**Menu Dan Panduan Penggunaan**

**RANCANG BANGUN SISTEM INFORMASI PERSEDIAAN BARANG MENGGUNAKAN METODE QUICK MODELLING (METODE SECARA CEPAT) PADA ADVANCE DI BEKASI**

Perkembangan zaman saat ini, menjadikan teknologi sangat berkembang dengan pesat, salah satunya adalah teknologi komputer. Kehadiran komputer sendiri mempunyai pengaruh positif yang sangat signifikan terutama di dunia kerja. Pengolahan data yang baik tentu akan menghasilkan informasi yang cepat, tepat dan akurat serta dapat mengatasi masalah-masalah yang terjadi, masalah yang sering timbul ketika pengumpulan data masih dilakukan dengan cara pengolahan data yang masih dilakukan dengan cara manual. Advance merupakan salah satu perusahaan penjualan yang masih menggunakan sistem manual dalam pencatatan stok barang mereka. Pencatatan dilakukan dengan manual dengan cara mendata semua barang yang ada. Metode pengembangan sistem pada aplikasi ini menggunakan metode model secara cepat (*Quick Modelling*) yang terdiri daribeberapa tahapan. Adapun tahapan tersebut yaitu analisis, desain, pengkodean, pengujian dan pemeliharaan sistem. Perancangan yang dibuat oleh penulis ini dimaksudkan untuk membantu admin/sales dalam pencatatan persediaan barang. Dalam sistem ini admin dapat mengetahui ataumelihat transaksi, data barang dan stok barang yang dibutuhkan melalui sistem yang sudah dibuat ini.

**DESAIN PROGRAM**

Dibawah ini adalah rancangan Halaman Program

1. Halaman *Login*



Gambar 1. Halaman *Login*

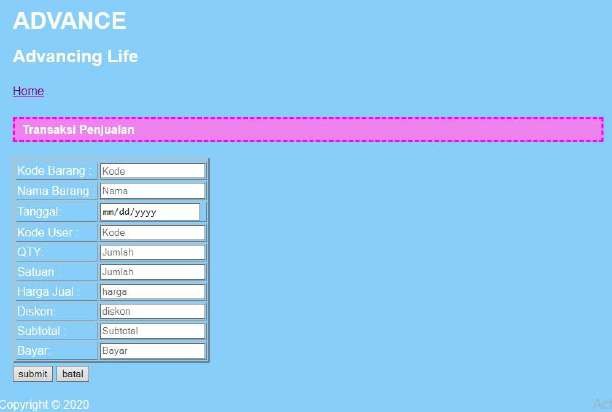
Halaman login ini digunakan oleh Admin dan Kasir Ketika login ke system, Ketika masuk halaman ini admin atau user lainya di arahkan untuk mengisi Username dan password yang sudah ditentukan, selanjutnya pilih Level User yaitu untuk level “1” Administrator , Level “2” Pimpinan Perusahaan dan Level “3” Admin dengan hak akses yang berbeda-beda setiap levelnya.

1. Halaman Utama (*Home*)

Gambar 2. Halaman Utama (*Home*)

Halaman Home ini adalah halaman awal Ketika masuk ke system, pada halaman home ini adalah halaman awal untuk alur masuk kebeberapa halaman diantaranya Halaman Transaksi, Halaman Barang, Halaman Return, Halaman Pembeli, Halaman User, Halaman Laporan dan Halaman Pegawai.

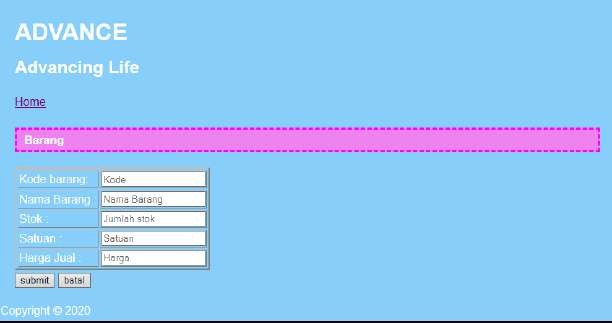
1. Halaman Transaksi Penjualan



Gambar 3. Transaksi Penjualan

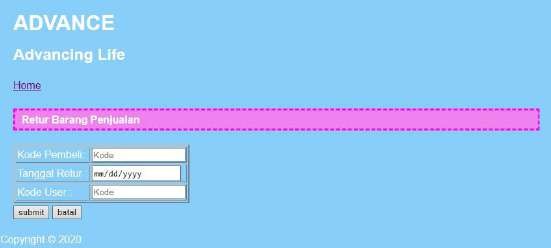
Halaman transaksi penjualan digunakan sebagai halaman untuk proses transaksi Penjualan barang dimana satuan barangnya terhubung dengan stok barang sehingga Ketika terjadi transaksi stok barang akan berkurang secara otomatis

1. Halaman Barang



Gambar 4. Barang

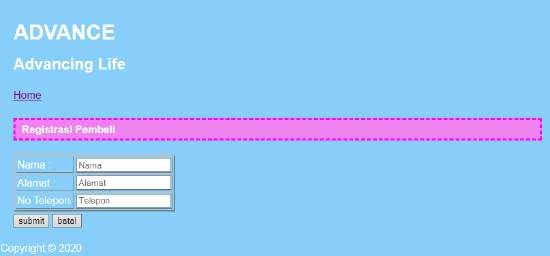
Halaman Barang digunakan sebagai halaman yang digunakan input barang yang tersedia pada perusahaan

1. Halaman Retur Penjualan

Gambar 5. Retur Penjualan

Halaman Retur penjualan diguanakan sebagai halaman untuk melakukan return barang.

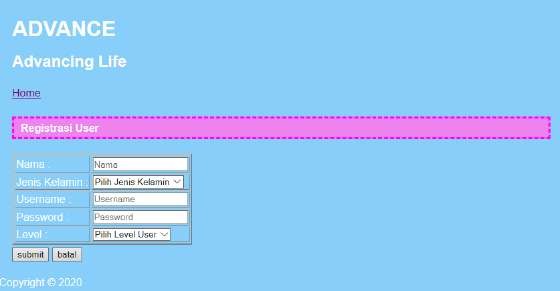
1. Halaman Pembeli



Gambar 6. Pembeli

Halaman pembeli digunakan untuk menyimpan data pembeli

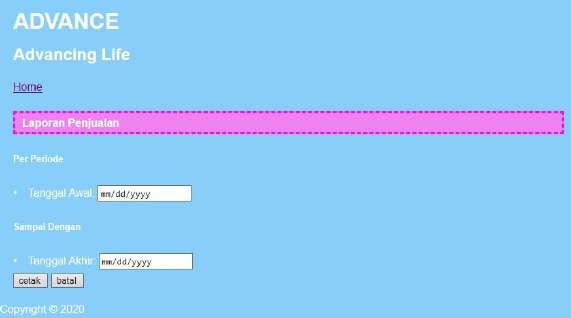
1. Halaman User



Gambar 7. User

Halaman user digunakan untuk menyimpan data user dan penentuan Level User

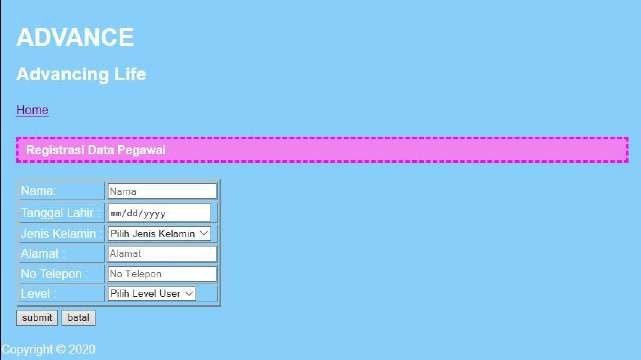
1. Halaman Laporan



Gambar 8. Laporan

Halaman Laporan digunakan untuk mencetak laporan berdasarkan tanggal bulan dan tahun

1. Pegawai



Gambar 9. Pegawai

Halaman Pegawai digunakan untuk penginputan dan penyimpanan data pegawai serta penetuan level user untuk pegawai.

**TABEL**

Dibawah ini sfesifikasi file yang digunakan pada Program

1. Spesifikasi File *User*

Nama File : *User*

Akronim : *user*

Fungsi : Untuk menyimpan data user

Tipe File : File Master

Organisasi File : Index Sequential

Akses File : Random

Media : Hard disk

Panjang Record : 75 byte

Kunci Field :kd\_user

Software : Xampp

Tabel 1. Spesifikasi File *User*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Elemen Data | Nama *Field* | Tipe Data | Size | Keterangan |
| 1 | Kode *User* | *kd\_user* | *Varchar* | 5 | *Primary Key* |
| 2 | *Username* | *username* | *Varchar* | 20 |  |
| 3 | *Password* | *password* | *Varchar* | 15 |  |
| 4 | Nama *User* | nama\_user | *Char* | 25 |  |
| 5 | Level *User* | *level\_user* | *Enum (‘1’,’2’,’3’)* | 10 |  |

1. Spesifikasi File Pembeli

Nama File : Pembeli

Fungsi : Untuk menyimpan data pembeli Tipe File : File Master

Organisasi File : Index Sequential

Akses File : Random

Media : Hard disk

Panjang Record : 112 Byte

Kunci Field : kd\_pembeli

Software : Xampp

Tabel 2. Spesifikasi File Pembeli

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Elemen Data | Nama Field | Tipe Data | Size | Keterangan |
| 1 | Kode Pembeli | kd\_ pembeli | *Varchar* | 5 | *Primary Key* |
| 2 | Nama Pembeli | nama\_ pembeli | *Char* | 20 |  |
| 3 | Alamat | Alamat | *Varchar* | 75 |  |
| 4 | No Telepon | no\_telepon | *Int* | 12 |  |

1. Spesifikasi File Transaksi

Nama File : Transaksi Penjualan

Akronim : transaksi\_jual

Fungsi : Untuk menerangkan tentang transaksi penjualan

Tipe File : File Transaksi

Organisasi File : *Index Sequential*

Akses File : *Random*

Media : *Hard disk*

Panjang Record : 40 *byte*

Kunci field : no\_nota

Software : Xampp

Tabel 3. Spesifikasi File Pembeli

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Elemen Data | Nama Field | Tipe Data | Size | Keterangan |
| 1 | No SO | no\_so | Int | 30 | *Primary Key* |
| 2 | Tanggal Penjualan | tgl\_penjualan | Date |  |  |
| 3 | Kode Pelanggan | kd\_pembeli | Varchar | 5 | *Foreign Key* |
| 4 | Kode *User* | *kd\_user* | Int | 5 | *Foreign Key* |

1. Spesifikasi File Detail Transaksi

Nama File : Detail Transaksi Penjualan

Akronim : Detail\_transaksi\_jual

Fungsi : Untuk lebih menerangkan data-data yang ada dalam transaksi penjualan

Tipe File : File Transaksi

Organisasi File : *Index Sequential*

Akses File : Random

Media : Hard disk

Panjang Record : 143 byte

Kunci field : -

Software : Xampp

Tabel 4. Spesifikasi File Pembeli

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Elemen Data | Nama Field | Tipe Data | Size | Keterangan |
| 1 | No Sales Order | no\_so | Int | 25 | Foreign Key |
| 2 | Kode Barang | kd\_barang | Varchar | 10 | Foreign Key |
| 3 | Nama Barang | nama\_barang | Varchar | 50 |  |
| 4 | Stok | Stok | Int | 5 |  |
| 5 | Qty | Qty | Int | 5 |  |
| 6 | Satuan | Satuan | Char | 8 |  |
| 7 | Harga Jual | harga\_jual | Int | 10 |  |
| 8 | Diskon | Diskon | Int | 10 |  |
| 9 | Subtotal | Subtotal | Int | 10 |  |
| 10 | Bayar | Bayar | Int | 10 |  |

1. Spesifikasi File Barang

Nama File : Barang

Akronim : barang

Fungsi : Untuk menyimpan data barang

Tipe File : File Transaksi

Organisasi File : Index Sequential

Akses File : Random

Media : Hard disk

Panjang Record : 80 byte

Kunci field : kd\_barang

Software : Xampp

Tabel 5. Spesifikasi File Pembeli

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Elemen Data | Nama Field | Tipe Data | Size | Keterangan |
| 1 | Kode Barang | kd\_barang | Varchar | 5 | Primary Key |
| 2 | Nama Barang | nama\_barang | Varchar | 50 |  |
| 3 | Stok | stok | Int | 5 |  |
| 4 | Satuan | satuan | Char | 10 |  |
| 5 | Harga Jual | harga\_jual | Int | 10 |  |

1. Spesifikasi File Retur

Nama File : Retur Penjualan

Akronim : Retur\_jual

Fungsi : Untuk menerangkan tentang transaksi pembelian

Tipe File : File Transaksi

Organisasi File : *Index Sequential*

Akses File : *Random*

Media : *Hard disk*

Panjang Record : 45 *byte*

Kunci field : -

Software : Xampp

Tabel 6. Spesifikasi File Pembeli

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Elemen Data | Nama Field | Tipe Data | Size | Keterangan |
| 1 | No Retur Penjualan | no\_retur\_jual | *Int* | 25 | *Primary Key* |
| 2 | Kode Pembeli | kd\_pemebeli | *Int* | 5 | *Foreign Key* |
| 3 | Tanggal Retur | tgl\_retur | *Date* |  |  |
| 4 | Kode User | kd\_user | *Varchar* | 15 | *Foreign Key* |

1. Spesifikasi File Detail Retur Penjualan

Nama File : Detail Retur Penjualan

Akronim : Detail\_retur\_jual

Fungsi : Untuk menerangkan detail tentang ada yang ada di retur penjualan

Tipe File : File Transaksi

Organisasi File : Index Sequential

Akses File : Random

Media : Hard disk

Panjang Record : 105 byte

Kunci field : -

Software : Xampp

Tabel 7. Spesifikasi File Pembeli

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Elemen Data | Nama Field | Tipe Data | Size | Keterangan |
| 1 | No Retur | no\_retur\_jual | *Int* | 15 | *Foreign Key* |
| 2 | Kode Barang | kd\_barang | *Varchar* | 5 | *Foreign Key* |
| 3 | Nama Barang | nama\_barang | *Varchar* | 30 |  |
| 4 | Jumlah Barang | jumlah\_barang | *Int* | 5 |  |
| 5 | Harga | harga\_beli | *Int* | 10 |  |
| 6 | Tanggal Garansi | tgl\_garansi | *Date* |  |  |
| 7 | Sebab Retur | sebab\_retur | *Char* | 40 |  |

1. Spesifikasi File Data Pegawai

Nama File : Data Pegawai

Akronim : pegawai

Fungsi : Untuk menyimpan data pelanggan

Tipe File : File Master

Organisasi File : *Index Sequential*

Akses File : *Random*

Media : *Hard disk*

Panjang Record : 155 *byte*

Kunci field : kd\_pegawai

Software : Xampp

Tabel 8. Spesifikasi File Pembeli

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Elemen Data | Nama Field | Tipe Data | Size | Keterangan |
| 1 | Kode Pegawai | kd\_pegawai | *Int* | 5 | *Primary Key* |
| 2 | Nama | Nama | *Varchar* | 18 |  |
| 3 | Tanggal Lahir | tgl\_lahir | *Date* |  |  |
| 4 | Jenis Kelamin | jenis\_kelamin | *Char* | 10 |  |
| 5 | Alamat | Alamat | *Varchar* | 100 |  |
| 6 | No Telepon | no\_telepon | *Int* | 12 |  |
| 7 | Bagian | Bagian | *Char* | 10 |  |

**SOURCE PROGRAM**

*===Database Inventory*

*== Table structure for table Inventory*

*|*

*|Column|Type|Null|Default*

*|*

*|//\*\*id\*\*//|int(11)|No|*

*|label|varchar(50)|No|*

*|Nama\_barang|int(11)|No|*

*== Dumping data for table course*

*== Table structure for table inventory*

*|*

*|Column|Type|Null|Default*

*|*

*|stock barang\_id|int(11)|No|*

*|Laporan penjualan\_id|int(11)|No|*

*|Sales\_order|float|No|*

*|description|varchar(150)|No|*

*== Dumping data for table sales order*

*== Table structure for table Laporan Stock*

*|*

*|Column|Type|Null|Default*

*|*

*|//\*\*id\*\*//|int(11)|No|*

*|Qty\_qty|varchar(50)|No|*

*|Harga\_Satuan|varchar(50)|No|*

*|Kode\_barang|char(1)|Harga\_Satuan|*

*|Total\_Harga|Retur\_penjualan||*

*Retur\_barang|text|No|*

*== Dumping data for table Inventory*

*== Table structure for table user*

*|*

*|Column|Type|Null|Default*

*|*

*|//\*\*id\*\*//|int(11)|No|*

*|username|varchar(50)|Nama|*

*|Jenis\_Kelamin|varchar(50)|User|Password*

*|Level created|date|No|*

*== Dumping data for table user*

**Source Code Control Servlet**

*package control;*

*import java.io.IOException; import java.io.PrintWriter;*

*import javax.servlet.ServletException; import javax.servlet.http.HttpServlet;*

*import javax.servlet.http.HttpServletRequest; import javax.servlet.http.HttpServletResponse;*

*import Persedian barang alat kesehatan.; //mengakses parameter import Transaksi;*

*import Barang; import Pembeli; import User; import Laporan; import Pegawai; import java.sql.\*;*

*import java.util.logging.Level; import java.util.logging.Logger;*

*import javax.servlet.annotation.WebServlet;*

*/\*\**

*\* @author michael*

*\*/*

*@WebServlet(name = " RANCANG BANGUN SISTEM INFORMASI PERSEDIAAN BARANG MENGGUNAKAN METODE QUICK MODELLING (METODE SECARA*

*CEPAT) PADA ADVANCE DI BEKASI Servlet", urlPatterns = {"/Laporan penjualan & Laporan stock barangTest1Servlet"})*

*public class pendaftaran Siswa Baru1Servlet extends HttpServlet {*

*/\*\**

* *Processes requests for both HTTP <code>GET</code> and*

*<code>POST</code>*

* *methods.*

*\**

* *@param request servlet request*
* *@param response servlet response*
* *@throws ServletException if a servlet-specific error occurs*
* *@throws IOException if an I/O error occurs*

*\*/*

*protected void processRequest(HttpServletRequest request, HttpServletResponse response)*

*throws ServletException, IOException, ClassNotFoundException, SQLException*

*{*

*response.setContentType("text/html;charset=UTF-8");*

*Sales Order= new Transaksi(); //--Inisialisasi Object—*

*String URL = "jdbc:mysql://localhost:3306/ ADVANCE DI BEKASI "; String USERNAME = "root";*

*String PASSWORD = "";*

*Connection koneksi = null; PreparedStatement pstmt = null; PreparedStatement pstmt2 = null; int result = 0;*

*try (PrintWriter out = response.getWriter()) {*

*/\* TODO output your page here. You may use following sample code. \*/ Class.forName("com.mysql.jdbc.Driver");*

*koneksi = DriverManager.getConnection(URL, USERNAME, PASSWORD);*

*String aksi = request.getParameter("aksi"); if (aksi != null) {*

*}*

*break; default:*

*break;*

*}*

*}*

*}*

*}*

*// <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the left to edit the code.">*

*/\*\**

* *Handles the HTTP <code>GET</code> method.*

*\**

* *@param request servlet request*
* *@param response servlet response*
* *@throws ServletException if a servlet-specific error occurs*
* *@throws IOException if an I/O error occurs*

*\*/ @Override*

*protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {*

*try {*

*processRequest(request, response);*

*} catch (ClassNotFoundException ex) { Logger.getLogger(Laporan Stock*

*Barang1Servlet.class.getName()).log(Level.SEVERE, null, ex);*

*} catch (SQLException ex) {*

*Logger.getLogger(pembelian1Servlet.class.getName()).log(Level.SEVERE, null,*

*ex);*

*}*

*}*

*/\*\**

* *Handles the HTTP <code>POST</code> method.*

*\**

* *@param request servlet request*
* *@param response servlet response*
* *@throws ServletException if a servlet-specific error occurs*
* *@throws IOException if an I/O error occurs*

*\*/ @Override*

*protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {*

*try {*

*processRequest(request, response);*

*} catch (ClassNotFoundException ex) { Logger.getLogger(Laporan*

*Penjualan1Servlet.class.getName()).log(Level.SEVERE, null, ex);*

*} catch (SQLException ex) {*

*Logger.getLogger(Stock barang1Servlet.class.getName()).log(Level.SEVERE, null, ex);*

*}*

*}*

*/\*\**

* *Returns a short description of the servlet.*

*\**

* *@return a String containing servlet description*

*\*/ @Override*

*public String getServletInfo() { return "Short description";*

*}// </editor-fold>*

*}*