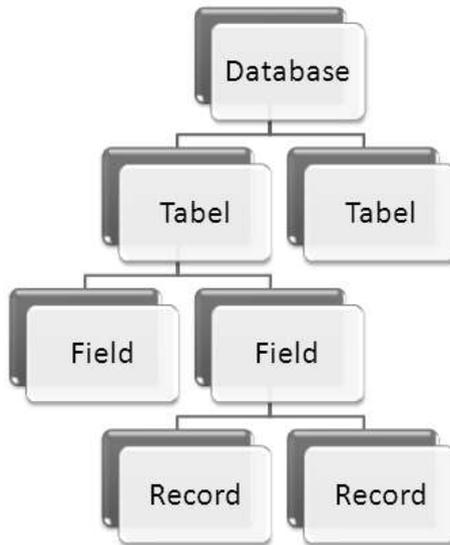
Merupakan level abstraksi data tertinggi yang hanya menampilkan data sesuai yang diinginkan oleh pengguna. Level data ini nantinya menjadi hasil dari keluaran query ataupun fungsi-fungsi.



Gambar 5. Level Abstraksi Data Database

Sebelum mempelajari lebih lanjut mengenai level abstraksi logical level, hendaknya dipahami

terlebih dulu struktur umum dari sebuah database dan unsur yang ada didalamnya. Struktur umum dari sebuah database terlihat seperti pada skema berikut ini:

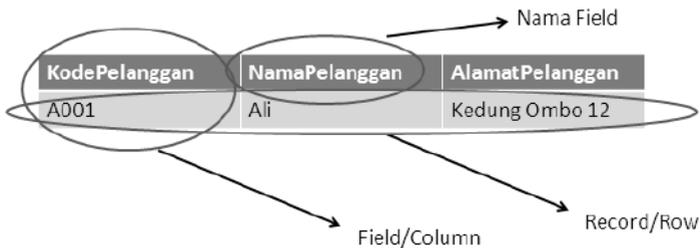


Gambar 6. Hirarki Database

Dari hirarki tersebut, terlihat bahwa sebuah database terdiri dari kumpulan beberapa tabel. Sedangkan sebuah tabel didalamnya memiliki kumpulan field atau juga disebut column. Dan dari tiap field tadi, terdiri dari banyak record yang merupakan

baris data yang nantinya merepresentasikan view level dalam database.

Andaikan dimiliki sebuah tabel untuk merepresentasikan data pelanggan, maka struktur tabel tersebut tampak pada skema berikut:



Gambar 7. Contoh Struktur Tabel

Dalam pemodelan data, sebuah tabel merupakan perwakilan dari *entity* atau obyek yang dikenali oleh pengguna. Dalam sebuah tabel, seharusnya terdapat sebuah field yang bersifat *unique* (tidak terduplikasi) yang menyatakan identitas dari sebuah entity. Field yang bersifat *unique* tersebut dinamakan sebagai *primary key*.

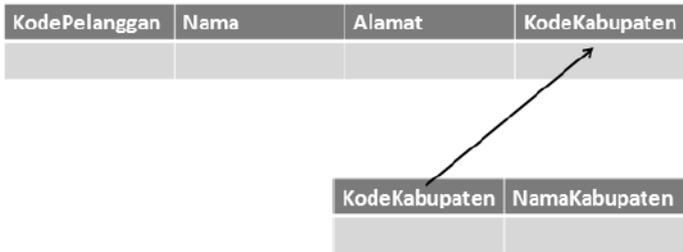
Dalam contoh struktur tabel pelanggan tersebut, seorang pelanggan bisa saja memiliki nama

yang sama, tetapi dengan adanya bantuan field *KodePelanggan* yang bersifat *unique*, maka data seorang pelanggan dapat diidentifikasi lebih jelas dan dapat membedakan antara seorang pelanggan dengan pelanggan yang lain.



Dalam sebuah tabel, seharusnya terdapat sebuah field yang bersifat *unique* (tidak terduplikasi) yang menyatakan identitas dari sebuah entity.

Apabila terdapat lebih dari satu tabel yang saling berelasi, maka antar satu tabel dengan tabel yang lain memiliki minimal satu field yang menjadi penghubung antar tabel tersebut. Sebagai contoh, jika seorang pelanggan bertempat tinggal di sebuah area kabupaten (yang dianggap sebagai entity yang terpisah), sedangkan di sebuah kabupaten tersebut didalamnya dapat memiliki lebih dari satu data pelanggan, maka selain tabel pelanggan juga dibutuhkan sebuah tabel kabupaten dan harus memiliki relasi dengan tabel pelanggan tersebut.



Gambar 8. Contoh Relasi Antar Tabel

Field yang menjadi penghubung atau relasi dari kedua tabel tersebut adalah *KodeKabupaten*. Field *KodeKabupaten* yang terdapat dalam tabel Pelanggan merupakan field yang disebut sebagai field *Foreign Key*.

Dalam relasi antar tabel, sebuah tabel bisa saja memiliki primary key yang terdiri lebih dari satu field. Dan juga foreign key dalam satu tabel yang memiliki relasi lebih dari satu tabel.

Studi Kasus: Sistem Informasi Katering

Tujuan:

- Mempraktekkan kapabilitas Microsoft Access sebagai perangkat untuk implementasi sistem informasi
- Mempraktekkan kapabilitas Microsoft Access sebagai perangkat database

Pengantar

.....

Di dalam studi kasus yang pertama, dibuat sebuah sistem informasi sederhana mengenai katering atau jasa layanan antar masakan harian. Jenis katering yang diambil dalam kasus ini diasumsikan sebagai jenis usaha kecil menengah yang umumnya melakukan layan antar masakan ke rumah maupun kantor.

Asumsi dari layanan harian tersebut adalah dengan melayani pemesanan dari para pelanggan yang melakukan pembayaran dengan cara tunai di saat paket masakan diantar. Sedangkan paket masakan dibentuk dari menu-menu yang sebelumnya telah disediakan oleh pihak perusahaan.

Nantinya, sistem informasi yang dibuat harus dapat melakukan penyimpanan informasi dari pengiriman yang dilakukan oleh katering hingga nantinya mengeluarkan laporan yang menunjukkan hasil transaksi dalam periode tertentu.

Pembuatan Database dan Tabel

.....

Di dalam sistem informasi dengan menggunakan Microsoft Access, langkah pertama yang dilakukan adalah membuat database yang didalamnya berisi tabel-tabel yang saling berelasi satu sama lain. Proses pembuatan tabel tersebut seharusnya didahului dengan analisa dan perancangan sistem informasi yang melibatkan berbagai langkah.

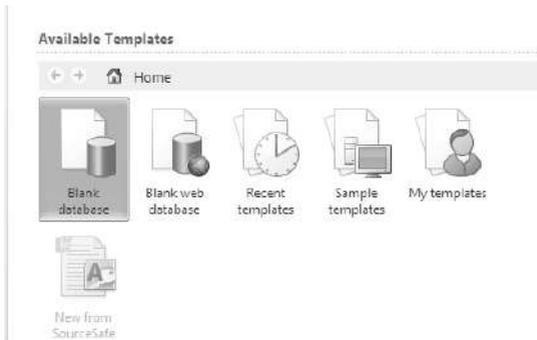


Disarankan agar bagi Anda yang benar-benar merasa pemula untuk membaca berbagai referensi yang didalamnya membahas mengenai analisa dan perancangan sistem informasi. Juga membaca buku mengenai rekayasa perangkat lunak untuk lebih memahami mengenai siklus hidup analisa dan perancangan.

Dalam buku ini, diasumsikan bahwa analisa dan perancangan telah dilakukan sebelumnya, sehingga telah terjadi sebuah rangkaian tabel yang diperlukan di

dalam database. Kini saatnya untuk membuat sistem informasi dari studi kasus yang pertama.

1. Buka aplikasi Microsoft Access, kemudian di dalam kotak dialog yang tersedia pilih opsi *Blank Database* dari template yang tersedia.

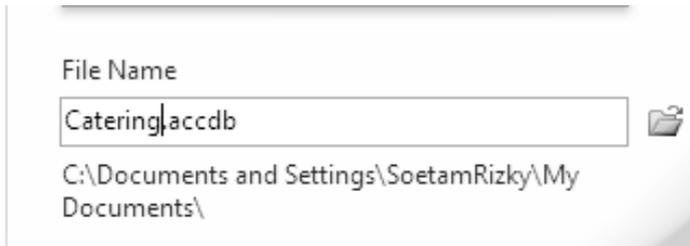


Gambar 9. Template Database



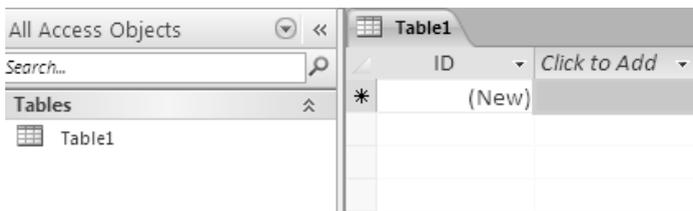
Microsoft Acces memiliki beberapa template yang telah disediakan pada saat instalasi pertama kali dilakukan. Anda dapat memanfaatkan template tersebut di bagian *Sample Templates*. Sedangkan pada template untuk *web database* hanya bisa dimanfaatkan jika Anda memiliki koneksi ke Sharepoint Portal.

2. Berikutnya, di bagian kanan bawah, terdapat isian untuk melakukan proses pemberian nama dari sistem informasi yang dibuat. Berikan nama *Catering* serta tentukan folder yang menjadi wahana bagi sistem informasi tersebut.



Gambar 10. Penamaan Database

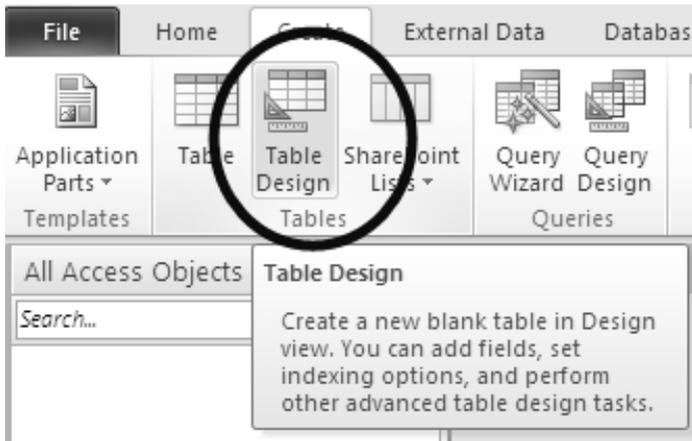
3. Pada saat awal pembuatan database kosong yang nantinya menjadi sistem informasi, maka langsung menuju ke *wizard* pembuatan tabel.



Gambar 11. Tabel Kosong

Di dalam pembuatan sistem informasi ini, kita selalu menggunakan mode *Table Design*

dibandingkan mode *Wizard*. Hal ini dikarenakan seluruh tabel yang dibuat merupakan hasil dari analisa dan perancangan yang telah dibuat sebelumnya. Sehingga langkah selanjutnya adalah memilih opsi *Table Design* yang terletak di dalam bagian ribbon *Create*.



Gambar 12. Table Design



Selain wizard, Microsoft Access juga menyediakan beberapa mode lain dalam pembuatan tabel seperti melakukan impor data dari Outlook atau dari sumber lainnya seperti SQL Server.

4. Selanjutnya, buat tabel pertama dalam sistem informasi yang nantinya berfungsi untuk menyimpan informasi dari perusahaan katering yang menggunakan. Tabel tersebut pada langkah berikutnya diberi nama sebagai tabel *Setting* dan memiliki struktur tabel sebagai berikut:

	Field Name	Data Type
🔑	Nama	Text
	Alamat	Text
	Telp	Text
	Kota	Text

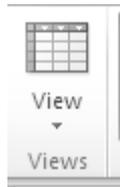
Gambar 13. Struktur Tabel Setting

Di dalam tabel tersebut, terdapat sebuah field yang dijadikan sebagai *primary key* yakni field *Nama*. Untuk menjadikan field tersebut sebagai *primary key*, maka klik pada field yang dimaksud lalu klik ikon *Primary key* yang terletak di bagian kiri atas layar.



Gambar 14. Icon Primary Key

Berikutnya, untuk mendukung pembuatan sistem informasi ini, isikan data contoh pada tabel yang baru saja dibuat dengan melakukan klik pada ikon View yang terletak di bagian kanan atas, sehingga menuju ke mode pengisian data.



Gambar 15. Icon Table View



Anda dapat mengisi data contoh sesuai dengan selera, tetapi perlu diingat bahwa data pada tabel setting seharusnya hanya terdiri dari satu record sebagai pengenalan dari sistem informasi ini.

Setting			
Nama	Alamat	Telp	Kota
Catering ABC	Jl. Tidar 567	0393848484	Malang
*			

Gambar 16. Isian Data Tabel Setting

5. Kemudian, buat tabel-tabel berikutnya dengan langkah pembuatan seperti yang telah dijelaskan pada langkah sebelumnya. Tabel-tabel tersebut adalah :

a. Tabel *Jenis*

Tabel ini nantinya berfungsi untuk menampung data jenis dari sebuah menu, misalkan jenis makanan, minuman atau pencuci mulut (buah dan sebagainya).

	Field Name	Data Type
🔑	Kode Jenis	AutoNumber
	Nama Jenis	Text

Gambar 17. Struktur Tabel Jenis

b. Tabel *Customer*

Seperti tertera pada namanya, tabel ini merupakan penampung data dari pelanggan yang melakukan pemesanan di katering. Perlu diperhatikan bahwa pada field *Aktif* yang memiliki tipe *boolean* atau *Yes/No* merupakan penanda jika seorang pelanggan telah dinyatakan tidak aktif lagi. Hal ini dimaksudkan agar tidak ada proses penghapusan data yang termasuk dalam

kategori *master*, tetapi yang ada adalah proses penonaktifan.

	Field Name	Data Type
🔑	Kode Customer	AutoNumber
	Nama	Text
	Alamat	Text
	Telp	Text
	Aktif	Yes/No

Gambar 18. Struktur Tabel Customer

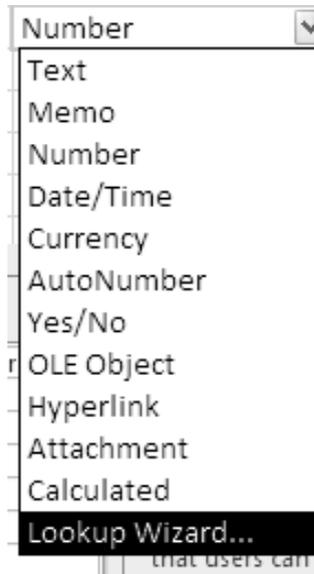
c. Tabel *Menu*

Tabel ini merupakan tabel yang didalamnya menjadi tempat penyimpanan data menu makanan yang dijadikan paket di dalam pengiriman katering untuk pelanggan.

	Field Name	Data Type
🔑	Kode Menu	AutoNumber
	Jenis Menu	Number
	Nama Menu	Text
	Harga	Number
	Aktif	Yes/No

Gambar 19. Struktur Tabel Menu

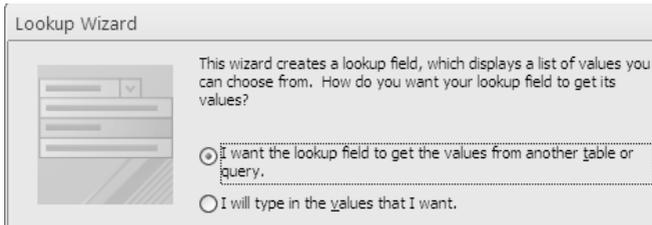
Pada pembuatan tabel ini, field *Jenis Menu* merupakan field yang bersifat *foreign key* atau mengacu kepada tabel lain. Sehingga pada saat Anda melakukan pembuatan field *Jenis Menu*, pilihlah tipe field *Lookup Wizard*.



Gambar 20. Tipe Data Lookup Wizard

Selanjutnya, pada kotak dialog yang tersedia, pilihlah opsi *I want to lookup field.....* Hal ini berarti bahwa kita mengarahkan tipe field *Jenis Menu* ke

sebuah tabel yang sebelumnya tersedia (dalam kasus ini adalah tabel *Jenis Menu*).



Gambar 21. Opsi Wizard

Pada saat Anda selesai memilih opsi tersebut dan menekan tombol *Next*, maka pilihlah kedua field yang tersedia dari tabel *Jenis* ke kotak yang tersedia.



Gambar 22. Pilihan Field Lookup

Langkah berikutnya dapat Anda ikuti hingga kotak dialog tersebut selesai dipilih secara keseluruhan. Dalam kasus pemilihan field yang merupakan referensi dari tabel lain, Microsoft Access secara otomatis melakukan deteksi kepada field yang bertipe

primary key sebagai field yang berelasi (yakni field *Kode Jenis*), dan field lainnya (yaitu *Nama Jenis*) sebagai field yang berfungsi sebagai keterangan dalam referensi. Fungsi dari field yang menjadi referensi tersebut dapat dilihat pada saat pembuatan form.



Perlu diingat bahwa field yang saling mereferensikan satu sama lain tidaklah harus memiliki nama yang sama, dalam hal ini adalah field *Jenis Menu* dari tabel *Menu* dan field *Kode Jenis* dari tabel *Jenis Menu*. Sebab yang terpenting adalah kedua field tersebut harus memiliki tipe data yang sama, bukan nama yang sama.

d. Tabel *Paket*

Di dalam pengiriman katering untuk pelanggan, terdiri dari paket yang dapat berbeda isi menu tiap hari. Satu paket dapat terdiri dari banyak menu dari berbagai jenis, misal : paket Senin terdiri dari menu jenis makanan dan menu jenis minuman.

Sehingga dalam penyimpanan data paket, terdiri dari dua tabel yakni tabel *Paket* dan tabel *Detail Paket* yang memiliki relasi *one to many*.

	Field Name	Data Type
🔑	Kode Paket	AutoNumber
	Nama Paket	Text

Gambar 23. Struktur Tabel Paket

e. Tabel *Detail Paket*

	Field Name	Data Type
🔑	Kode Detail	AutoNumber
	Paket	Number
	Menu	Number
	Harga	Number

Gambar 24. Struktur Tabel Detail Paket

Pada tabel ini terdapat dua field yang menjadi *foreign key* yakni field *Paket* yang mereferensi ke field *Kode Paket* dari tabel *Paket*. Serta field *Menu* yang mereferensi ke field *Kode Menu* dari tabel *Menu*.

Gunakan langkah yang sama seperti pada tabel *Menu* saat mereferensi ke tabel *Jenis Menu* di langkah sebelumnya.

f. Tabel *Transaksi*

Untuk menyimpan data transaksi, diperlukan dua buah tabel yakni tabel *Transaksi* serta tabel *Detail Transaksi*. Kedua tabel tersebut nantinya memiliki relasi *one to many* dan direlasikan melalui field yang memiliki tipe *Lookup Wizard*. Hal ini berarti bahwa dalam tiap transaksi pengiriman, seorang pelanggan dapat memesan banyak paket sekaligus dalam sekali transaksi.

Seperti halnya yang tertera di pembuatan tabel *Paket* dan *Detail Paket*, maka pembuatan tabel *Transaksi* dan *Detail Transaksi* memiliki kemiripan dalam penyusunan struktur tabel sekaligus kemiripan dalam proses pembuatannya.

	Field Name	Data Type
?	NoTransaksi	AutoNumber
	Tanggal	Date/Time
	Tanggal Kirim	Date/Time
	Customer	Number
	Lunas	Yes/No
	Terkirim	Yes/No

Gambar 25. Struktur Tabel Transaksi



Perhatikan pada field yang memiliki tipe *AutoNumber*, ini berarti bahwa field tersebut secara otomatis terisi angka secara berurutan pada saat terjadi proses pengisian data. Keuntungan dari tipe ini adalah field yang bersangkutan selalu berisi angka yang urut sehingga mudah untuk dilacak. Tetapi di lain sisi, jika terjadi proses penghapusan di beberapa record yang posisinya di tengah maka menjadi sebuah anomali dalam data. Misal: jika telah terdapat sepuluh record, sehingga terjadi angka 1 hingga 10, lalu terjadi proses penghapusan di record kelima, maka nomor lima menjadi *hilang* sehingga seakan-terjadi sebuah keanehan dalam data yang ada atau sering juga disebut sebagai anomali.

Khusus pada field *Tanggal* di tabel tersebut, set properti *Default Value* menjadi `=Date()`. Hal ini berarti bahwa pada saat terjadi pengisian tabel *Transaksi*, maka field *Tanggal* secara otomatis terisi dengan tanggal dari sistem operasi (atau tanggal hari ini).

Caption	
Default Value	<code>=Date()</code>
Validation Rule	
Validation Text	

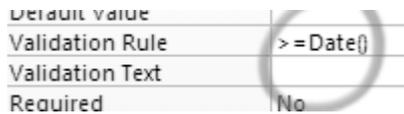
Gambar 26. Opsi Default Value



Teknik seperti ini seringkali dilakukan jika pada suatu field yang membutuhkan sejenis *timestamp* untuk mencatat tanggal saat sebuah transaksi terjadi.

Sedangkan pada field *Tanggal Kirim*, set properti *Validation Rule* menjadi `>=Date()`. Hal ini menandakan bahwa pada field ini hanya boleh diisi dengan data tanggal yang terjadi minimal pada hari ini (sesuai dengan tanggal sistem. Dengan kata lain, bahwa

tanggal pengiriman pada katering ini tidak memperbolehkan adanya *back order* atau pemesanan yang dilakukan setelah pengiriman dilakukan.



Default Value	
Validation Rule	> = Date{}
Validation Text	
Required	No

Gambar 27. Opsi Validation Rule

Terakhir, pada field *Lunas* dan field *Ter kirim* yang sama-sama memiliki tipe boolean atau *Yes/No*, set properti *Default Value* menjadi 0. Hal ini menandakan bahwa kedua field tersebut secara otomatis terisi dengan nilai *No* atau *False* pada saat terjadi proses pengisian data baru.



Format	TRUE/Fals
Caption	
Default Value	0
Validation Rule	

Gambar 28. Opsi Default Value 0

g. Tabel *Detail Transaksi*

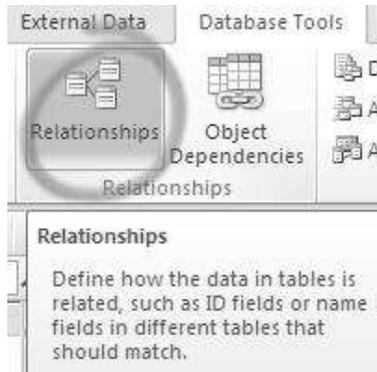
	Field Name	Data Type
🔑	No Detail	AutoNumber
	NoTransaksi	Number
	Paket	Number

Gambar 29. Struktur Tabel Detail Transaksi

Pada tabel ini, terdapat dua field yang mereferensi ke tabel lain (atau memiliki tipe *Lookup Wizard*) yakni:

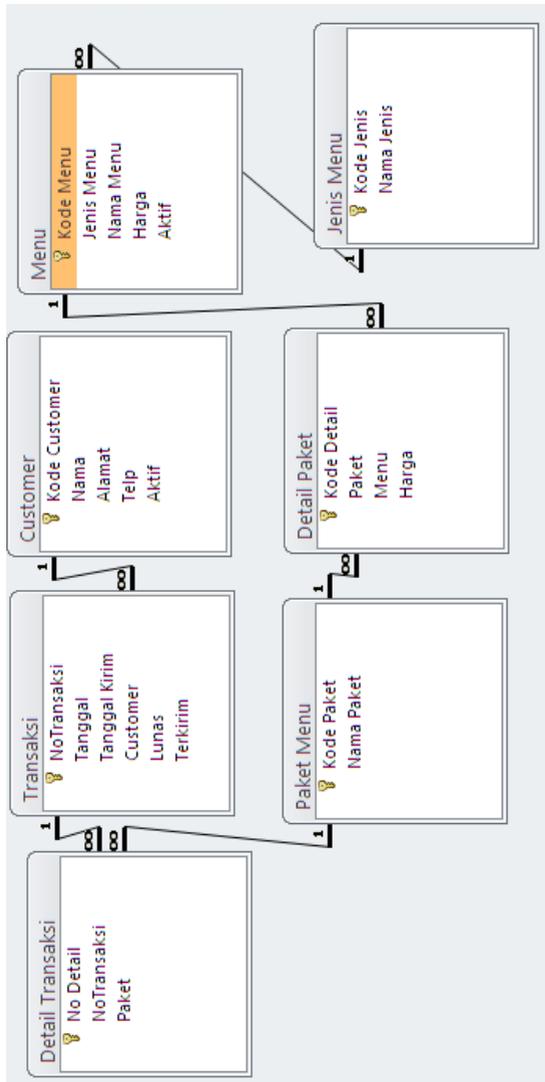
- Field *NoTransaksi* yang bereferensi ke field *NoTransaksi* di tabel *Transaksi*
- Field *Paket* yang bereferensi ke field *Kode Paket* di tabel *Paket*

6. Lalu, untuk memastikan apakah semua tabel yang direferensikan telah benar, klik pada ribbon *Database Tools* dan pilih ikon *Relationships*.



Gambar 30. Icon Relationship

Apabila seluruh langkah pembuatan tabel telah benar, maka tampilan dari relasi tabel di dalam database tampak seperti pada gambar berikut ini:



Gambar 31. Relasi Tabel

Pembuatan Form

.....

Langkah berikutnya dalam implementasi sistem informasi catering ini adalah melakukan pembuatan form. Pembuatan form dalam sistem informasi ini terbagai menjadi tiga kelompok yakni kelompok data master yang sepenuhnya mengandalkan fasilitas *wizard* dari Microsoft Access dengan sedikit modifikasi didalamnya. Kemudian kelompok form untuk transaksi dan laporan (dijelaskan dalam sub bab tersendiri khusus untuk pembuatan laporan) yang didalamnya terdapat beberapa pemanfaatan fitur VBA maupun macro. Dan yang terakhir adalah kelompok navigasi yang merupakan pembuatan *navigation* yang menjadi form utama dalam sistem informasi ini.



Bagi Anda yang telah familiar dengan Microsoft Access versi sebelumnya, kelompok *navigation* mirip dengan fitur *switchboard*.

Dalam proses pembuatan form ini sesungguhnya dapat dilakukan secara tidak berurutan, namun demikian sangat disarankan untuk mengikuti terlebih dulu langkah-langkah yang diterangkan. Terutama bagi Anda para pemula, agar nantinya tidak mengalami kesulitan dalam menyelesaikan sistem informasi ini.

Form Setting

Form yang pertama kali kita buat adalah form yang menangani setting sistem informasi. Form ini merupakan form yang paling sederhana dalam sistem karena hanya melibatkan proses edit. Sebab seperti yang telah disebutkan sebelumnya, isi data dari tabel *Setting* hanya memiliki satu record sebagai tanda pengenalan dari sistem informasi itu sendiri.

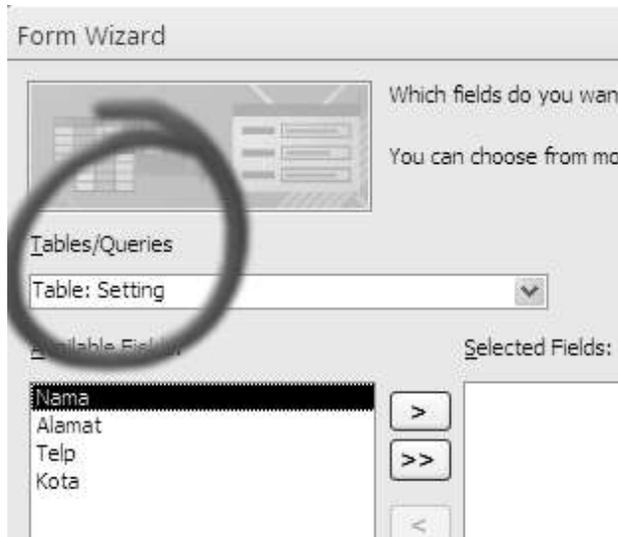
Langkah-langkah pembuatan dari form ini adalah:

1. Klik di ribbon *Create* dan pilih opsi *Form Wizard*.



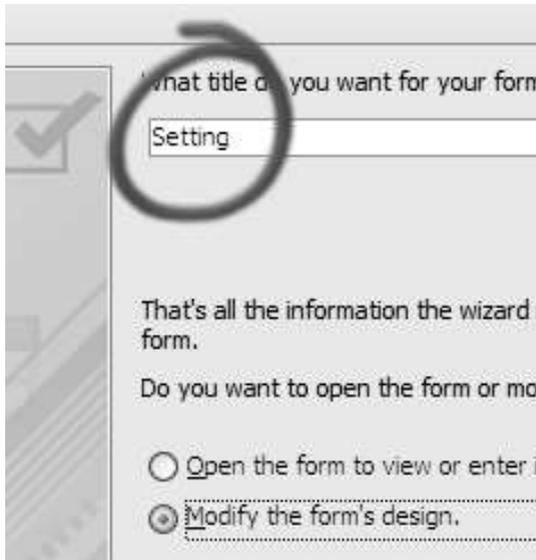
Gambar 32. Opsi Form Wizard

2. Pada kotak dialog yang tersedia, pilih tabel *Setting* pada isian tabel dan arahkan semua field dari tabel tersebut sebagai pilihan.



Gambar 33. Pemilihan Tabel Untuk Form

3. Selanjutnya, pilih opsi default untuk layout hingga sampai pada dialog untuk memilih judul dan aktifitas selanjutnya. Pada dialog tersebut, beri judul untuk form serta pilih opsi untuk memodifikasi form yang diselesaikan.



Gambar 34. Judul Form Setting

4. Pada mode *Design*, tambahkan sebuah Button ke dalam form tersebut. Penambahan Button dilakukan dengan memilih ribbon *Design* dan melakukan klik di toolbar Button. Tempatkan Button di bagian *Header* dari form yang dimaksud.



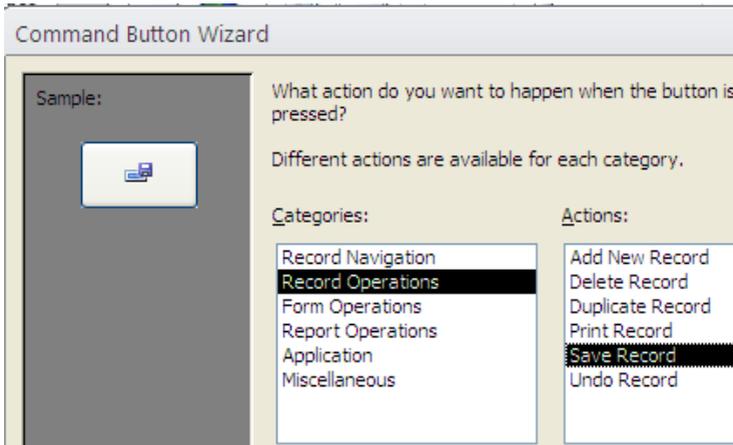
Gambar 35. Icon Button



Hampir semua komponen yang dipilih untuk ditempatkan ke dalam sebuah form pada Microsoft Access menampilkan kotak dialog *wizard* yang menuntun kita untuk memilih aktifitas yang dilakukan. Namun demikian, pada beberapa kasus, kotak dialog tersebut harus ditinggalkan dan diganti dengan pengetikan macro atau listing program di VBA (Visual Basic for Application).

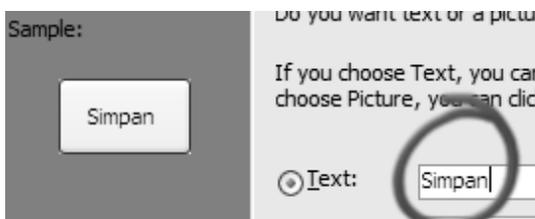
5. Saat Button dipilih dan keluar kotak dialog *Command Button Wizard*, pilih aktifitas *Record Operations* khususnya pada detail *Save Record*. Button ini nantinya menjadi tombol yang

digunakan pengguna untuk menyimpan data hasil edit setting sistem informasi.



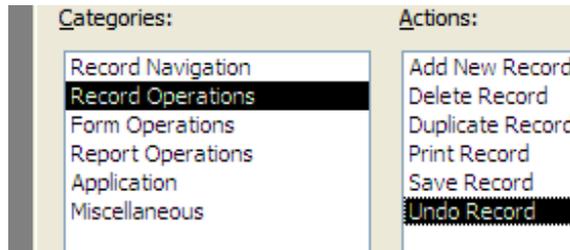
Gambar 36. Opsi Tombol Save

6. Setelah melakukan klik di tombol *Next*, beri judul pada Button tersebut dengan kata *Simpan*. Dan selesaikan hingga kotak dialog tertutup.



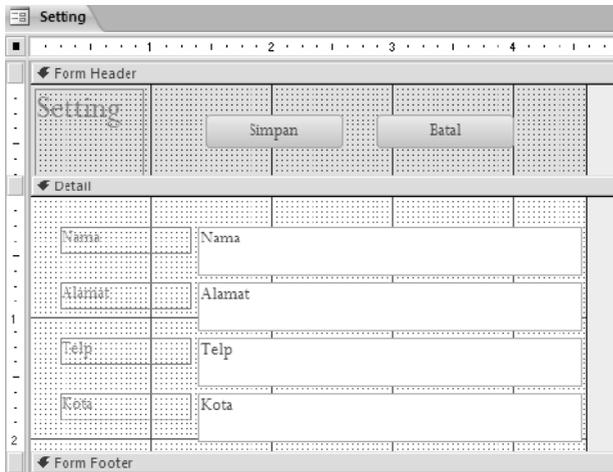
Gambar 37. Pengisian Button Save

7. Kemudian, buat Button yang kedua untuk operasi *Cancel*. Lakukan cara yang sama seperti Button yang pertama, tetapi pada pilihan detail aktifitas, pilih *Undo Record*. Lalu beri judul teks *Batal* pada Button tersebut.



Gambar 38. Pembuatan Button Batal

8. Jika pembuatan kedua Button berjalan lancar, maka form yang tampak dalam mode Design mirip dengan gambar berikut :



Gambar 39. Tampilan Desain Awal Form Setting



Penempatan kedua Button di bagian *Form Header* sebenarnya tidak mutlak dilakukan. Anda juga dapat menempatkan Button di bagian *Detail* maupun di bagian *Footer* sesuai selera. Tetapi sangat lebih baik jika menghindari penempatan di bagian *Detail*, karena di bagian tersebut merupakan bagian dari form yang lebih dikhususkan untuk penempatan data, bukan komponen lainnya.

9. Kini saatnya melakukan uji coba pertama dari form Setting, lakukan pemindahan ke mode *View* dengan memilih ikon yang bertulisan sama di area pojok kanan atas.



Gambar 40. Icon View



Terdapat tiga jenis mode dalam pembuatan form yakni mode View, Layout serta Design. Dua mode yang nantinya paling sering digunakan adalah mode View untuk melakukan testing hasil pembuatan form, serta mode Design yang digunakan untuk memanipulasi desain form.

10. Setelah muncul dalam mode tersebut, cobalah untuk melakukan proses edit data dan menyimpan dalam form tersebut. Dan jika selesai, maka saatnya beralih ke form yang kedua.

The image shows a web-based 'Setting' form. At the top, there is a title bar with the word 'Setting' and a close button. Below the title bar, there are two buttons: 'Simpan' (Save) and 'Batal' (Cancel). The form itself consists of four labeled input fields:

Nama	Catering ABC
Alamat	Jl. Tidar 567
Telp	0393848484
Kota	Malang

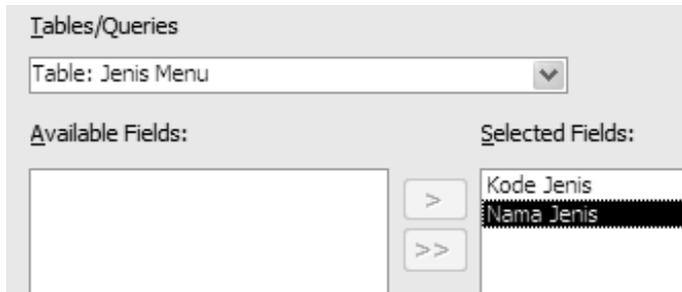
Gambar 41. Hasil Form Setting

Form Jenis Menu

Pembuatan form yang kedua adalah form untuk manipulasi tabel *Jenis Menu*. Berbeda dengan form yang pertama, form ini melakukan aktifitas yang lebih kompleks, yakni untuk navigasi, penambahan data serta proses edit data yang disertai dengan pencarian sederhana.

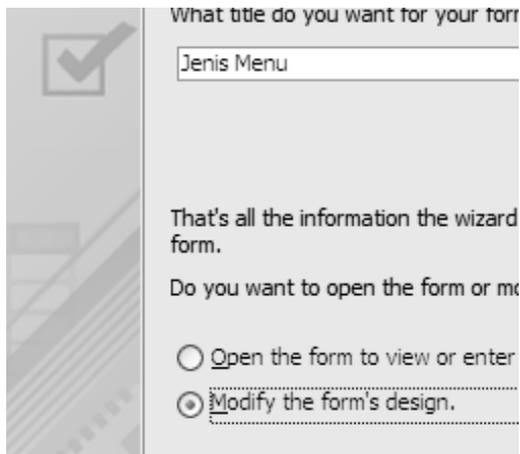
Meski demikian, pembuatan form ini relatif lebih sederhana karena field yang dilibatkan dari tabel *Jenis Menu* hanya berjumlah dua buah. Dan berikut adalah langkah pembuatan untuk form *Jenis Menu*.

1. Seperti halnya pada pembuatan form yang pertama, klik pada ribbon *Create* dan pilih *Form Wizard*.
2. Pada kotak dialog yang terbuka, pilih tabel *Jenis Menu* dan pindahkan seluruh field ke bagian kanan daftar field yang digunakan dalam form.



Gambar 42. Setting Tabel Form Menu

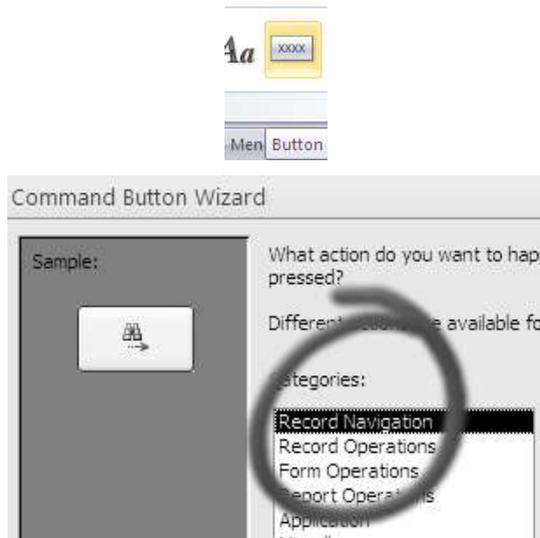
3. Selanjutnya, selesaikan kotak dialog tersebut hingga pada dialog terakhir, pilih opsi untuk melakukan modifikasi terhadap form tersebut.



Gambar 43. Modifikasi Form Menu

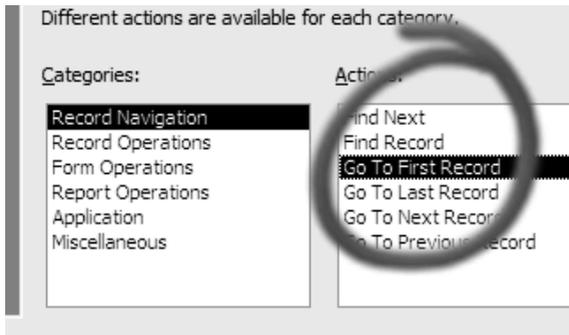
4. Kemudian, tempatkan secara berturut-turut empat Button yang nantinya berfungsi sebagai navigator

dalam form tersebut. Tiap pembuatan Button diawali dengan memilih ribbon *Design* dan klik pada komponen *Button*. Pada saat muncul kotak dialog wizard untuk Button tersebut, pilih categories *Record Navigation*, dan tentukan untuk tiap Button berikut ini :



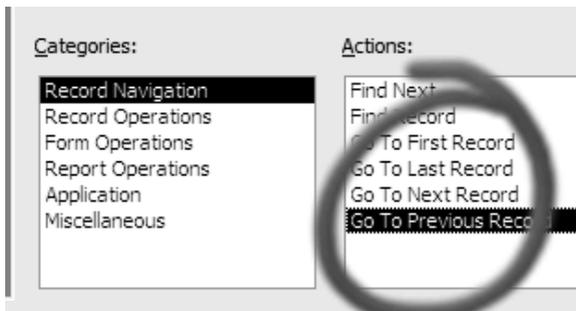
Gambar 44. Pilihan Navigasi Record

a. Button pertama: *First Record*



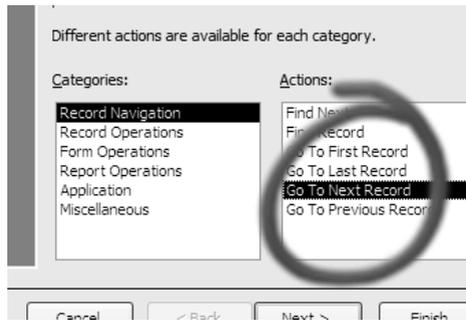
Gambar 45. Opsi First Record

b. Button kedua : *Previous Record*



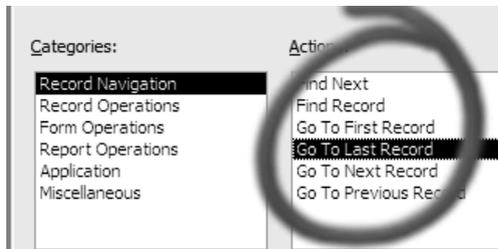
Gambar 46. Opsi Previous Record

c. Button ketiga: *Next Record*



Gambar 47. Opsi Next Record

d. Button keempat: *Last Record*



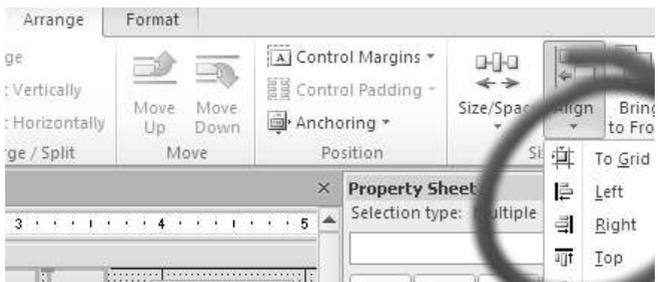
Gambar 48. Opsi Last Record

Biarkan semua pilihan pada kotak dialog berikutnya sebagai default atau tanpa perubahan. Sehingga jika keempat Button tersebut telah terbentuk terlihat seperti pada gambar berikut:



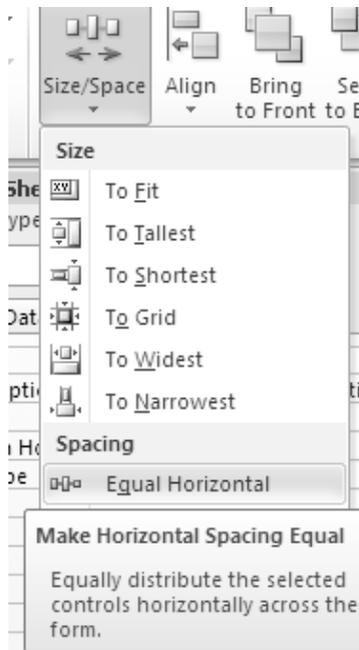
Gambar 49. Button Navigasi

Untuk membuat keempat Button tersebut sejajar, Anda dapat melakukan pemilihan keempat Button tersebut (dengan cara klik-tahan pada mouse atau seperti drag) lalu pilih ribbon *Arrange* dan klik pada opsi *Align* dan pilih sesuai dengan keadaan yang ada.



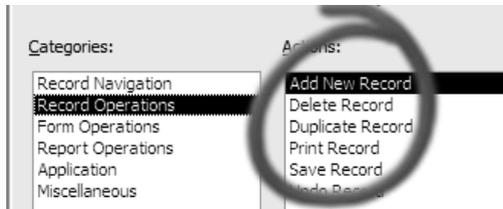
Gambar 50. Opsi Align Button

Sedangkan untuk membuat keempat Button tersebut memiliki jarak yang sama satu sama lain, dapat memilih opsi *Size/Space* di ribbon yang sama (ribbon *Arrange*).



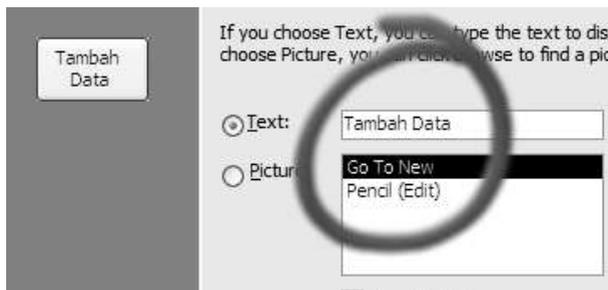
Gambar 51. Opsi Ukuran Button

5. Kini, tambahkan Button berikutnya yang berfungsi untuk menambahkan data pada tabel *Jenis Menu*. Masih sama dengan cara sebelumnya, tambahkan satu Button baru melalui ribbon *Design* dan kemudian pilih *Record Operation* pada kategori yang terdapat pada kotak dialog Button wizard. Kali ini pilihlah opsi *Add New Record* dari tahapan pertama.



Gambar 52. Button Add

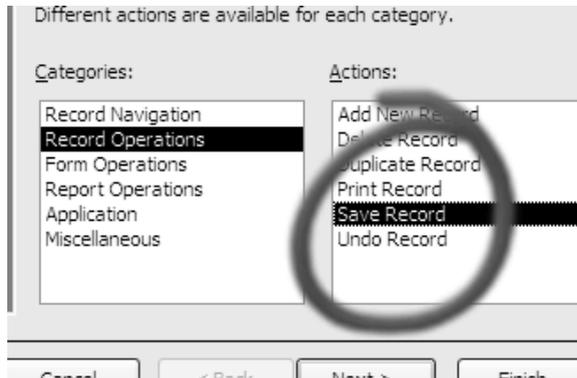
Lalu pada tahapan berikutnya, berikan judul *Tambah Data* di isian *Text* sebagai judul dari Button yang kita buat.



Gambar 53. Judul Button Add

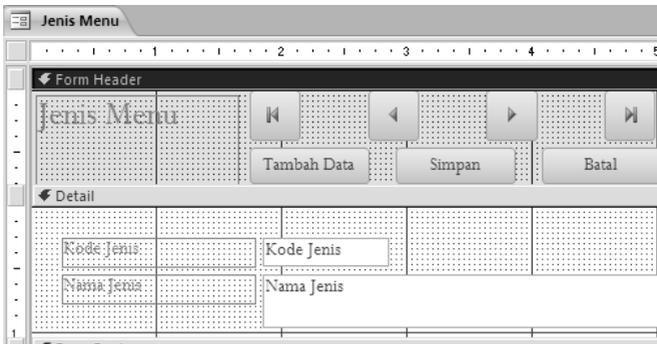
6. Berikutnya, tambahkan Button untuk penyimpanan data, baik pada operasi tambah data maupun edit data. Perlu diingat bahwa secara default, proses simpan dapat dibatalkan oleh Microsoft Access jika pengguna melakukan proses *Undo*. Untuk menambahkan Button tersebut, lakukan langkah yang sama seperti langkah sebelumnya. Tetapi

pada saat pemilihan opsi di tahapan pertama pilihlah *Save Record* untuk kategori *Record Operation*.



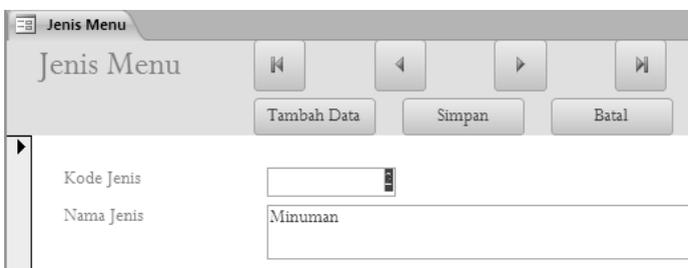
Gambar 54. Opsi Button Save

7. Kemudian tambahkan Button yang terakhir, yakni Button untuk membatalkan operasi atau *Undo*. Proses untuk penambahan Button ini juga sama seperti Button sebelumnya, namun pada opsi *Actions* pilih *Undo Record*.
8. Jika seluruh Button telah tersedia, maka tampilan form di mode *Design* tampak seperti gambar berikut ini:



Gambar 55. Desain Akhir Form Menu

9. Langkah terakhir dari pembuatan form ini adalah melakukan uji coba terhadap manipulasi data dari tabel *Jenis Menu*. Cobalah untuk melakukan penambahan beberapa data sekaligus melakukan proses edit didalamnya. Penambahan data sangat penting dilakukan agar pada form berikutnya juga dapat membantu proses uji coba.



Gambar 56. Hasil Form Menu



Perhatikan pada saat uji coba penambahan data, field *Kode Jenis* secara otomatis terisi dengan nomor urut. Hal ini terjadi karena field tersebut memiliki tipe *Auto Number*.

Setelah menyelesaikan pembuatan form *Jenis Menu*, maka Anda dapat melanjutkan untuk proses pembuatan form *Menu* yang berasal dari tabel *Menu* serta form *Customer* yang berasal dari tabel *Customer*. Kedua form tersebut memiliki proses pembuatan serta layout yang sama dengan form *Jenis Menu*.

Jika kedua form tersebut telah selesai maka tampak seperti pada gambar berikut:

Menu

Tambah Data Simpan Batal

Kode Menu

Jenis Menu Makanan

Nama Menu Nasi Pecel

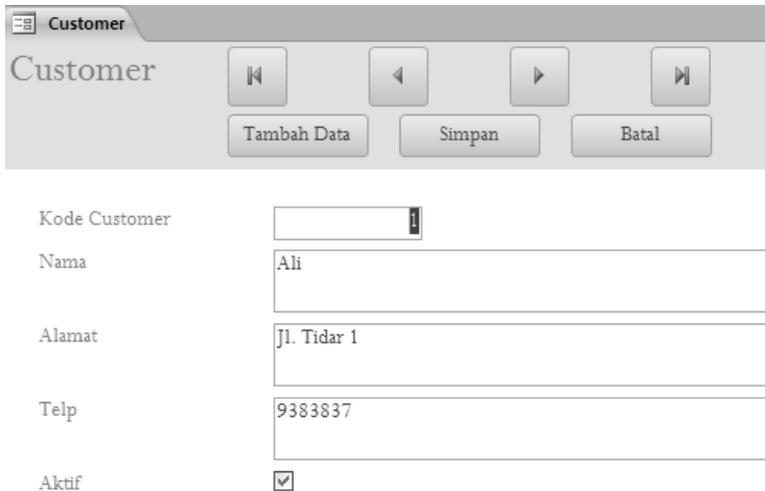
Harga 3000

Aktif

Gambar 57. Uji Coba Form Menu

Pada form *Menu* tampak sebuah field yang secara otomatis berbentuk *Combobox* yakni field *Jenis Menu*. Hal ini diakibatkan tipe dari field tersebut pada saat pembuatan berasal dari *Lookup Wizard* sehingga secara otomatis mengacu ke tabel lain (dalam kasus ini adalah tabel *Jenis Menu* sebagai tabel referensi). Sehingga, jika pada pembuatan form yang Anda lakukan belum berbentuk *Combobox* untuk field tersebut, hal yang perlu dilakukan adalah melakukan revisi terhadap struktur tabel, bukan revisi terhadap form yang bersangkutan.

Begitu pula pada field *Aktif* yang secara otomatis berbentuk *Checkbox* sebagai tanda bahwa field tersebut memiliki tipe *Yes/No* atau sering juga disebut sebagai tipe *Boolean*. Sehingga dengan format tersebut sangat memudahkan pengguna dalam melakukan manipulasi data.



Customer	
Kode Customer	<input type="text"/>
Nama	Ali
Alamat	Jl. Tidar 1
Telp	9383837
Aktif	<input checked="" type="checkbox"/>

Gambar 58. Uji Coba Checkbox

Untuk form *Customer* secara sekilas tidak terlihat sesuatu yang khusus, karena struktur tabel yang ada di dalam tabel tersebut tergolong sederhana. Dan kini, setelah ketiga form yang memiliki kategori sederhana tersebut selesai dilakukan, maka saatnya berlanjut ke form yang lebih membutuhkan perhatian ekstra.

Form Paket Menu

Sebelum memulai proses pembuatan form untuk *Paket Menu*, terlebih dulu pastikan bahwa Anda telah melakukan pengisian data di tabel *Jenis Menu* maupun pada tabel *Menu*. Hal ini dimaksudkan agar pada saat form *Paket Menu* selesai dibuat dapat dilakukan uji coba yang mendekati kenyataan.

Form *Paket Menu* nantinya melibatkan empat tabel sekaligus yakni tabel *Jenis Menu* dan tabel *Menu* sebagai tabel referensi. Serta tabel *Paket* dan tabel *Detail Paket* yang keduanya dimanipulasi secara bersamaan dalam satu form yang sama.



Di dalam sistem informasi katering yang sedang dibuat, pemesanan dari tiap pelanggan tidak berdasarkan pada satu menu atau satu jenis menu. Melainkan berdasarkan satu paket menu yang sebelumnya telah didefinisikan oleh pihak katering. Sehingga yang dimaksud dengan paket menu adalah satu kesatuan menu yang ditawarkan oleh pihak katering ke pelanggan, atau dengan kata lain pelanggan tidak bisa menentukan sendiri paket menu yang diinginkan di dalam sistem informasi ini.

Manipulasi tabel majemuk dalam sebuah form mutlak memerlukan teknik penempatan *sub form* atau form yang terletak di dalam form lain. Penempatan sub form dapat dilakukan dengan dua macam cara yaitu: pembuatan sub form secara terpisah, dan penempatan sub form secara langsung kepada form induk.

Di dalam pembuatan form kali ini dilakukan penempatan sub form secara langsung kepada form induk. Hal ini dilakukan agar pembuatan form lebih

cepat serta mempermudah pemahaman terhadap langkah demi langkah yang dijelaskan di tahapan berikutnya. Dan berikut adalah langkah pembuatan form untuk keperluan pengisian data *Paket Menu*.

1. Buat sebuah form dengan menggunakan fasilitas *form wizard* dengan menggunakan tabel *Paket* sebagai dasar pembuatan form.



Gunakan langkah yang sama seperti pada pembuatan form sebelumnya, yakni dengan memilih ribbon *Create* dan klik pada ikon *Form wizard*. Lalu pada kotak dialog yang tersedia, pilih tabel *Paket* dan arahkan semua field untuk masuk ke dalam form. Jika proses pembuatan form berjalan lancar, maka pada mode *Design* tampak seperti pada gambar berikut.

Form Header				
Paket Menu				

Detail				
Kode Paket:		Kode Paket		
Nama Paket:		Nama Paket		

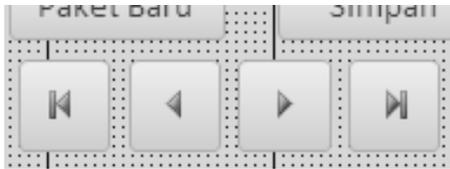
Gambar 59. Desain Form Paket Menu

2. Dengan menggunakan langkah yang sama seperti pada pembuatan form sebelumnya, lakukan penambahan tiga button baru untuk penambahan data paket baru, penyimpanan data serta pembatalan aktifitas.

nu	Paket Baru	Simpan	Batal
----	------------	--------	-------

Gambar 60. Tombol Form Paket

3. Kemudian tambahkan empat button untuk navigasi (First, Previous, Next dan Last Record). Gunakan cara yang sama seperti pada pembuatan form sebelumnya.



Gambar 61. Navigasi Button

4. Salah satu perbedaan dari form ini dibanding form sebelumnya adalah ditambahkan sebuah komponen baru yakni Combobox. Fungsi dari Combobox tersebut nantinya adalah untuk melakukan navigasi secara cepat dengan menyediakan fasilitas pemilihan paket menu yang diedit oleh pengguna. Untuk menambahkan Combobox tersebut, klik pada ribbon *Design* lalu pilih komponen Combobox.



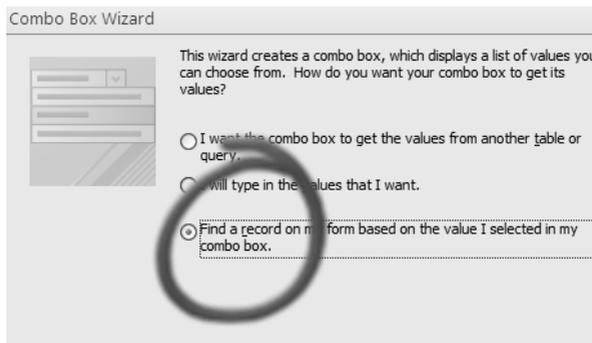
Gambar 62. Icon Combobox

Berikutnya, tempatkan di sebelah kanan dari rangkaian button navigasi, sehingga tampak seperti pada layout berikut:



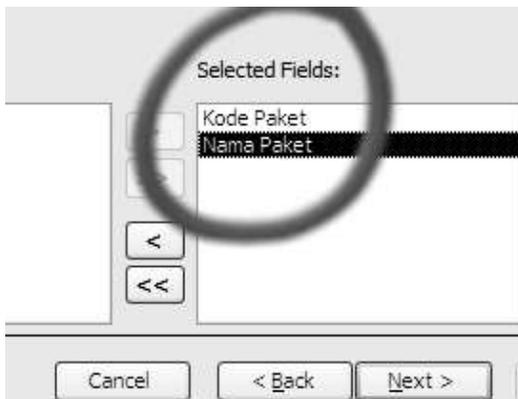
Gambar 63. Layout Combobox

5. Pada saat penempatan Combobox, muncul kotak dialog wizard untuk melakukan pemilihan data yang nantinya ditempatkan dalam komponen tersebut. Pada kotak dialog tahap pertama, pilih opsi ketiga yakni untuk pencarian data dalam sebuah form.



Gambar 64. Opsi Combobox

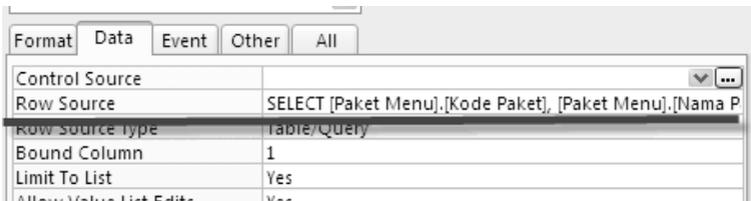
6. Kemudian, pada tahap kedua pilih semua field dari tabel *Paket* sebagai field yang masuk ke dalam Combobox tersebut. Dari dua field yang dipilih hanya field *Nama Paket* yang menjadi tampilan dalam Combobox, sebab field *Kode Paket* sebagai field *Primary Key* hanya menjadi penanda navigasi pada saat filter data dieksekusi.



Gambar 65. Pilihan Field Combobox

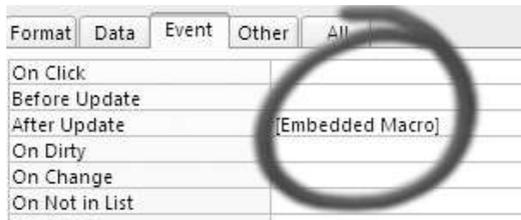
7. Lalu selesaikan kotak dialog wizard tersebut, sehingga penempatan Combobox selesai dilakukan. Dan jika ditelaah, sebenarnya terdapat dua proses yang telah dilakukan di dalam proses wizard sebelumnya. Proses yang pertama adalah menempatkan query untuk pemilihan data di dalam Combobox. Query tersebut dapat dilihat di jendela

Property tepat di area tab *Data*. Kemudian perhatikan pada property *Row Source* yang didalamnya telah terisi query untuk menampilkan data dari tabel *Paket*.



Gambar 66. Opsi Row Source

Sedangkan proses yang kedua adalah penempatan listing *Macro* pada event *After Update* yang merupakan pemicu dari proses filter data di dalam form. Untuk melihat listing macro hasil dari kotak dialog wizard tersebut, maka dapat dilihat pada jendela Property di tab *Event* serta perhatikan pada event *After Update*.



Gambar 67. Embeded Macro

Jika Anda ingin mengetahui listing yang terdapat didalam macro tersebut, bisa diklik pada button yang tersedia. Maka tampak listing macro sebagai berikut :

```
SearchForRecord
  Object Type
  Object Name
  Record First
  Where Condition = "[Kode Paket] = " & Str(Nz([Screen].[ActiveControl],0))
```

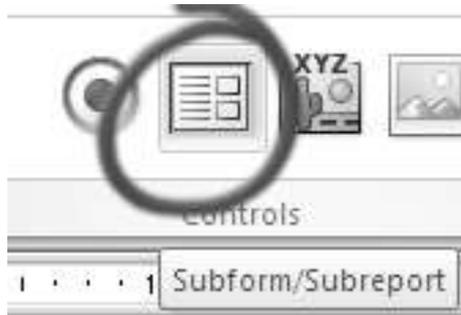


Pada penempatan macro dengan menggunakan wizard, pengembang sistem informasi memang merasakan kemudahan. Tetapi di dalam beberapa kasus tertentu, pengembang sistem informasi tetap harus melakukan pembuatan macro sendiri. Namun demikian, di dalam Microsoft Access 2010 pembuatan macro menjadi sangat mudah dan nyaman dilakukan bahkan oleh para pemula sekalipun.

Logika sederhana dari listing macro tersebut adalah menempatkan terlebih dulu navigasi record ke record yang pertama (*Record First*). Kemudian

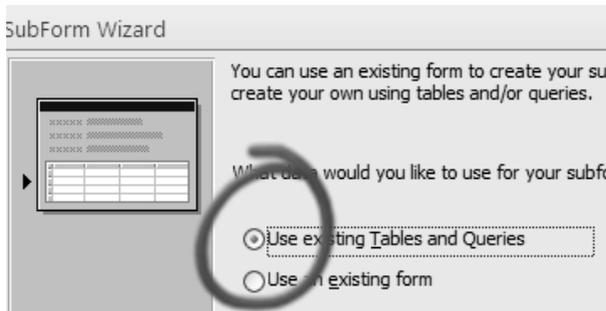
dilakukan filter dengan menggunakan field *Kode Paket* sebagai kunci pencarian. Field *Kode Paket* yang dicari didapat dari Combobox pada form.

- Langkah selanjutnya adalah menempatkan sub form dari tabel *Detail Paket* sebagai bagian terintegrasi dalam manipulasi data tabel *Paket*. Untuk menempatkan sub form, tetap pilih ribbon *Design* dan klik pada komponen *Sub Form/Sub Report*.



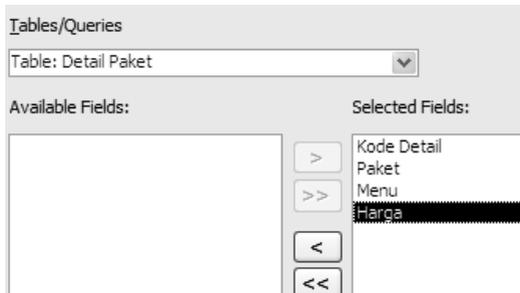
Gambar 68. Icon Subform

Di dalam kotak dialog wizard yang muncul setelah penempatan komponen tersebut, pilih opsi untuk menggunakan tabel yang sudah ada. Pemilihan ini dilakukan karena belum pernah dilakukan pembuatan form untuk tabel *Detail Paket*.



Gambar 69. Opsi Subform

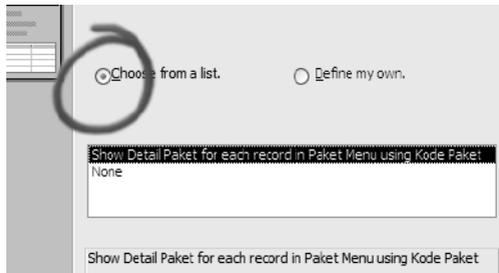
Pada tahap berikutnya di kotak dialog yang sama, pilih tabel *Detail Paket* dan arahkan semua field agar tampil di dalam sub form yang ditempatkan.



Gambar 70. Field Subform

Di tahapan yang ketiga pada kotak dialog wizard tersebut, pilih opsi untuk merelasikan antara tabel *Detail Paket* dengan tabel *Paket*. Relasi ini terbentuk secara otomatis jika pada saat

pembuatan tabel *Detail Paket*, field *Paket* dibuat dengan memilih tipe *Lookup Wizard* (lihat lagi pada sub bab mengenai pembuatan tabel *Detail Paket*).



Gambar 71. List Subform



Kunci utama dari sebuah sistem informasi yang baik adalah analisa dan perancangan yang akurat. Sehingga pada tahapan ini bisa dilihat, andai relasi yang dibuat di awal mengalami kesalahan, maka pembuatan form mengalami kesulitan bahkan kemungkinan gagal. Sehingga proses analisis harus dilaksanakan lebih cermat dan karena pada saat pembuatan sistem informasi telah dibantu dengan adanya perangkat lunak yang sangat memudahkan pekerjaan pengembang seperti halnya Microsoft Access.

9. Langkah terakhir dalam pembuatan form ini adalah menambahkan sebuah rangkaian komponen yang nantinya menampilkan total harga dari sebuah paket menu. Untuk menempatkan total harga, tempatkan sebuah Textbox tepat di bawah sub form *Detail Paket*.



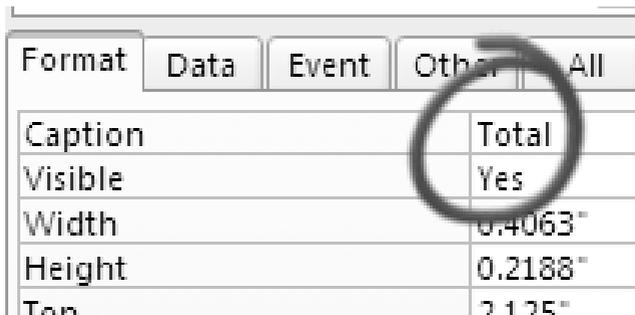
Penempatan Textbox di dalam form Microsoft Access secara otomatis menempatkan sebuah komponen Label di sebelah kirinya. Sehingga pada saat penempatan Textbox menjadi satu paket dengan penempatan Label.

Penempatan Textbox memiliki proses yang sama dengan penempatan komponen lainnya. Pilih terlebih dulu ribbon *Design* dan klik pada komponen Textbox.



Gambar 72. Icon Textbox

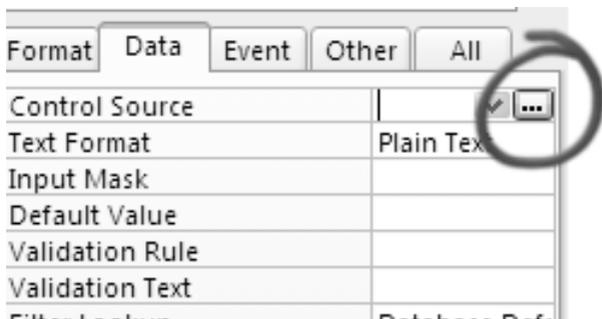
Kemudian set property *Caption* pada Label menjadi *Total* sebagai keterangan pada rangkaian komponen tersebut.



Format	Data	Event	Other	All
Caption				Total
Visible				Yes
Width				0.4063"
Height				0.2188"
Top				0.175"

Gambar 73. Opsi Caption

Selanjutnya pada Textbox yang tersedia set property *Control Source* dengan melakukan klik pada button yang ada.

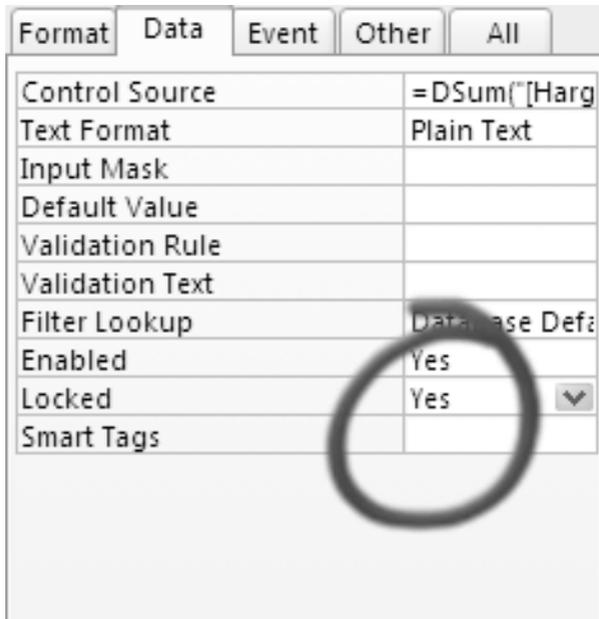


Gambar 74. Opsi Control Source

Di dalam kotak dialog *Expression Builder* yang muncul setelah menekan button tersebut, ketikkan listing berikut :

```
=DSum("[Harga]", "[Detail  
Paket]", "[Paket]=[Kode Paket] ")
```

Selanjutnya set property *Locked* pada Textbox tersebut menjadi Yes. Hal ini dimaksudkan agar Textbox tidak bisa diedit oleh pengguna, sebab isi dari Textbox tersebut hanya untuk keterangan.



Gambar 75. Opsi Locked

10. Kini saatnya melakukan testing terhadap form yang sudah dibuat. Klik pada ikon *View* untuk menuju mode eksekusi form. Berikut adalah contoh hasil dari form untuk *Paket*.

Paket Menu

Kode Paket

Nama Paket

Detail Paket subform

Menu	Harga
Nasi Pecel	3000
Es Jeruk	1500
Buah kecil	1500
*	

Total

Gambar 76. Hasil Form Paket Menu

Form Transaksi Pemesanan

Form berikutnya merupakan “nyawa” dari sistem informasi yang dibuat. Form tentang transaksi pemesanan dari pelanggan ini secara proses sesungguhnya jauh lebih sederhana dibandingkan form-form sebelumnya. Hal ini disebabkan dalam form transaksi pemesanan hanya terdapat satu proses yakni penambahan data, tanpa ada proses lainnya seperti edit data maupun navigasi data.

Di dalam pembuatan form transaksi pemesanan melibatkan dua buah tabel yakni tabel *Transaksi* dan tabel *Detail Transaksi*. Namun demikian cara penempatan tabel *Detail Transaksi* tidak sama dengan form sebelumnya yakni form *Paket*.

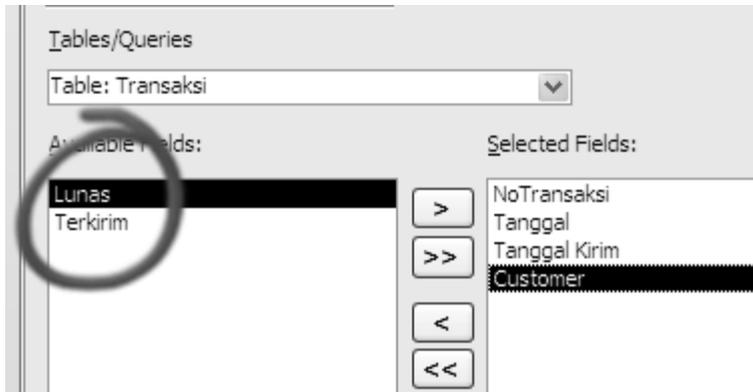
Jika di dalam form *Paket* menggunakan teknik penempatan komponen sub form untuk tabel kedua, maka di dalam form transaksi pemesanan menggunakan listing VBA (Visual Basic for Application). Hal ini sengaja dilakukan agar Anda dapat pula mengenal sekaligus memahami berbagai jenis teknik pembuatan dan modifikasi form di dalam Microsoft Access.



Dengan memahami berbagai jenis teknik pembuatan dan modifikasi form membuat seorang pengembang sistem informasi menjadi lebih “kuat”. Sehingga diharapkan tidak mudah menyerah jika terjadi kondisi yang didalamnya tidak lagi dapat dipecahkan menggunakan macro biasa.

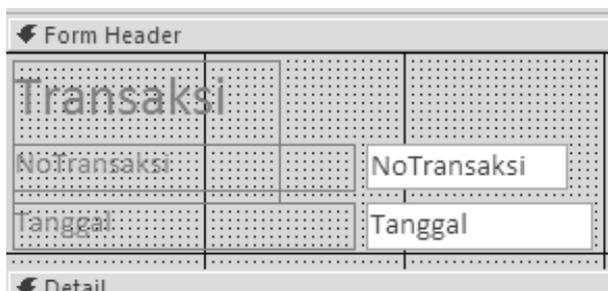
Berikut adalah langkah-langkah pembuatan form untuk transaksi pemesanan:

1. Seperti pada pembuatan form sebelumnya, buat sebuah form melalui ribbon *Create* dan memilih ikon *Form Wizard*. Tetapi kali ini dengan menggunakan tabel *Transaksi* sebagai tabel utama dalam form tersebut. Tetapi pada saat kotak dialog wizard sampai di tahapan untuk pemilihan field, tinggalkan field *Lunas* dan field *Terkirim* untuk tidak ditampilkan di dalam form.



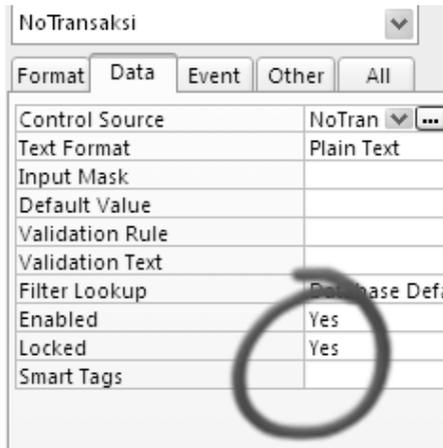
Gambar 77. Pemilihan Field Form Transaksi

2. Kemudian pindahkan rangkaian komponen untuk field *NoTransaksi* dan field *Tanggal* ke bagian *header* dari form. Pemindahan ini dimaksudkan agar pengguna merasa bahwa form transaksi untuk pemesanan mirip dengan pengisian sebuah nota pemesanan biasa.



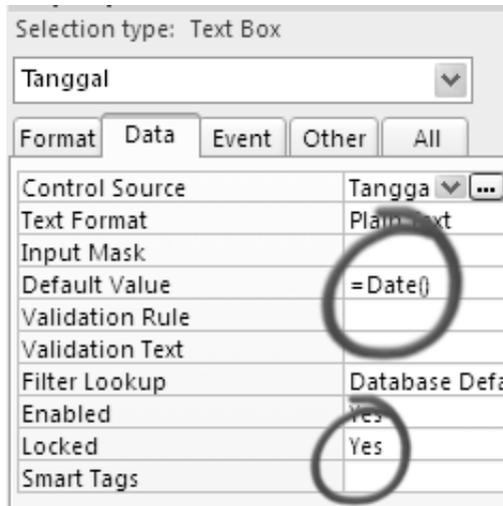
Gambar 78. Desain Header Form Transaksi

3. Selanjutnya, set property *Locked* untuk kedua Textbox yang baru saja dipindahkan menjadi *Yes*. Setting property ini dilakukan karena pada saat pengisian transaksi, kedua field tersebut diisi secara otomatis.



Gambar 79. Opsi Enabled dan Locked

4. Khusus untuk isian field *Tanggal*, set property *Default Value* menjadi `=Date()`. Setting ini nantinya berfungsi agar pada saat pengisian pemesanan, tanggal transaksi secara otomatis terisi dengan tanggal sistem (atau tanggal dari komputer pengguna).



Gambar 80. Default Value dengan tipe Date

5. Kini tambahkan dua buah Button yang masing-masing berfungsi untuk menambahkan data baru (*Record Operation* → *Add New Record*) dan melakukan penyimpanan data (*Record Operation* → *Save Record*). Untuk melakukan penambahan button, lihat kembali ke pembahasan form sebelumnya.



Gambar 81. Tombol Atas Form Transaksi

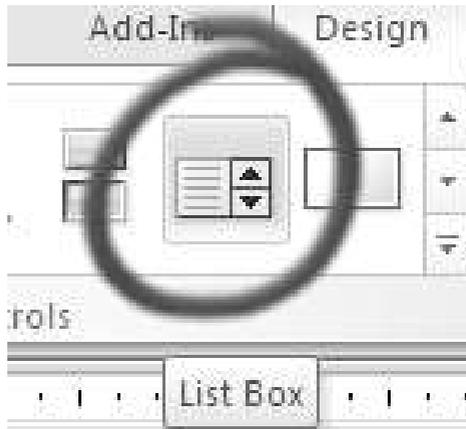
- Langkah berikutnya adalah menambahkan rangkaian komponen untuk membuat transaksi pemesanan yang dilakukan oleh seorang pelanggan dapat memilih beberapa paket sekaligus, serta secara otomatis menghitung total pesanan yang telah dilakukan. Dengan menggunakan teknik yang sedikit berbeda, rangkaian komponen untuk kepentingan ini melibatkan dua buah Listbox, dua buah Button serta sebuah rangkaian Textbox dan Label.

Listbox yang pertama nantinya digunakan sebagai tempat penampung data paket menu yang tersedia, sedangkan Listbox yang kedua berfungsi sebagai penampung data paket menu yang dipesan oleh pelanggan.

Sedangkan untuk Button yang ada, satu Button nantinya berfungsi untuk menambahkan paket

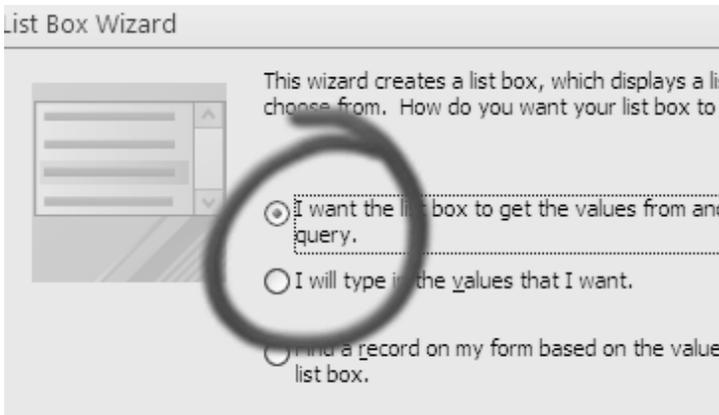
menu dari Listbox pertama ke Listbox kedua. Dan satu Button lainnya memiliki fungsi yang merupakan kebalikan dari fungsi Button pertama. Untuk rangkaian Textbox dan Label digunakan sebagai penampung total harga pemesanan yang dilakukan oleh pelanggan. Dan dari seluruh komponen tersebut, digunakan

7. Untuk penambahan Listbox yang pertama, di dalam ribbon *Design* klik ikon Listbox.



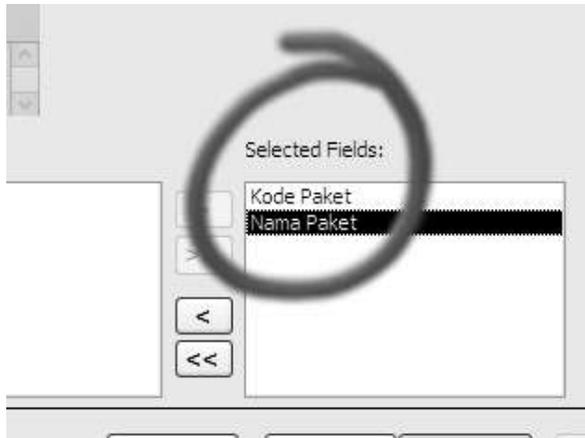
Gambar 82. Icon Listbox

Di dalam kotak dialog wizard pada tahapan yang pertama, pilih opsi untuk memilih data dari tabel yang tersedia.



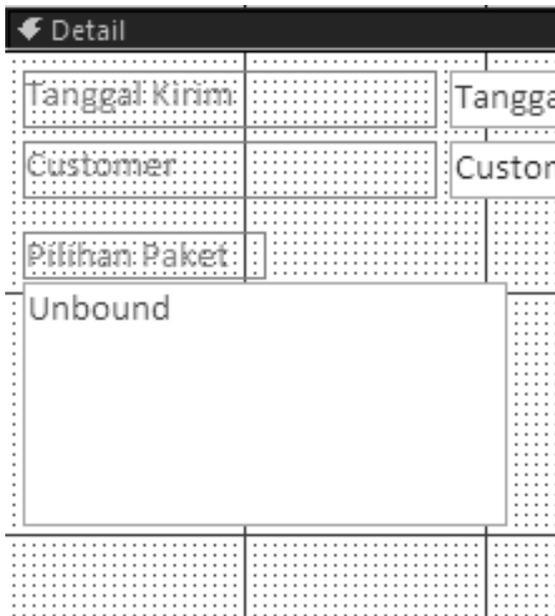
Gambar 83. Opsi Listbox

Berikutnya pilih tabel *Paket* sebagai tabel yang dipilih untuk mengisi Listbox. Dari field yang tersedia, pilih semua field yang tersedia, sehingga nantinya Listbox hanya terisi field *Nama Paket* dan field *Kode Paket* sebagai primary key secara otomatis memiliki status *hidden*.



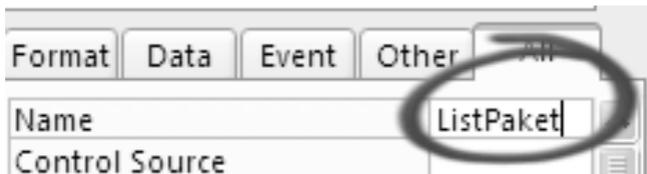
Gambar 84. Field Listbox

Kemudian atur posisi dari Listbox sedemikian rupa sehingga menjadi seperti layout berikut:



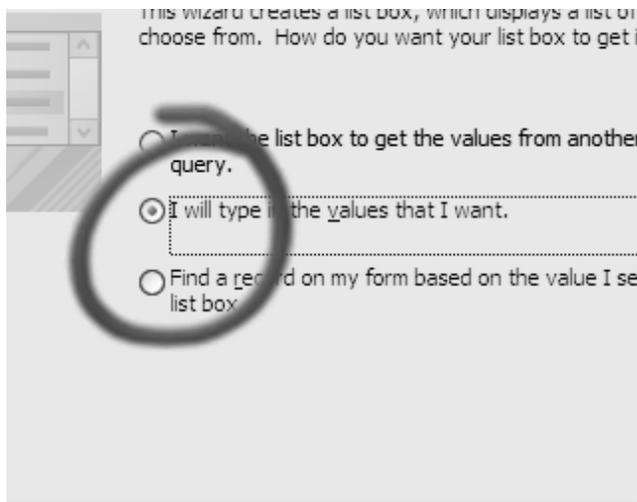
Gambar 85. Tampilan Listbox

Beri nama Listbox yang pertama dengan nama *ListPaket*. Cara pemberian nama dilakukan melalui property *Name* yang terdapat di dalam jendela property.



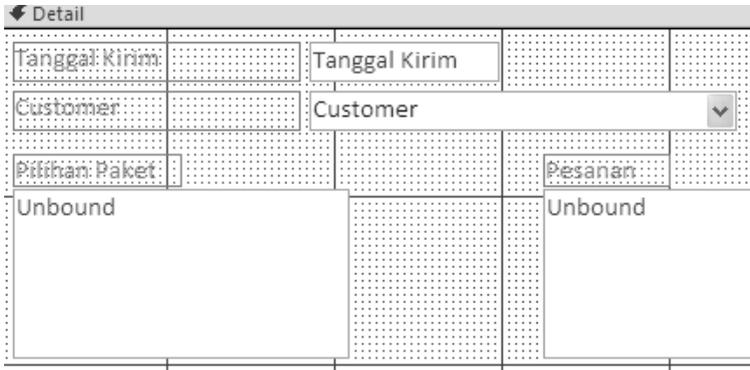
Gambar 86. Opsi Nama Listbox

8. Selanjutnya tambahkan Listbox yang kedua tepat di sebelah kanan Listbox yang pertama. Pada kotak dialog wizard untuk Listbox yang kedua, pilih opsi untuk tidak melibatkan data apapun di dalam Listbox tersebut. Hal ini dilakukan karena nantinya pada Listbox kedua hanya terisi jika pengguna melakukan klik pada Button yang dibuat di langkah berikutnya.



Gambar 87. Opsi Isian Listbox Manual

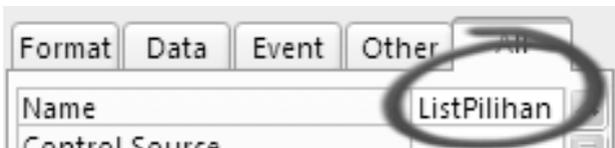
Kemudian atur layout dari Listbox tersebut sehingga tampak seperti pada gambar berikut:



Detail			
Tanggal Kirim:		Tanggal Kirim	
Customer:		Customer	▼
Pilihan Paket:			Pesanan
Unbound			Unbound

Gambar 88. Hasil Listbox Kedua

Seperti halnya Listbox yang pertama, beri nama Listbox yang kedua dengan nama *ListPilihan* di dalam jendela property, tepatnya di property *Name*.

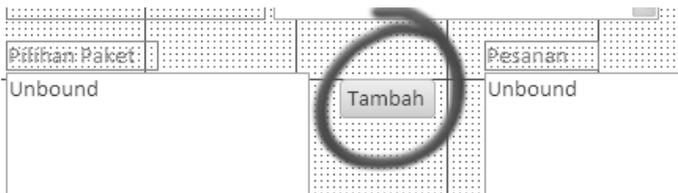


Gambar 89. Opsi Nama Listbox Kedua



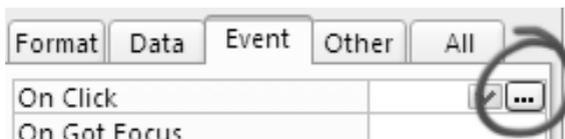
Pada proyek sistem informasi yang didalamnya memiliki kompleksitas tinggi, disarankan melakukan pemberian nama tiap komponen. Supaya memudahkan jika terjadi revisi di dalam sistem itu.

9. Kemudian tempatkan button yang digunakan untuk menambahkan data dari Listbox pertama ke Listbox kedua. Pada kotak dialog wizard yang tersedia, klik tombol *Cancel* untuk membatalkan wizard karena nantinya button ini diisi dengan listing VBA.



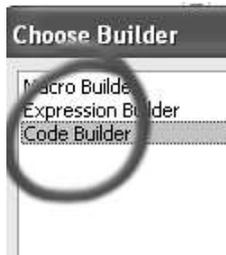
Gambar 90. Button Tambah

Kini klik pada button yang terdapat pada event *On Click* tepat di dalam jendela property dari button tersebut untuk mengetikkan listing VBA.



Gambar 91. Opsi On Click

Pada kotak dialog yang tersedia, pilih opsi *Code builder* sebagai opsi untuk VBA. Kedua opsi lain yang tersedia merupakan opsi untuk mengisi *event* dari komponen dengan menggunakan macro.



Gambar 92. Opsi Code Builder

Setelah menuju ke *code editor* atau tempat pengetikan listing VBA, ketikkan listing berikut :

```
Dim strItems As String
Dim intCurrentRow As Integer
For intCurrentRow = _
    0 To ListPaket.ListCount - 1
    If
ListPaket.Selected(intCurrentRow) Then
        strItems = _
ListPaket.Column(0, _
                intCurrentRow) & ";" & _
                ListPaket.Column(1, _
                intCurrentRow) & ";" & _
ListPaket.Column(2, intCurrentRow) _
        ListPilihan.AddItem _
        Item:=strItems
```

```
End If
Next intCurrentRow
hitungTotal
```

Pada listing tersebut, dua baris pertama merupakan penanda untuk deklarasi variabel yang digunakan. Kemudian pada looping yang tersedia (*For....Next*), merupakan iterasi yang digunakan untuk melakukan pemindahan item dari Listbox pertama ke Listbox kedua.

Namun sebelum pemindahan dilakukan, terlebih dulu dilakukan pengecekan apakah item tersebut sudah ada di Listbox yang kedua. Jika ternyata item belum ada, maka item baru boleh ditambahkan.

Pemindahan item dilakukan dengan menggunakan tanda pemisah *semicolon* (titik koma → ;). Hal ini dilakukan karena pada Listbox pertama yang berisi data dari tabel *Paket* sebenarnya terdiri dari dua field yaitu field *Kode* dan *Nama*.

Sedangkan untuk baris terakhir yang merupakan pemanggilan nama fungsi untuk melakukan perhitungan total yang dibuat pada tahapan akhir.

10. Setelah selesai untuk pembuatan button yang pertama, kini kembali ke mode desain form dan

menambahkan button yang kedua. Proses penambahan button yang kedua sama dengan saat penambahan button yang pertama (sama-sama menggunakan opsi *Code builder*). Tempatkan button yang kedua tepat di bawah button yang pertama.



Gambar 93. Button Hapus

Setelah sampai di tempat pengetikan listing VBA, maka ketikkan listing berikut:

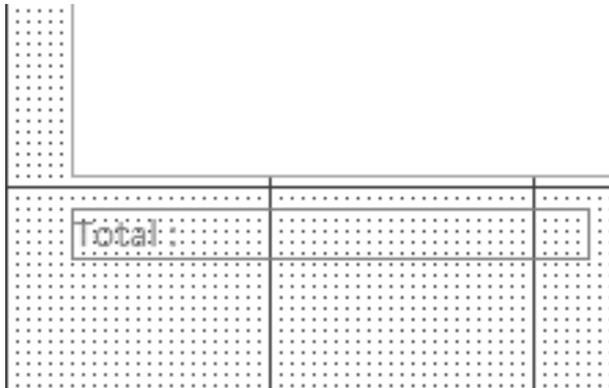
```
Dim intCurrentRow As Integer
For intCurrentRow = 0 To
ListPilihan.ListCount - 1
    If
ListPilihan.Selected(intCurrentRow)
Then
ListPilihan.RemoveItem(intCurrentRow)
    End If
Next intCurrentRow
hitungTotal
```

Pada listing tersebut, hanya dilakukan sebuah iterasi untuk menghapus data dari Listbox kedua. Penghapusan hanya dilakukan jika pengguna melakukan pemilihan item tertentu dari Listbox yang kedua (ditandai dengan adanya pengecekan melalui percabangan → perintah *If*). Setelah item yang dimaksud terhapus, maka dilakukan perhitungan ulang untuk total pesanan yang ada.



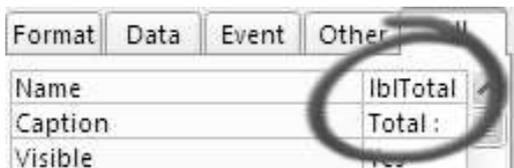
Sangat perlu diperhatikan bahwa VBA atau Visual Basic for Application sama sekali berbeda dengan Visual Basic .NET. VBA hanya berlaku di lingkup Microsoft Office, baik Word, Excel, Power Point maupun Access. Meski terdapat banyak kesamaan dalam hal sintaks bahasa pemrograman, tetapi VBA memiliki alur yang berbeda untuk tiap penggunaan dalam Microsoft Office.

11. Berikutnya, letakkan sebuah Label tepat di bawah Listbox yang kedua. Kemudian ganti property *Caption* pada Label tersebut dengan teks *Total*.



Gambar 94. Label Total

Selain penggantian property *Caption*, lakukan pula penggantian pada property *Name*. Hal tersebut dilakukan agar pada saat terjadi perhitungan ulang untuk total pesanan yang dilakukan di dalam listing VBA menjadi lebih mudah.



Gambar 95. Opsi Nama Label

Jika seluruh komponen telah ditempatkan ke dalam form Transaksi, maka terlihat seperti pada gambar berikut :

Transaksi			
NoTransaksi		NoTransaksi	
Tanggal		Tanggal	
Detail			
Tanggal Kirim		Tanggal Kirim	
Customer		Customer	▼
Pilihan Paket		Pesanan	
Unbound		Tambah	Unbound
		Hapus	
		Total	

Gambar 96. Desain Form Transaksi

12. Dengan posisi masih tetap di form yang sama, ketik kembali prosedur yang digunakan untuk menghitung sub total dari transaksi yang dilakukan. Jika kesulitan untuk mengakses kembali listing VBA, klik pada toolbar untuk icon *view code*:



Gambar 97. Icon View Code

```
Sub hitungTotal()  
Dim xTotal As Integer  
For intCurrentRow = 0 To _  
    ListPilihan.ListCount - 1  
    xTotal = xTotal + _  
    CInt(ListPilihan.Column _  
        (2, intCurrentRow))  
Next intCurrentRow  
lblTotal.Caption = _  
    "Total : Rp. " & xTotal  
End Sub
```

Prosedur untuk menghitung sub total tersebut, pada dasarnya melakukan perulangan seluruh data yang ada di dalam listbox dari pemesanan yang telah dilakukan. Prosedur tersebut dipanggil setiap kali terjadi penambahan pesanan ataupun penghapusan pesanan, sehingga setiap pesanan berubah di dalam form, maka sub total selalu dimodifikasi ke nilai yang terbaru.

13. Berikutnya tambahkan sebuah button ke dalam form tersebut, tepatnya di bagian header dari form.



Gambar 98. Icon Button



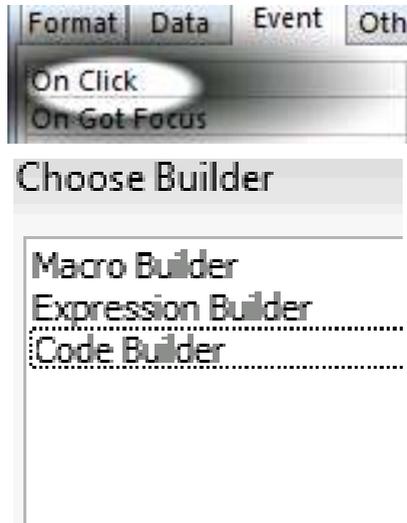
Gambar 99. Opsi Nama Button

Berikan teks *Simpan Pesanan* untuk property *Caption* di button tersebut. Perlu diperhatikan bahwa untuk button ini tidak menggunakan fasilitas wizard yang telah disediakan, sehingga jika muncul kotak dialog untuk pendefinisian fungsi dari button tersebut bisa diabaikan dengan menekan tombol cancel.



Gambar 100. Penempatan Button Simpan

Selanjutnya di jendela property, klik pada *event on click* dan pada kotak dialog yang tersedia, pilih opsi untuk *code builder* agar masuk ke jendela VBA.



Gambar 101. Opsi Code Builder

Selanjutnya ketikkan listing berikut ke dalam prosedur *on click* untuk button tersebut.

```
DoCmd.Save  
DoCmd.RefreshRecord  
Dim xRs As DAO.Recordset  
Set xRs = CurrentDb.OpenRecordset _  
("Detail Transaksi", dbOpenDynaset)  
Dim intCurrentRow As Integer  
For intCurrentRow = 0 To _  
ListPilihan.ListCount - 1  
With xRs
```

```
.AddNew
!NoTransaksi = Me.NoTransaksi
!Paket = ListPilihan.Column _
    (0, intCurrentRow)
.Update
End With
Next intCurrentRow
For intCurrentRow = 0 To _
    ListPilihan.ListCount - 1
    ListPilihan.RemoveItem (0)
Next intCurrentRow
lblTotal.Caption = "Total"
DoCmd.Requery
```

Pada saat proses penyimpanan di form transaksi ini sedikit berbeda dengan proses penyimpanan di form-form sebelumnya. Sebab dalam form ini dilakukan dua kali proses penyimpanan yaitu di dalam tabel *Transaksi* dan tabel *Detail Transaksi*.

Untuk tabel *Transaksi*, hanya dilakukan satu kali penyimpanan record didalamnya dengan menggunakan perintah *DoCmd.Save*. Sehingga proses penyimpanan ini masih mirip dengan proses

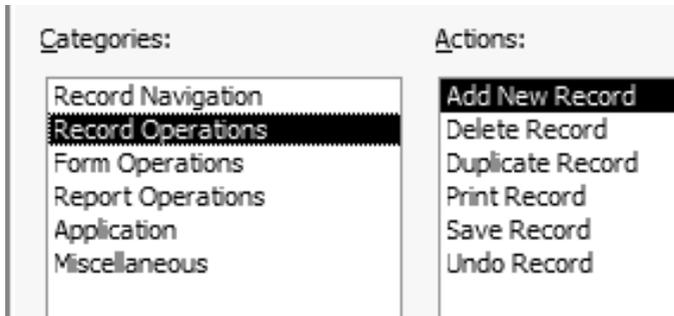
penyimpanan di form yang sebelumnya telah dikerjakan.

Sedangkan untuk tabel *Detail Transaksi*, dilakukan penyimpanan dengan jumlah yang sesuai dengan jumlah detail pemesanan yang ada di dalam listbox. Satu hal yang penting diperhatikan di dalam penyimpanan data pada tabel ini adalah referensi nomor transaksi yang harus sama dengan yang ada pada tabel *Transaksi*. Jika tahapan tersebut sudah selesai dilakukan, maka dilakukan perulangan untuk menyimpan tiap item yang tersedia di dalam listbox dengan melakukan koneksi database yang terpisah dan hanya tersedia di dalam listing VBA. Langkah terakhir adalah melakukan "pembersihan" dari listbox yang isinya telah disimpan ke dalam tabel *Detail Transaksi*.



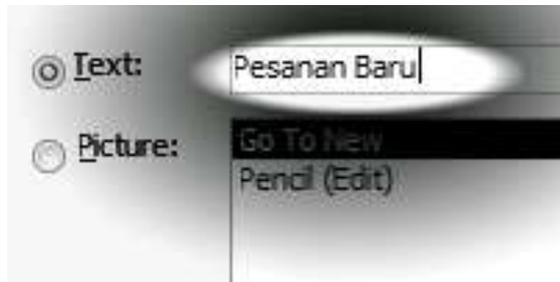
Perlu diperhatikan bahwa terdapat sangat banyak variasi teknik pemrograman di dalam melakukan implementasi proses transaksi di dalam suatu proses bisnis. Anda dapat menciptakan sendiri teknik pemrograman sesuai dengan kebutuhan dan kasus yang dihadapi. Namun demikian, diharapkan dengan contoh yang tersedia di dalam buku ini dapat menginspirasi untuk menciptakan solusi yang sesuai dengan kebutuhan Anda sendiri.

14. Langkah terakhir adalah menambahkan satu button lagi untuk penambahan pesanan. Kali ini untuk penambahan button, dapat menggunakan kotak dialog wizard yang telah disediakan oleh Microsoft Access 2010. Pada langkah pertama dari kotak dialog tersebut, pilih kategori *Record Operation* dan opsi *action* untuk penambahan *Add New Record*.



Gambar 102. Opsi Add New

Untuk property teks di button tersebut, berikan teks *Pesanan Baru*.



Gambar 103. Opsi Teks New

Hasil dari desain form yang telah lengkap terlihat seperti pada gambar berikut:

The image shows a wireframe of a transaction form. It is divided into two main sections: 'Form Header' and 'Detail'.
Form Header: Contains a title 'Transaksi' and several input fields: 'NoTransaksi', 'Tanggal', 'Pesanan Baru', and 'Simpan Pesanan'.
Detail: Contains a table with columns for 'Tanggal Kirim', 'Customer', and 'Pilihan Paket'. The table has two rows. The first row has 'Unbound' in the first column, 'Tambah' in the second, and 'Unbound' in the third. The second row has 'Unbound' in the first column, 'Hapus' in the second, and 'Unbound' in the third. Below the table is a 'Total' row with a 'Total' label and a numerical input field.

Gambar 104. Desain Akhir Form Transaksi

15. Kini lakukan uji coba untuk form yang telah dibuat, dengan melakukan entri data transaksi dan cek data yang telah dimasukkan di tabel *Transaksi* dan tabel *Detail Transaksi*.

Transaksi

NoTransaksi

Tanggal

Tanggal Kirim

Customer

Pilihan Paket

Paket 1	6000
Paket 2	5500
Paket 3	6000

Pesanan

Paket 1	6000
Paket 2	5500

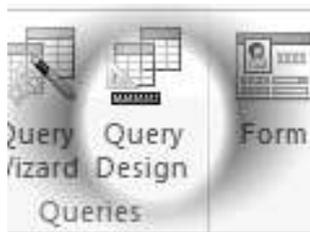
Total : Rp. 11500

Gambar 105. Uji Coba Form Transaksi

Form Pelunasan

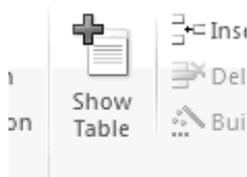
Form terakhir yang ada di dalam ruang lingkup transaksi, yakni form untuk pelunasan transaksi. Konsep awal dari form ini adalah dengan melakukan pencarian terlebih dulu terhadap transaksi yang memiliki status belum lunas, kemudian pihak catering diasumsikan telah menerima pembayaran dari pelanggan dan mengubah status tersebut menjadi lunas.

Sebelum melakukan pembuatan form, terlebih dulu dilakukan pembuatan query yang nanti digunakan di dalam desain form. Query tersebut dibuat dengan melakukan klik ikon pada toolbar untuk *Query Design* (tidak dengan menggunakan wizard).



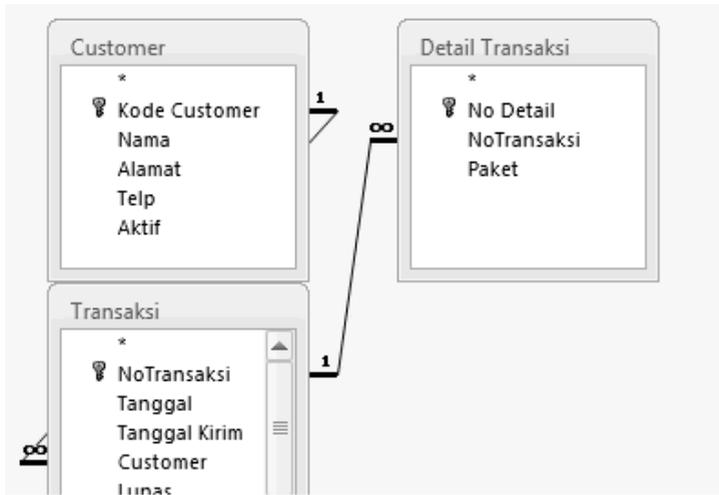
Gambar 106. Icon Query Design

Selanjutnya, tambahkan tiga tabel dengan terlebih dulu dengan menjalankan klik pada ikon *show table*. Pada mode pembuatan query ini, tabel yang disertakan di dalam query memang ditambahkan secara manual, tidak secara otomatis.



Gambar 107. Icon Show Table

Berikutnya tambahkan tiga tabel yang saling berelasi satu sama lain (dan terbentuk relasinya secara otomatis), yaitu tabel *customer*, tabel *transaksi* dan tabel *detail transaksi*.



Gambar 108. Relasi Transaksi

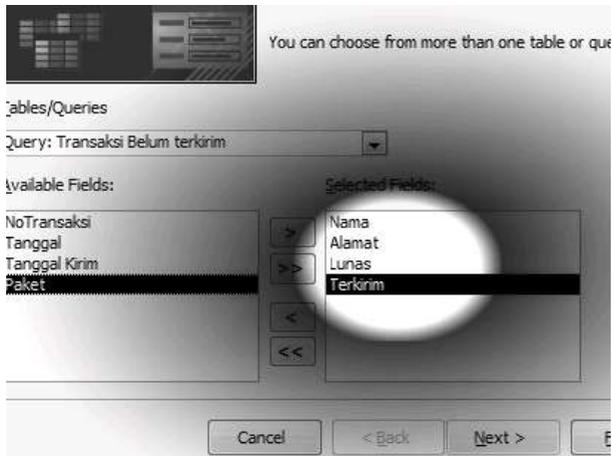
Sedangkan untuk field yang disertakan ke dalam query, dapat dilihat pada gambar berikutnya. Khusus untuk field *Terkirim* dari tabel *Transaksi* dimodifikasi di ranah kriteria agar hanya tampil yang isi dari record field tersebut yang bernilai *False* atau belum terkirim (dan secara otomatis juga memiliki status belum lunas). Jangan lupa untuk melakukan penyimpanan terhadap query yang telah dibuat, dalam kasus ini disimpan dengan nama *Transaksi Belum Terkirim*.

NoTransaksi Transaksi	Tanggal Transaksi	Tanggal Kirim Transaksi	Nama Customer	Alamat Customer	Paket Detail Transaksi	Lunas Transaksi	Terkirim Transaksi
<input checked="" type="checkbox"/>							
							False

Gambar 109. Opsi Field Query

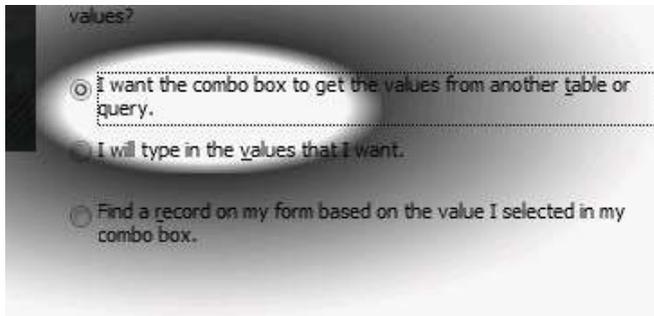
Berikutnya kita dapat melanjutkan ke pembuatan form untuk pelunasan. Langkah pembuatan form tersebut dapat mengikuti langkah-langkah berikut:

1. Buat form dengan menggunakan fasilitas wizard, kemudian ambil query dari yang baru saja dibuat yaitu query *Transaksi Belum Terkirim*. Namun di dalam form, nantinya hanya melibatkan empat field terlebih dulu yakni field *Nama*, *Alamat*, *Lunas* dan *Terkirim*. Nantinya untuk field *Lunas* dan *Terkirim* secara otomatis diterjemahkan menjadi komponen checkbox karena memiliki tipe *yes/no* atau *boolean*.



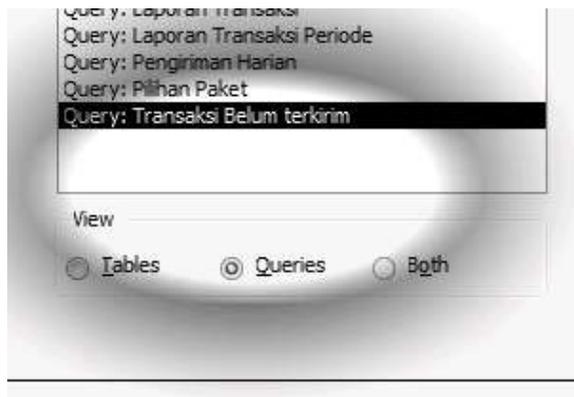
Gambar 110. Field Form

2. Selanjutnya tempatkan satu combobox ke dalam form tersebut. Di dalam kotak dialog wizard yang muncul, pilih opsi yang nantinya memilih data yang berada dalam combobox berasal dari query.



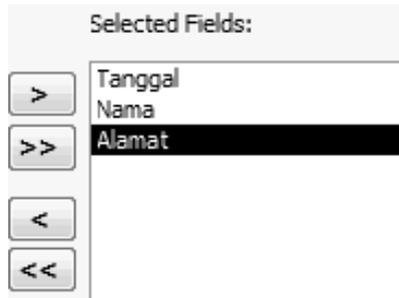
Gambar 111. Opsi Combo Form

3. Kemudian pilih data yang berasal dari query *Transaksi Belum Terkirim*. Ini berarti bahwa di dalam form tersebut, sebenarnya seluruh data juga berasal dari query yang sama, namun dimodifikasi pada saat pengguna melihat tampilan form.



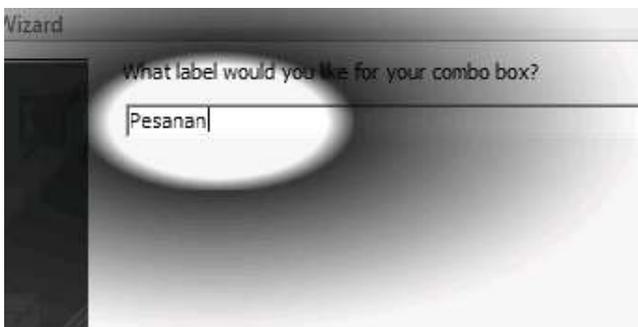
Gambar 112. Pemilihan Query

4. Lalu pilih tiga field yang dimasukkan ke dalam combobox tersebut, yakni field *Tanggal*, *Nama* dan field *Alamat*.



Gambar 113. Pemilihan Field

5. Di isian selanjutnya pada kotak dialog wizard untuk combobox tersebut, isikan teks untuk label keterangan yang nanti terdapat di sebelah kiri dari combobox.





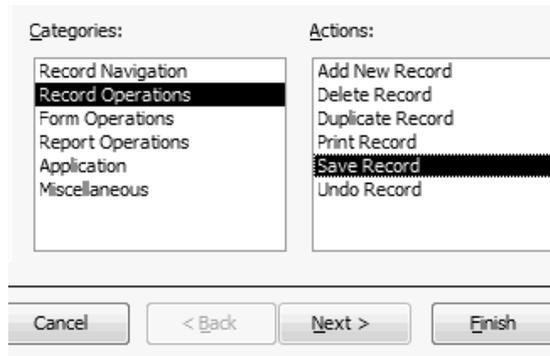
Perhatikan bahwa teknik penggunaan combobox yang menggunakan query yang sebelumnya berasal dari tabel majemuk dapat membuat form kita terlihat lebih “canggih”. Sehingga secara teknis, kemampuan kita untuk menyusun sebuah query merupakan kunci dari pembuatan sistem informasi yang berguna bagi pengguna.

6. Agar tampilan form terlihat lebih rapi, atur kembali susunan komponen pada form sehingga terlihat seperti pada gambar berikut:

The image shows a web form titled "Pelunasan Transaksi" (Transaction Settlement). The form is divided into sections: "Form Header" containing the title, "Detail" containing input fields, and "Form Footer" containing checkboxes and a button. The fields are: "Pesanan" (Order) with a dropdown menu set to "Unbound"; "Nama" (Name) with a text input field; "Alamat" (Address) with a text input field; "Lunas" (Paid) with a checked checkbox; and "Terkirim" (Delivered) with a checked checkbox. A "Simpan" (Save) button is located at the bottom right of the form.

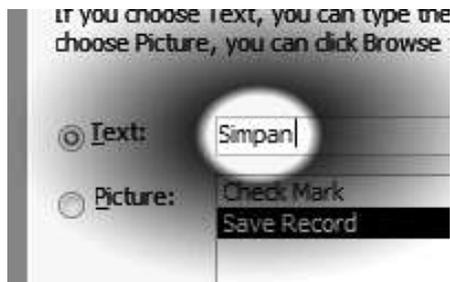
Gambar 114. Desain Form Pelunasan

- Langkah berikutnya adalah menambahkan sebuah button ke dalam form tersebut, dan gunakan fasilitas wizard di dalam proses penambahan button tersebut. Pada kotak dialog tersebut, pilih fitur untuk *record operation* dan pilih opsi untuk penyimpanan data.



Gambar 115. Penempatan Button

8. Berikan teks *Simpan* untuk button yang baru dibuat tersebut. Pemberian teks dapat diberikan di dalam kotak dialog wizard yang tersedia.



Gambar 116. Teks Button Simpan

9. Kini form tersebut telah siap untuk diujicoba dan diimplementasikan ke dalam rangkaian transaksi. Tampilan form tersebut mirip dengan gambar berikut:

Transaksi Belum terkirim

Pelunasan Transaksi

Pesanan			
Nama	7/10/2010	Caca	Jl. Tidar 3
Alamat	7/10/2010	Caca	Jl. Tidar 3
	7/8/2010	Baba	Jl. Tidar 2
<input type="checkbox"/> Lunas	7/8/2010	Baba	Jl. Tidar 2
<input type="checkbox"/> Ter	9/20/2010	Caca	Jl. Tidar 3

Record: 1 of 5 | No Filter | Search

Gambar 117. Uji Coba Form Pelunasan

Pembuatan Report

.....

Langkah selanjutnya dalam rangkaian pembuatan sistem informasi ini adalah membuat laporan. Dalam pembuatan laporan dengan menggunakan Microsoft Access, telah terdapat banyak fitur yang sangat membantu para pengembang dalam membuat laporan yang sesuai dengan kebutuhan para pengguna.

Tetapi langkah penting yang sesungguhnya di dalam pembuatan sebuah laporan adalah memahami apa yang sebenarnya dibutuhkan oleh pengguna. Oleh karena itu, dibutuhkan sebuah analisa dan survei yang mendalam ke pengguna mengenai apa yang sebenarnya dibutuhkan oleh mereka.

Di dalam studi kasus ini, dilakukan pembuatan laporan sebanyak tiga buah dengan tujuan agar dapat diketahui pembuatan laporan dengan variasi teknik yang beragam. Baik dari laporan yang sederhana, hingga laporan yang menggunakan parameter tunggal maupun majemuk.

Diharapkan dari contoh pembuatan laporan yang ada di dalam buku ini nantinya dapat

Studi Kasus: Sistem Informasi Katering

menginspirasi untuk pembuatan laporan yang lebih kompleks dan bermanfaat bagi para pengguna.

Laporan Daftar Menu

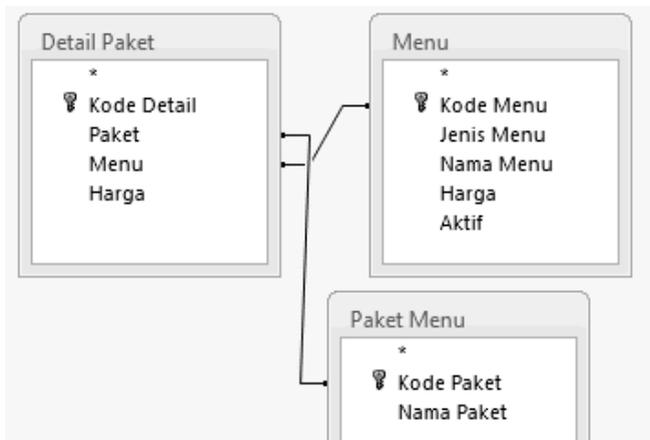
Laporan yang pertama dibuat dengan menggunakan laporan secara murni tanpa bantuan form sama sekali. Laporan tersebut nantinya menampilkan paket menu yang dimiliki oleh catering dan umumnya digunakan untuk brosur sederhana ke para pelanggan.

Sebelum memulai pembuatan laporan, maka terlebih dulu dibuat sebuah query yang melibatkan tiga buah tabel untuk pembuatan laporan tersebut. Pembuatan query dengan menggunakan fitur query design, mirip dengan langkah pembuatan query seperti pada pembuatan form pelunasan transaksi.



Bagi Anda yang merasa pakar di bidang query atau perintah SQL, perlu diingat bahwa keahlian di bidang query adalah sebuah senjata pamungkas di dalam pembuatan sistem informasi dengan menggunakan Microsoft Access.

Setelah masuk ke query design, tambahkan tiga tabel kedalamnya yaitu tabel *Menu*, *Paket Menu* serta tabel *Detail Menu* ke dalam query tersebut.



Gambar 118. Query Design Report

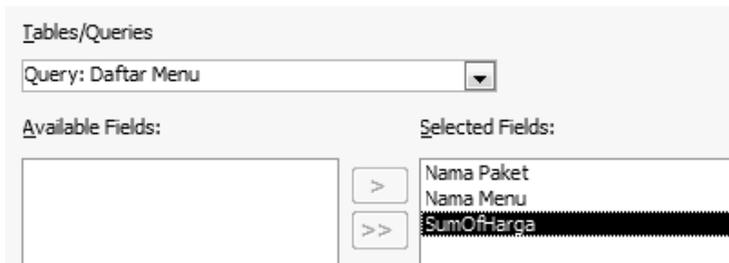
Kemudian dari ketiga tabel tersebut, ambil field yang dibutuhkan di dalam laporan nanti, yakni field *Nama Paket* dari tabel *Paket Menu*, field *Nama Menu* dari tabel *Menu* serta field *Harga* dari tabel *Menu*. Khusus untuk field *Harga* yang berupa field numerik, aplikasikan perhitungan penjumlahan atau *sum* agar nantinya muncul harga paket menu di dalam laporan tersebut.

Field:	Nama Paket	Nama Menu	Harga
Table:	Paket Menu	Menu	Detail Paket
Total:	Group By	Group By	Sum
Sort:			
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:			

Gambar 119. Pemilihan Field Query

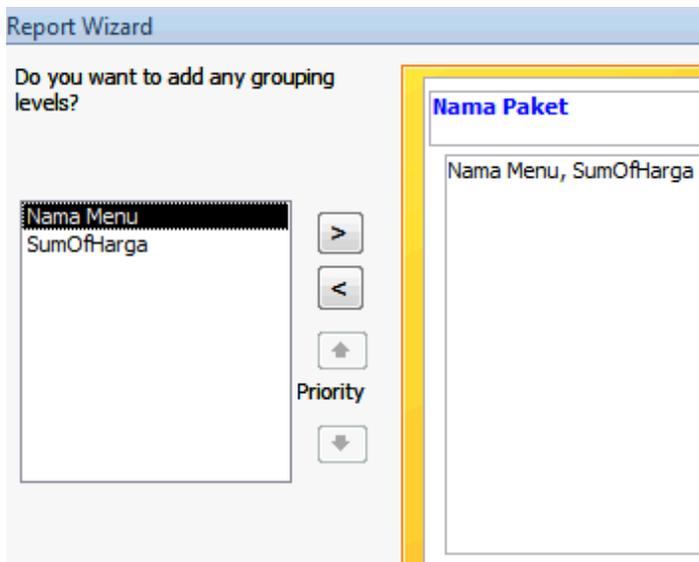
Langkah terakhir dari pembuatan query tersebut adalah menyimpan hasil query dengan nama *Daftar Menu*. Kini dapat dilanjutkan dengan pembuatan laporan dengan mengikuti langkah-langkah berikut ini:

1. Buat report dengan fasilitas wizard dengan memilih ikon yang ada pada toolbar. Pada kotak dialog yang tersedia pilih query yang baru saja dibuat yakni dari query *Daftar Menu*.



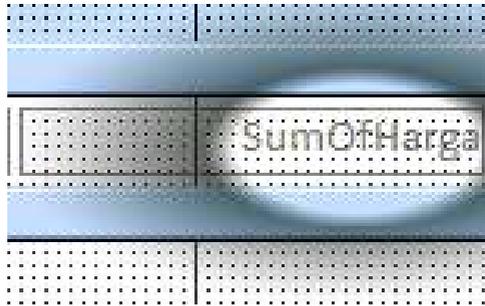
Gambar 120. Pilihan Field Query

2. Selanjutnya tambahkan group pada laporan dengan menggunakan field *Nama Paket* sebagai referensi utama dari grup laporan yang dibuat. Penggunaan field ini nantinya berpengaruh terhadap hasil penjumlahan pada field *sumOfHarga* yang mengkalkulasi harga dari tiap menu per paket yang telah disediakan.



Gambar 121. Report Wizard

3. Setelah kotak dialog wizard selesai ditutup (hingga akhir), maka selanjutnya edit *header* dari field *sumOfHarga* menjadi teks *Harga* untuk penyesuaian tampilan laporan.



Gambar 122. Label SUM

4. Jika sudah selesai dilakukan, maka laporan telah siap digunakan, dan untuk lengkapnya bisa dilihat dalam mode preview seperti pada gambar berikut:

Daftar Menu

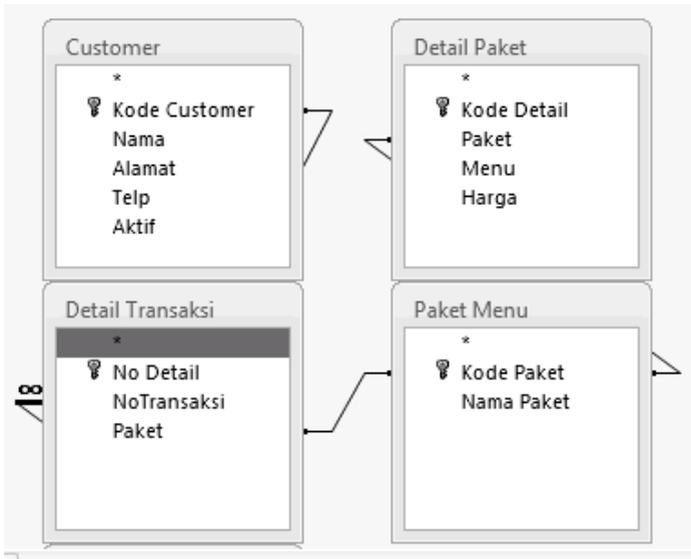
Paket 1	6,000
Buah kecil	
Es Jeruk	
Nasi Pecel	
Paket 2	5,500
Es Teh	
Nasi Rawon	
Puding kecil	
Paket 3	6,000
Buah kecil	

Gambar 123. Preview Report

Laporan Transaksi Per Periode

Laporan yang kedua di dalam sistem informasi ini menggunakan bantuan form sebagai penampung parameter untuk pembuat laporan dalam periode tertentu. Sama seperti pada pembuatan laporan yang pertama, maka pembuatan laporan ini didahului dengan pembuatan query melalui fitur query design.

Setelah masuk ke kotak dialog untuk pembuatan query, tambahkan empat tabel yang nantinya dibutuhkan bagi pembuatan tersebut. Tabel tersebut adalah tabel *Customer*, *Paket Menu*, *Detail Paket* dan tabel *Transaksi*.



Gambar 124. Query Design Report Transaksi

Sedangkan untuk field yang ditampilkan di dalam query tersebut adalah field *Tanggal*, *Terkirim* dan *Tanggal Kirim* yang berasal dari tabel *Transaksi*, field *Nama* dan *Alamat* yang berasal dari tabel *Customer*, field *Nama Paket* dari tabel *Paket Menu* serta field *Harga* dari tabel *Detail Paket*.

Field:	Tanggal Transaksi	Tanggal Kirim	Nama Customer	Alamat Customer	Nama Paket Paket Menu	Harga Detail Paket	Terkirim Transaksi
Table:	Transaksi	Transaksi	Customer	Customer	Paket Menu	Detail Paket	Transaksi
Total:	Group By	Sum	Group By				
Sort:							
Show:	<input checked="" type="checkbox"/>						

Gambar 125. Field Report

Khusus untuk field *Tanggal* yang dijadikan sebagai parameter di dalam pembuatan laporan, maka disertakan parameter yang ditempatkan ke dalam isian *criteria*. Pada isian tersebut, ketikkan parameter berikut:

```
>=[Forms]![Laporan Transaksi Per Periode]![Text0] And  
<=[Forms]![Laporan Transaksi Per Periode]![Text2]
```



Yang perlu diperhatikan di dalam pengisian parameter yang diisikan nilainya dari sebuah form adalah mengenai perencanaan penamaan form dan laporan. Pada saat pembuatan parameter di query ini, sebagai contoh, perencanaan nama form dan juga nama komponen yang dijadikan penampung parameter telah dibuat pada saat query dalam proses pembuatan.

Jika sudah selesai, maka jangan lupa untuk menyimpan query tersebut agar bisa digunakan ke dalam laporan. Dalam studi kasus ini query tersebut diberi nama *Laporan Transaksi Periode*. Berikutnya adalah pembuatan form dan laporan yang menjadi satu rangkaian laporan dengan penggunaan parameter.

Untuk pembuatan keduanya, dapat dilakukan langkah-langkah berikut:

1. Buat terlebih dulu sebuah laporan dengan menggunakan fasilitas wizard. Sebagai acuan, ambil dari query yang telah dibuat sebelumnya yakni dari query *Laporan Transaksi Periode*. Masukkan seluruh field yang ada kecuali field *Terkirim* yang di dalam query tersebut lebih berfungsi untuk penampung parameter.



Gambar 126. Opsi Field Report

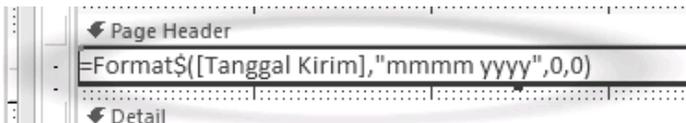
2. Setelah seluruh rangkaian kotak dialog wizard selesai dilalui, maka selanjutnya dari hasil desain report yang tampil, modifikasi agar menjadi tampilan seperti pada gambar berikut ini:

Detail			
Tanggal	Nama	Alamat	
Tanggal Kirim	Nama Paket		SumOfHarga

Gambar 127. Desain Form untuk Report

Modifikasi yang dilakukan adalah dengan menghapus semua keterangan yang ada di bagian header report kecuali header untuk field *Tanggal* dan *Tanggal kirim*. Kemudian untuk field lainnya disusun secara bertumpuk satu sama lain.

3. Berikutnya tambahkan sebuah teks di bagian *Page Header* tambahkan sebuah field yang nantinya menampilkan bulan dari tanggal pengiriman yang ada. Karena sesungguhnya tidak ada field khusus untuk bulan, maka untuk menampilkan hal tersebut cukup dengan menggunakan fungsi format dengan meniadakan format tanggal dan memanjangkan format untuk bulan. Lebih lengkapnya dapat dicontoh pada gambar berikut :



Gambar 128. Isian Format Tanggal

4. Langkah selanjutnya adalah memasukkan komponen sub report di dalam laporan tersebut. Teknik ini digunakan jika di dalam sebuah laporan yang sebelumnya telah menggunakan tabel atau query, kemudian ingin ditambahkan sebuah kumpulan data yang berasal dari tabel lain. Dalam kasus ini, sub laporan yang ingin ditambahkan berasal dari tabel *Setting* yang berfungsi untuk identitas dari laporan tersebut.

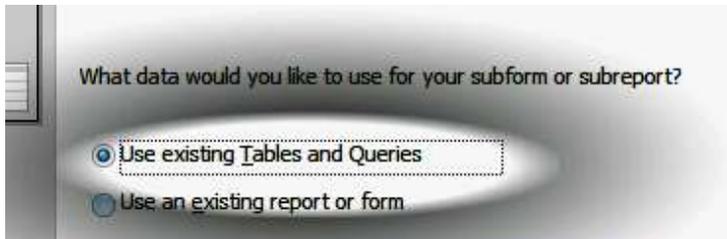


Gambar 129. Opsi Subreport



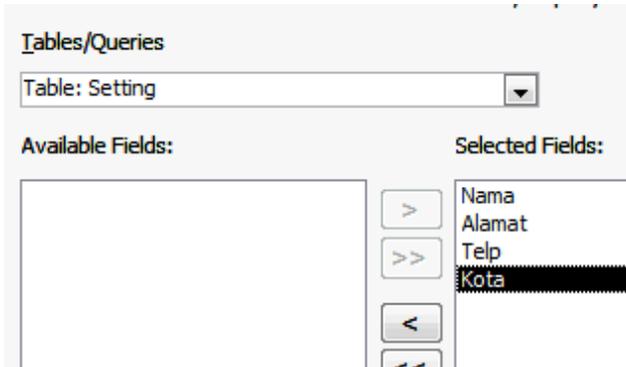
Dalam sebuah laporan yang dimasukkan ke dalam sistem informasi, terdapat beberapa hal yang wajib ada di dalam sebuah laporan, antara lain: identitas perusahaan, nomor halaman dan judul laporan itu sendiri. Di dalam kasus ini, identitas perusahaan ditentukan di dalam tabel *Setting* sehingga dapat dimodifikasi sesuai dengan keinginan, terutama jika dibuat sebagai produk komersial yang massal.

5. Pada saat muncul kotak dialog wizard yang ada untuk memasukkan sub laporan, pilih opsi pengambilan dari tabel yang sudah ada.



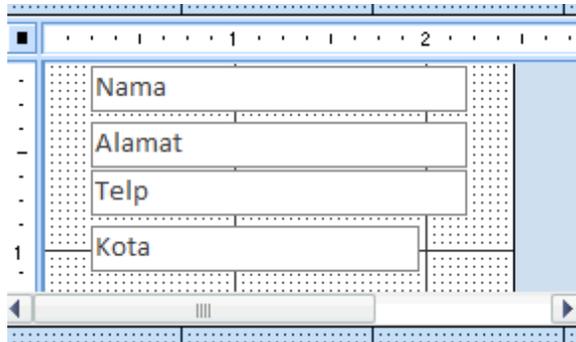
Gambar 130. Opsi Penggunaan Query

6. Lalu pilih tabel *Setting* dan ambil seluruh field yang ada di dalam tabel tersebut. Jangan lupa untuk menempatkan sub report tersebut di dalam bagian *Header* dari laporan yang utama.



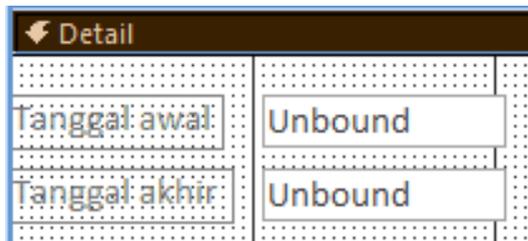
Gambar 131. Field Tabel Setting

7. Dari sub report yang sudah terbentuk, atur sedemikian rupa sehingga terlihat seperti pada gambar berikut:



Gambar 132. Pengaturan Awal

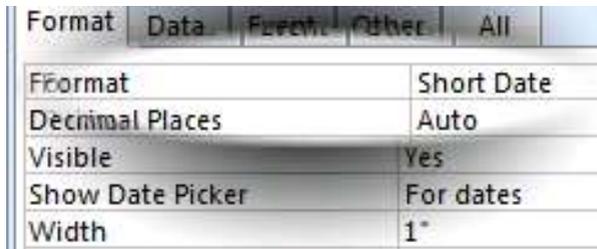
8. Setelah semua laporan selesai diatur maka simpan dengan nama *Laporan Transaksi Periode*.
9. Buatlah sebuah form tanpa menggunakan fasilitas wizard, dan tempatkan dua buah textbox ke dalam form tersebut, dengan teks pada label diisi nilai *Tanggal Awal* dan *Tanggal Akhir*.



Gambar 133. Isian Tanggal

10. Selanjutnya di property textbox yang baru saja ditempatkan, set property *Format* ke nilai *short*

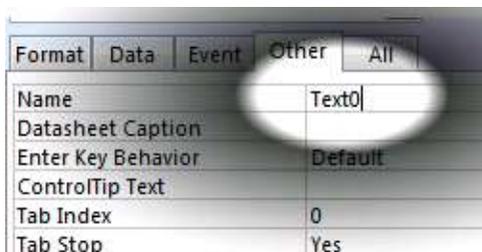
date. Dengan melakukan setting tersebut, maka secara otomatis nanti menampilkan *date picker* pada saat pengguna berada di dalam textbox tersebut.



Format	Data	Event	Other	All
Format			Short Date	
Decimal Places			Auto	
Visible			Yes	
Show Date Picker			For dates	
Width			1"	

Gambar 134. Format Short Date

11. Berikutnya pastikan bahwa property *Name* dari textbox yang pertama adalah *Text0* dan textbox yang kedua adalah *Text2*. Hal ini dilakukan agar nantinya sesuai dengan parameter query yang sebelumnya telah dibuat di awal bab ini.

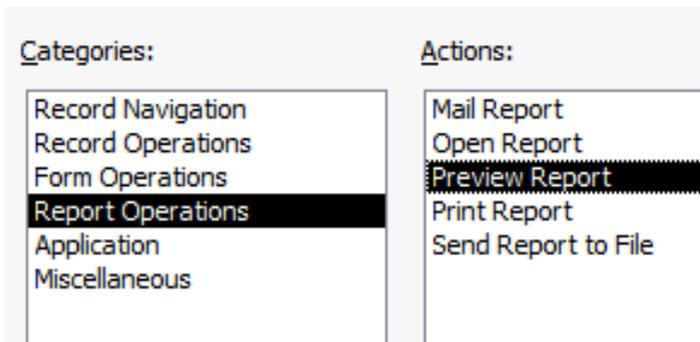


Format	Data	Event	Other	All
Name			Text0	
Datasheet Caption				
Enter Key Behavior			Default	
ControlTip Text				
Tab Index			0	
Tab Stop			Yes	

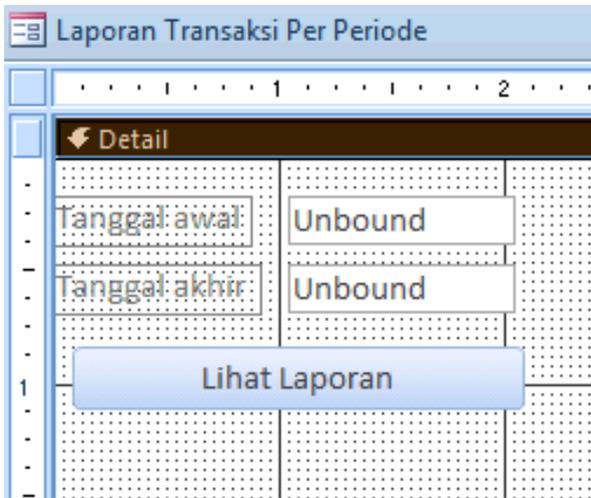
Gambar 135. Opsi Nama Textbox Report



Kemudian tempatkan sebuah button di bawah kedua textbox tersebut, dan di dalam kotak dialog yang muncul untuk textbox tersebut pilih opsi untuk operasi menampilkan report. Perlu diperhatikan bahwa opsi *Preview* dan *Print* berbeda dari sisi tampilan. Opsi *Preview* lebih bersifat tampilan ke layar, sedangkan *Print* berdasarkan kesiapan dari laporan untuk dicetak ke printer.

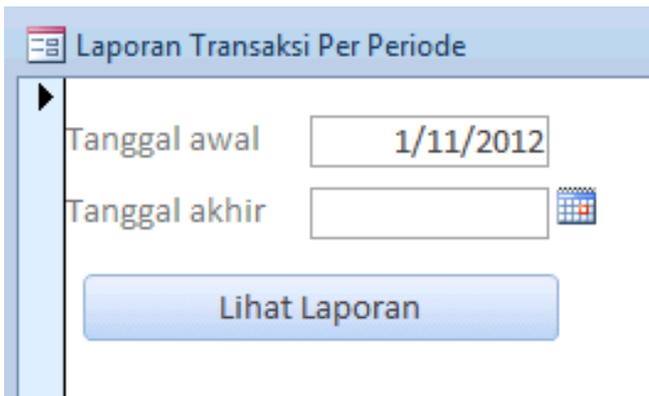


Gambar 136. Button Preview



Gambar 137. Desain Laporan Transaksi

12. Jika selesai, maka tampilan dari form terlihat seperti pada gambar berikut:



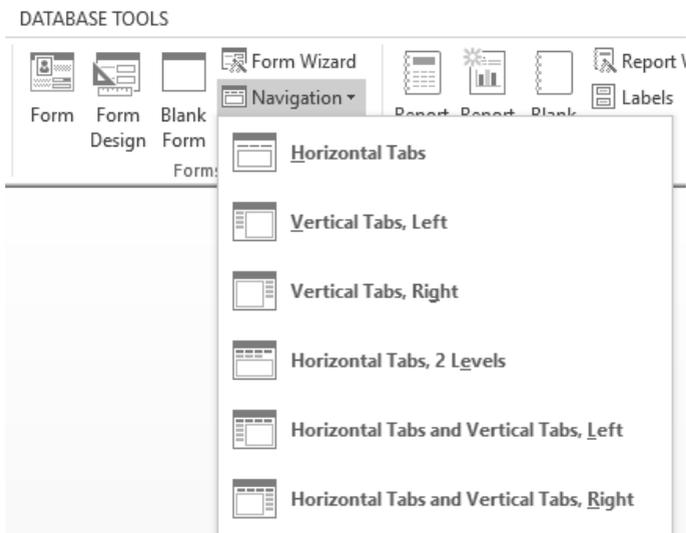
Gambar 138. Uji Coba Laporan Transaksi

Navigasi Aplikasi

.....

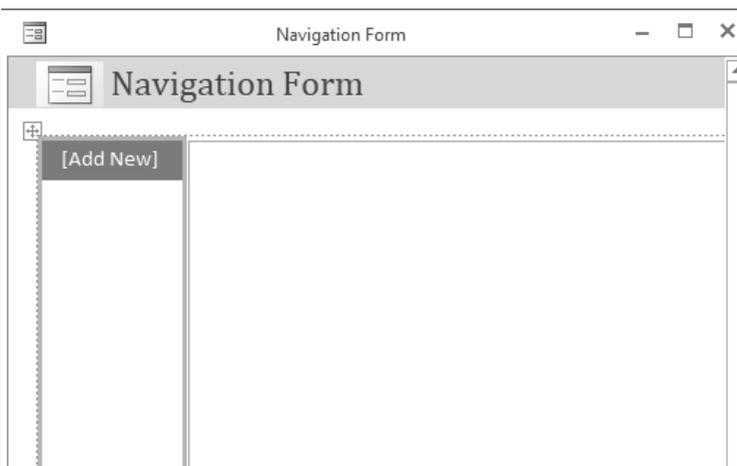
Langkah terakhir dari pembuatan sistem informasi ini adalah melakukan pembuatan form untuk navigasi aplikasi. Pada Microsoft Access versi 2010 ke bawah, form navigasi biasa disebut sebagai *switch board*. Namun pada versi berikutnya, form tersebut diberi nama sesuai dengan fungsinya yakni form navigasi.

Pembuatan form untuk navigasi aplikasi dilakukan dengan memilih ikon *Navigation*



Gambar 139. Menu Navigasi

Selanjutnya, dapat dilakukan pemilihan jenis form navigasi yang diterapkan pada sistem informasi. Pada studi kasus ini, dipilih model *Vertical Tabs, Left*. Ini berarti bahwa pada form navigasi tersebut nantinya memiliki menu yang terletak di sebelah kiri dan form yang dieksekusi terletak di sebelah kanan. Hasil awal dari pemilihan jenis form navigasi tersebut tampak pada gambar berikut:



Gambar 140. Form Navigasi Awal

Berikutnya dapat dilakukan drag form yang ditempatkan pada menu tersebut. Urutan dari menu dapat disesuaikan, baik dari form maupun report atau laporan yang sudah dibuat sebelumnya.



Urutan menu pada dasarnya tidak memiliki rumus baku, apakah berdasarkan kronologi atau berdasarkan fungsi yang umum digunakan. Namun demikian, umumnya urutan menu di sistem informasi berdasarkan data yang harus dimasukkan terlebih dulu oleh pengguna, misal: data yang bersifat master (customer, barang) lalu disusul dengan data transaksi.

Berikut adalah hasil drag form dan laporan yang telah dibuat sebelumnya (sebagai catatan, bahwa beberapa form tidak dijelaskan di dalam buku, karena diasumsikan memiliki langkah yang sama dengan form lain).

Setiap hasil drag form menghasilkan *button* yang secara otomatis menampilkan form yang dimaksud saat pengguna melakukan klik terhadap button tersebut. Hal ini mempermudah pengembang sistem informasi saat



Gambar 141. Desain Menu

Kini saatnya melakukan testing terhadap form navigasi tersebut, klik salah satu menu yang tersedia sehingga muncul form yang dimaksud. Sebagai contoh adalah percobaan untuk form *Paket Menu* yang muncul di bagian kiri.



Gambar 142. Desain Akhir Form Navigasi



Jika pada saat testing form navigasi terjadi kesalahan, terutama jika salah satu menu yang terdapat di bagian kiri tidak berfungsi, maka cukup dilakukan proses *drag* ulang form yang tersedia ke dalam form. Anda tidak perlu lagi melakukan proses setting *property* atau pembuatan listing, kecuali sangat diperlukan.

Berikutnya adalah melakukan proses setting aplikasi melalui menu *File* → *Options*.

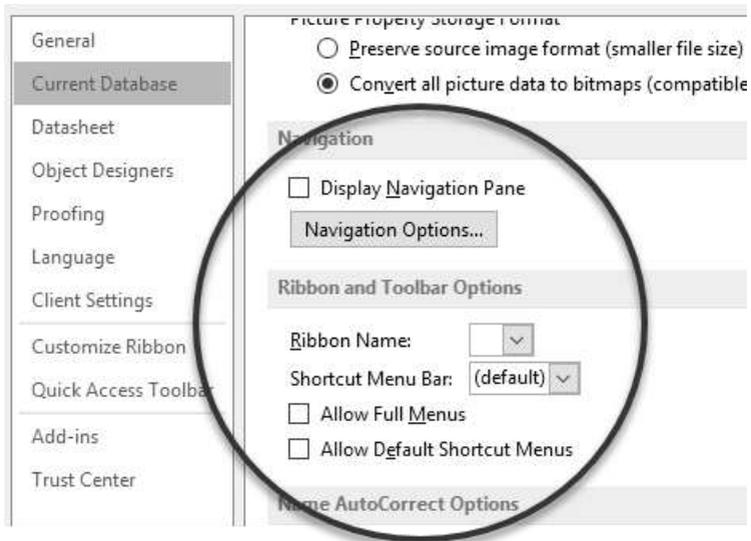


Gambar 143. Setting Aplikasi

Pada kotak dialog yang tersedia, terdapat beberapa setting yang perlu diisi ulang sebelum sistem informasi yang dibuat siap didistribusikan ke pengguna.

Pada kotak dialog tersebut, terdapat isian mengenai tampilan untuk *Navigation Pane* yang berarti jika opsi tersebut tidak dipilih, maka seluruh rangkaian menu navigasi database yang terletak di sebelah kiri, disembunyikan.

Opsi berikutnya yang diperhatikan adalah opsi untuk *Ribbon and Toolbar Options*. Pada opsi ini, kita hilangkan seluruh opsi yang ada, yakni untuk *full menu* dan *shortcut*. Hal ini mengakibatkan seluruh menu yang terdapat pada ribbon di bagian atas menghilang, begitu pula dengan menu yang terdapat pada bagian *home*.



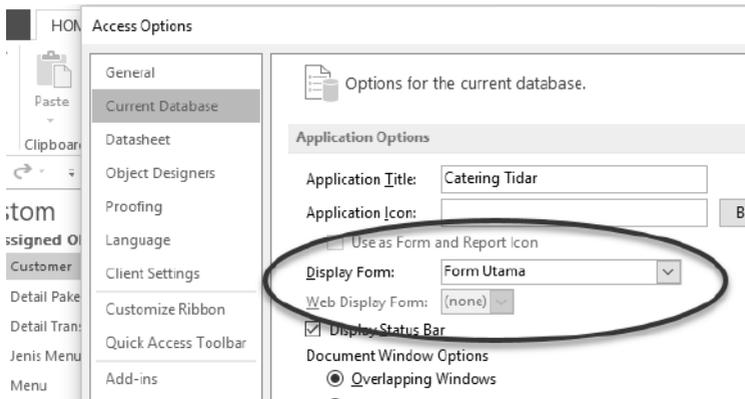
Gambar 144. Opsi Navigasi



Disarankan agar sebelum melakukan langkah-langkah yang ada di bagian ini, database yang sudah terbentuk dapat terlebih dulu dibackup dengan cara menyimpan ulang dengan nama yang berbeda. Sebab jika terjadi kesalahan pada langkah yang ada, dapat menyebabkan database tidak bisa diedit lagi.

Lalu pastikan bahwa form navigasi yang sebelumnya dibuat, menjadi form pertama yang tampil

saat file database dibuka oleh pengguna. Hal ini dapat dilakukan dengan membuka menu *File* → *Options*, kemudian pada kotak dialog yang tersedia pilih isian *Display Form* dan arahkan pada form navigasi yang telah dibuat.



Gambar 145. Opsi Display

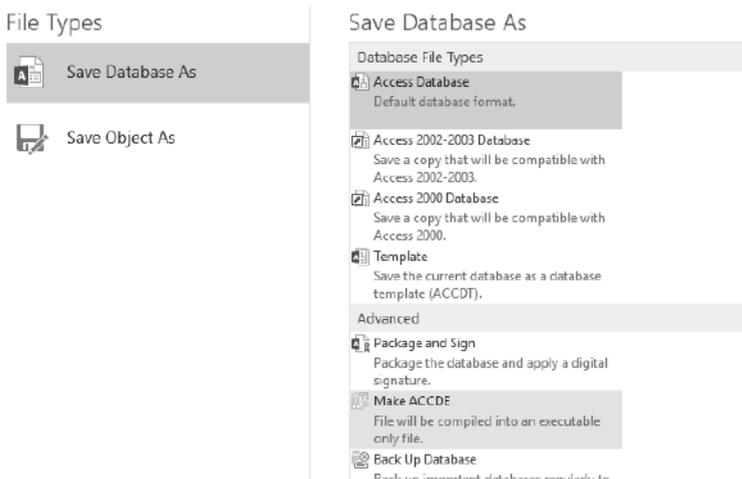


Pada contoh ini, form navigasi yang dijadikan contoh telah diberi nama *Form Utama* untuk kemudahan pemrosesan data.

Berikutnya, lakukan penyimpanan ulang database dengan format AACDE. Ini berarti bahwa database baru yang disimpan ulang, menjadi *executable* atau siap untuk didistribusikan ke klien atau

customer. Sehingga pada saat dieksekusi, seluruh menu dan opsi tabel disembunyikan, sehingga pihak pengguna hanya dapat melihat form navigasi dan menjalankan menu yang telah tersedia saja.

Save As



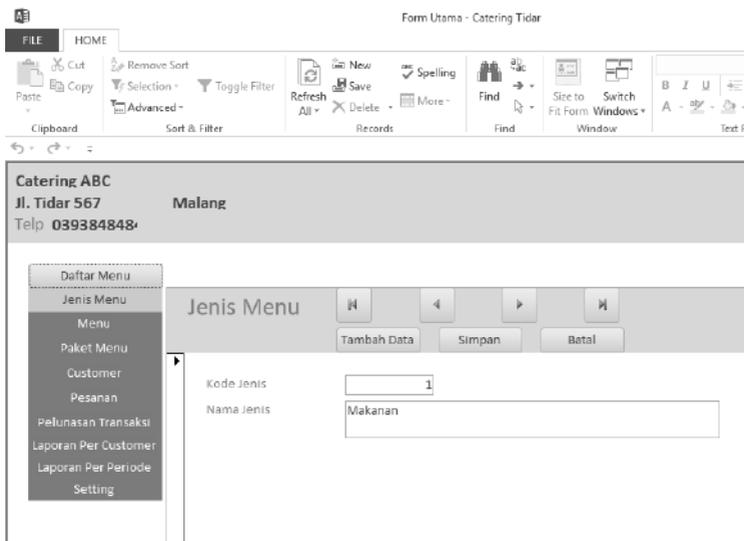
Gambar 146. Menu Ekspor Package



Pastikan bahwa nama database baru yang telah dikompilasi dan siap dieksekusi (format AACDE), memiliki nama yang berbeda dengan file yang asli agar tidak *overwrite*.

Studi Kasus: Sistem Informasi Katering

Setelah seluruh setting dan form navigasi selesai dibuat, maka langkah berikutnya Microsoft Access dapat ditutup dan kemudian, file baru format AACDE dapat dieksekusi (dobel klik). Contoh tampilan database yang telah terkompilasi, tampak pada gambar berikut:



Gambar 147. Kompilasi Database

Dapat ditelaah, bahwa menu yang tersedia telah hilang dan pengguna hanya dapat memilih dan mengeksekusi form yang sudah ada. Sehingga tidak ada lagi modifikasi tabel (baik struktur maupun data), kecuali dari form yang tersedia.

Daftar Pustaka

- Couch, A. (2010) *Microsoft Access 2010, VBA Programming, Network*. Redmond: Microsoft Press.
- Cox, J. and Lambert, J. (2013) *Microsoft Access 2013*. Redmond: Microsoft Press. doi: 10.1017/CBO9781107415324.004.
- Darwen, H. (2010) *An Introduction to Relational Database, Database*. BookBoon. Available at: <http://zums.ac.ir/files/research/site/ebooks/it-programming/an-introduction-to-relational-database-theory.pdf> (Accessed: 20 May 2017).
- Jennings, R. (2011) *Access 2010*. Indianapolis: Wiley. Available at: http://ptgmedia.pearsoncmg.com/imprint_downloads/que/bookreg/9780789743077/MSAccess2010InDepth_9780789743077.pdf.
- Microsoft (2016) Microsoft Official Academic Course: MICROSOFT ACCESS 2016, Access. Redmond: Wiley.
- Silberschatz, A., Korth, H. F. and Sudarshan, S. (2011) *Database System Concepts - 6th. ed., Database*. New York: McGraw-Hill. doi: 10.1145/253671.253760.
- Silberschatz, A., Korth, H. F. and Sudarshan, S. (2011) *Database System Concepts - 6th. ed., Database*. New York: McGraw-Hill. doi: 10.1145/253671.253760.
- Tutorials Point (2013) *Database Management System [DBMS] Tutorial*. Tutorials Point. Available at: https://www.tutorialspoint.com/dbms/dbms_tutorial.pdf (Accessed: 20 May 2017).

- University of Reading (2016) *Microsoft Access 2013™ An Essential Guide (Level 1)*, webpage.
Reading: University of Reading. Available at:
<https://www.reading.ac.uk/web/files/its/AccessEssen2013.pdf> (Accessed: 20 May 2017).
- Viescas, J. L. (2005) *Building Microsoft Access Application*. Redmond: Microsoft Press.

Glosarium

Visual Basic for Application	integritas antar aplikasi serta pemrograman yang <i>menempel</i> di dalam aplikasi
Microsoft Access	keluarga dari Microsoft Office yang memiliki fungsi sebagai DBMS (<i>Database Management System</i>)
Sistem database	kolaborasi dari file yang saling berelasi dan sebuah set aplikasi yang memungkinkan untuk memodifikasi file tersebut.
DBMS (DataBase Management System)	sebuah perangkat lunak yang bertujuan untuk mengatur dan mengorganisasi sistem database.
Data model	struktur dari data yang merepresentasikan tujuan tertentu dalam bentuk yang logis.

Physical level	Merupakan level terendah yang menyatakan bagaimana data itu benar-benar disimpan dalam media penyimpanan.
Logical level	Level ini menyatakan data apa yang disimpan dalam database dan relasi antar data yang disimpan tersebut
View level	Merupakan level abstraksi data tertinggi yang hanya menampilkan data sesuai yang diinginkan oleh pengguna
Entity	obyek yang dikenali oleh pengguna
Primary key	Field yang bersifat unique

Pemrograman Visual
aplikasi database dengan
Visual Basic .NET



Soetam Rizky

Sistem Operasi

Rudy Setiawan

Learning By Sample
**PHP+
using ASP .NET**



**Hands On Lab :
SQL
Fundamen**

Soetam Rizky Wicaksono

Learning By Sar
ASP .NET

**Material
Teknik**
Purnomo

ISBN 978-602-52194-2-9



9 786025 219429

**Seribu
Bintang**

 www.SeribuBintang.co.id
 info@SeribuBintang.co.id
 fb.com/cv.seribu.bintang