

# **LAUNDRY POINT OF SALE EMBEDDED SYSTEM MENGGUNAKAN RASPBERRY PI**



**Oleh**

**ZULKIFLI SAID**

**2015130038**

**SISTEM KOMPUTER  
PROGRAM PASCASARJANA  
STMIK HANDAYANI  
MAKASSAR 2017**

# LAUNDRY POINT OF SALE EMBEDDED SYSTEM MENGGUNAKAN RASPBERRY

Oleh

**ZULKIFLI SAID**

**2015130038**

Telah dipertahankan di depan Dewan Penguji Tesis  
pada hari selasa 08 agustus 2017 dan  
dinyatakan telah memenuhi syarat

Menyetujui  
Komisi Penasehat

Dr. Ir. Syafruddin Syarif, M.T.

Ketua

Dr. Ir. Zahir Zainuddin, M.Sc

Sekretaris

Mengetahui,

Ketua Program Studi  
Magister Sistem Komputer

Dr. Ir. Zahir Zainuddin, M.Sc.

Direktur Program Pascasarjana  
STMIK Handayani Makassar

Dr. Eng. Armin Lawi, M.Eng.

## PERNYATAAN KEASLIAN TESIS

Dengan ini saya yang bertandatangan di bawah ini,

Nama : ZULKIFLI SAID  
NIM : 2015130038  
Judul Tesis : POINT OF SALE EMBEDDED SYSTEM MENGGUNAKAN RASPBERRY

Menyatakan bahwa Tesis dengan judul di atas beserta keseluruhan isi adalah benar-benar karya saya sendiri, dan saya tidak melakukan penjiplakan atau pengutipan dengan cara-cara yang tidak sesuai dengan etika keilmuan yang berlaku dalam masyarakat keilmuan. Atas pernyataan ini, saya siap menanggung resiko atau sanksi yang dijatuhkan kepada saya apabila kemudian ditemukan adanya pelanggaran terhadap etika keilmuan dalam karya saya ini, atau ada klaim dari pihak lain terhadap keaslian karya saya ini.

Makassar, ... September 2017

Yang Membuat Pernyataan,



Zulkifli Said  
NPM 2015130038

## **KATA PENGANTAR**

Segala puji bagi Allah Subhanahu wataallah atas segala limpahan Rahmat dan Hidayah-Nya, sehingga penulis dapat menyelesaikan penyusunan tesis ini, Salawat dan salam di haturkan kepada junjungan Nabi Muhammad Salallahu Alaihi Wasalam beserta keluarga, sahabat dan pegikutnya hingga akhir jaman.

Penulis dengan segala kerendahan hati mengucapkan terima kasih kepada :

1. Bapak Dr. Eng. Agussalim, M.T., selaku Ketua STMIK Handayani Makassar.
2. Bapak Dr. Eng. Armin Lawi, M.Eng., selaku Direktur Pascasarjana STMIK Handayani Makassar.
3. Bapak Dr. Ir. Zahir Zainuddin, M.Sc., selaku Ketua Program Studi Pascasarjana STMIK Handayani Makassar.
4. Bapak Dr. Ir. Syafruddin Syarif, M.T. dan Dr. Ir. Zulfajri Basri Hasanuddin, M.Eng. Masing-masing selaku Ketua Komisi Penasehat dan anggota komisi penasehat yang telah memberikan bimbingan, saran dan koreksi dalam penyusunan tesis ini.
5. Bapak Dr. Amil Ahmad Ilham, S.T, M.IT, Ph.D, Dr. Adnan, S.T, M.T, Ph.D dan Dr. Eng. Muhammad Niswar, S.T, M.IT, Ph.D. Masing-masing selaku penguji dalam penyusunan tesis ini.
6. Bapak dan Ibu Dosen Program Studi Sistem Komputer Pascasarjana STMIK Handayani Makassar yang telah memberikan ilmunya kepada penulis selama perkuliahan.

7. ibunda Fatimah serta kakak penulis Nurasiah Said dan Saaenab Said yang telah memberikan dukungan do'a dan materi selama penulis menempuh perkuliahan.
8. Keluarga besar penulis yang tidak dapat penulis sebutkan satu-persatu yang telah memberikan dukungan do'a dan materi selama penulis menempuh perkuliahan.
9. Rekan mahasiswa pascasarjana STMIK Handayani angkatan 2015 yang telah banyak memberikan bantuan dan dukungan ilmu dan materi pada penulis selama menempuh perkuliahan.
10. Semua pihak yang tidak dapat penulis sebutkan satu persatu dalam tesis ini atas dukungan moril dan materi hingga penyelesaian studi S2 ini.

Semoga Allah membalas semua kebaikan ini menjadi pahala amal zariyah berlipat ganda, akhir kata penulis menyampaikan permohonan maaf atas segala kekurangan dalam isi penulisan tesis ini, saran dan kritik sangat diharapkan untuk penyempurnaan isi tesis ini.

Makassar, September 2017

Penulis

Zulkifli Said

## ABSTRAK

**Zulkifli Said.** Laundry Point Of Sale Embedded System Menggunakan Raspberry Pi (dibimbing oleh Syafruddin Syarif dan Zulfajri Basri Hasanuddin).

Untuk menunjang fasilitas kegiatan usaha jasa binatu maka dimanfaatkan perangkat mini komputer raspberry pi sebagai perangkat pendukung yaitu alat timbangan digital dan perangkat lunak point of sale laundry yang memiliki user interfase manarik dan kemampuan penyimpanan data transaksi. Raspberry Pi memiliki port GPIO berfungsi sebagai *input-output* yang dapat dihubungkan dengan sensor *load cell* dengan bantuan modul Hx711 sebagai *Analog to Digital Converter*. Melalui input GPIO Raspberry Pi yang masih dalam bentuk data digital 24 bit kemudian diproses menggunakan bahasa pemrograman pascal yang akan ditampilkan pada layar LCD sebagai satuan berat. Variabel berat kemudian dijadikan sebagai faktor pengali dari harga paket pada usaha binatu yang menghasilkan nilai total harga transaksi, sehingga dapat membantu pegawai menghitung total harga cucian dalam bentuk rupiah yang kemudian disimpan ke dalam database mysql sebagai laporan pertanggung jawaban kepada pimpinan usaha binatu.

Hasil penelitian menunjukkan bahwa kemampuan timbangan digital maksimal 20kg, dengan tingkat keakurasian sama dengan timbangan digital konvensional dan hasil keluaran harga total yang benar berdasarkan sampel penelitian, kelurahan berat cucian  $3.45 \text{ Kg} \times 5,000=17,250$ ,  $2.18 \text{ Kg} \times 7,000=15,260$  dan  $5.27 \text{ Kg} \times 4,000=21,080$ .

**Kata Kunci :** Raspberry Pi, Load cell, timbangan digital, pascal, Hx711.

## ABSTRACT

**Zulkifli Said.** Laundry Point Of Sale Embedded System Using Raspberry Pi (guided by Syafruddin Syarif and Zulfajri Basri Hasanuddin).

To support the laundry service business activities, the use of mini computer raspberry pi device as a supporting tool that is digital scales and point of sale laundry software that has user interface userfriendly and transaction data storage capability. Raspberry Pi has GPIO port serves as input-output which can be connected with load cell sensor with the integrate of Hx711 module as Analog to Digital Converter. Through the Raspberry Pi GPIO input which is still in the form of 24 bit digital data then processed using pascal programming language that will be displayed on the LCD screen as the unit weight. The weight variable is then used as the multiplier factor of the package price on the laundry business resulting in the total value of the transaction price, which can help the employee to calculate the total laundry price in the form of rupiah which is then stored into the mysql database as accountability report to the head of the laundry business.

The results showed that the digital scalability capability was 20kg, with the same accuracy as the conventional digital scales and the correct total output results based on the research sample, the heavy districts of laundry  $3.45 \text{ Kg} \times 5,000 = 17.250$ ,  $2.18 \text{ Kg} \times 7,000 = 15.260$  and  $5.27 \text{ Kg} \times 4,000 = 21,080$ .

Keywords: Raspberry Pi, Load cell, digital scales, pascal, Hx711.

## DAFTAR ISI

|  |           |
|--|-----------|
| HALAMAN JUDUL .....                              | i         |
| HALAMAN PENGESAHAN .....                         | ii        |
| LEMBAR PERENYATAAN KEASLIAN TESIS .....          | iii       |
| KATA PENGANTAR .....                             | iv        |
| ABSTRAK .....                                    | v         |
| DAFTAR ISI .....                                 | viii      |
| DAFTAR GAMBAR .....                              | x         |
| DAFTAR TABEL .....                               | xi        |
| <b>BAB I PENDAHULUAN</b> .....                   | <b>1</b>  |
| 1.1 Latar Belakang .....                         | 1         |
| 1.2 Rumusan Masalah .....                        | 2         |
| 1.3 Tujuan Penelitian .....                      | 2         |
| 1.4 Batasan Masalah .....                        | 3         |
| 1.5 Manfaat Penelitian .....                     | 3         |
| <b>BAB II TINJAUAN PUSTAKA</b> .....             | <b>4</b>  |
| 2.1 Penelitian Terkait .....                     | 4         |
| 2.2 Teori Pendukung .....                        | 10        |
| 2.2.1 Timbangan Digital .....                    | 10        |
| 2.2.2 Sensor Load Cell .....                     | 11        |
| 2.2.3 Modul IC Hx711 .....                       | 12        |
| 2.2.4 Raspberry Pi .....                         | 14        |
| 2.2.5 Bisnis Laundry.....                        | 17        |
| 2.2.6 Point Of Sale .....                        | 18        |
| 2.2.7 Bahasa Pemrograman pada Raspberry Pi ..... | 19        |
| 2.2.8 Lazarus .....                              | 22        |
| <b>BAB III METODOLOGI PENELITIAN</b> .....       | <b>24</b> |
| 3.1 Metode Penelitian .....                      | 24        |

|   |           |
|---|-----------|
| 3.2 Analisis Data .....                                     | 24        |
| 3.3 Analisis Dokumen .....                                  | 24        |
| 3.4 Analisis Teknologi .....                                | 24        |
| 3.6 Analisis PIECES.....                                    | 25        |
| 3.6 Konfigurasi Sistem Secara Umum .....                    | 26        |
| 3.7 Tahapan Perancangan Pada Penelitian.....                | 29        |
| <b>BAB IV HASIL PENELITIAN DAN PEMBAHASAN.....</b>          | <b>31</b> |
| 4.1 Hasil Perancangan Alat.....                             | 31        |
| 4.2 Pengujian Sistem .....                                  | 33        |
| 4.2.1 Pengujian Performa Raspberry Pi saat akses GPIO ..... | 33        |
| 4.2.2 Pengujian Pembacaan perbandingan kalibrasi alat ..... | 34        |
| 4.3 Pengujian Blackbox .....                                | 35        |
| 4.3.1 Pengujian hasil perhitungan total harga cucian .....  | 35        |
| 4.3.2 Pengujian hasil penyimpanan database mysql .....      | 36        |
| 4.4 Implementasi Bahasa Pemrograman .....                   | 36        |
| <b>BAB V KESIMPULAN DAN SARAN .....</b>                     | <b>40</b> |
| 5.1 Kesimpulan .....  | 40        |
| 5.2 Saran .....   | 40        |
| <b>DAFTAR PUSTAKA.....</b>                                  | <b>42</b> |
| <b><u>LAMPIRAN-LAMPIRAN</u> .....</b>                       | <b>44</b> |

## DAFTAR GAMBAR

|            |  |    |
|------------|--|----|
| Gambar 2.1 | Diagram blok rancangan sistem .....                    | 6  |
| Gambar 2.2 | Diagram blok Timbangan digital dengan load cell .....  | 7  |
| Gambar 2.3 | Timbangan digital Komvensional .....                   | 11 |
| Gambar 2.4 | Load Cell .....  | 12 |
| Gambar 2.5 | Modul Hx711. ....                                      | 13 |
| Gambar 2.6 | PCB Board Modul Hx711. ....                            | 14 |
| Gambar 2.7 | Tampilan raspberry pi 3 tampak dari atas .....         | 15 |
| Gambar 2.8 | GPIO raspberry pi 3 Model B .....                      | 16 |
| Gambar 2.9 | Configurasi Lazarus IDE .....                          | 23 |
| Gambar 3.1 | Blok Diagram Sistem.....                               | 27 |
| Gambar 3.2 | Rangkaian GPIO raspberry pi 3 dengan Modul Hx711 ..... | 27 |
| Gambar 3.3 | Rangkaian Load Cell dengan Modul Hx711 .....           | 28 |
| Gambar 3.4 | Teknik pesangan load cell .....                        | 29 |
| Gambar 3.5 | Flowchart Sistem .....                                 | 25 |
| Gambar 4.1 | Hasil Perancangan Alat .....                           | 31 |
| Gambar 4.2 | Tampilan Halaman depan laundry point of sale .....     | 32 |
| Gambar 4.3 | Pengukuran berat pada timbangan .....                  | 32 |
| Gambar 4.4 | Pengujian respon GPIO Raspberry Pi.....                | 34 |
| Gambar 4.5 | Hasil penyimpanan data harga cucian .....              | 29 |

## DAFTAR TABEL

|   |    |
|---|----|
| Tabel 2.1 Spesifikasi Load Cell.....                      | 12 |
| Tabel 2.2 Spesifikasi Raspberry Model A dan Model B ..... | 17 |
| Tabel 3.1 Deskripsi Metode analisis PIECES .....          | 25 |
| Tabel 4.1 Berat Cucian Yang Diuji .....                   | 34 |
| Tabel 4.2 Hasil Validasi tingkat keakurasian.....         | 35 |

## **BAB I**

### **PENDAHULUAN**

#### **1.1 Latar Belakang**

Pertumbuhan bisnis franchise laundry di Indonesia saat ini menunjukkan perkembangan yang cukup signifikan. Tidak hanya di kota-kota besar saja, di pelosok daerah pun kini bisnis binatu tersebut mulai menjamur dengan berbagai tawaran fasilitas dan pelayanan. Maraknya bisnis laundry juga seiring dengan semakin tingginya tingkat kepercayaan masyarakat akan jasa laundry. Di samping itu, perubahan gaya hidup masyarakat yang lebih memilih mendatangi laundry dibandingkan mencuci sendiri membuktikan bahwa bisnis ini memiliki prospek yang menjanjikan.

Ketua Asosiasi Profesi Laundry Indonesia (APLI), Wasono Raharjo mengatakan jika ingin membuka usaha rumah cucian dengan modal sedikit, laundry kiloan adalah pilihan yang tepat. Para pebisnis dapat membangun usaha jasa tersebut di lokasi-lokasi strategis seperti perumahan penduduk, kampus atau kost-kostan. Dengan besarnya minat konsumen, membuat pelaku bisnis laundry menemui kendala-kendala dalam menjalankan usaha tersebut seperti yang sering muncul adalah proses transaksi yang bersifat konvensional, analisis keuangan dan neraca laba rugi untuk kemajuan usaha tersebut.

Untuk menjawab kendala tersebut para pelaku bisnis laundry mulai menerapkan perangkat komputer konvensional dan perangkat lunak sebagai fasilitas pendukung dalam menjalankan transaksi usaha jasa laundry. Dalam prakteknya implementasi komputer konvensional dan dukungan perangkat lunak terhadap usaha jasa laundry di Indonesia banyak menghabiskan sumber daya yaitu penggunaan energi listrik dengan sia-sia disebabkan besarnya spesifikasi perangkat komputer yg digunakan, Hal tersebut terjadi karena belum adanya dukungan pengembang perangkat komputer atau

pengembang perangkat lunak untuk menangani secara khusus proses transaksi bisnis laundry agar lebih efisien. Digital yang dibuat lebih praktis karena dapat merekam dan menyimpan data langsung ke komputer tanpa harus melakukan pencatatan secara manual ( Kamirul, dkk, 2015 ). Perlu kita ketahui juga Penggunaan alat pengukuran digital dalam kehidupan sehari-hari sangat membantu memudahkan kinerja manusia (Afdali, dkk, 2017).

Untuk mendorong penggunaan komputer lebih efisien, tepat guna dan membantu pelaku bisnis laundry dalam proses transaksi dan mengontrol jalannya usaha maka penulis mencoba merancang sebuah sistem laundry *point of sale* dengan embedded sistem menggunakan raspberry pi.

## 1.2 Rumusan Masalah

Berikut rumusan masalah pada penelitian ini adalah “Bagaimana perangkat mini komputer raspberry pi dimanfaatkan sebagai teknologi pendukung dalam bidang bisnis jasa laundry di Indonesia”.

Permasalahan yang terjadi pada objek penelitian, antara lain adalah sebagai berikut:

- a. Dalam pencatatan data transaksi pelanggan konvensional, sehingga pelayanan kepada pelanggan lambat.
- b. Pencarian data pelanggan jasa laundry kurang cepat, tepat dan akurat.
- c. Kesulitan dalam menyusun laporan, pendataan yang disimpan belum maksimal dan optimal dalam lembar-lembar catatan yang berbentuk arsip walaupun tetap menggunakan komputer dengan alat bantu.

## 1.3 Tujuan Penelitian

Tujuan penelitian ini adalah merancang sistem laundry pos embedded sistem menggunakan raspberry pi yang akan didesain sebagai

perangkat pendukung dalam bidang bisnis jasa laundry yang efektif dan efisien.

#### **1.4 Batasan Masalah**

Batasan masalah pada penelitian ini adalah:

1. Mengoptimalkan perangkat sistem yang dibangun menggunakan Raspberry Pi
2. Mengoptimalkan jangkauan dan akurasi timbang digital maksimal 20 kg.
3. Merancang interface point of sale yang terintegrasi dengan timbangan digital berbasis raspberry pi

#### **1.5 Manfaat Penelitian**

Adapun manfaat yang ingin dicapai jika penelitian ini terselesaikan yaitu:

1. Berfungsi sebagai perangkat standar dalam mengelola proses bisnis jasa laundry
2. Dapat meningkatkan pelayanan dibidang usaha jasa laundry
3. Menambah referensi bagi peneliti yang ingin meneliti dibidang komputer embedded system menggunakan raspberry pi.
4. Dapat menegetahui sejauh mana kemajuan teknologi mini computer raspberry pi dimanfaatkan dalam bidang bisnis.

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Penelitian Terkait**

Sebagai bahan pertimbangan dalam penelitian ini akan dicantumkan beberapa hasil penelitian terdahulu, antara lain:

1. R. Arif Tri Rahmawanto, (2014) : dengan penelitian Pengembangan Timbangan Buah Digital Berbasis Mikrokontroler Atmega16, Sistem yang digunakan terdiri atas perangkat keras (hardware) dan perangkat lunak (software). Perangkat keras yang dikembangkan memiliki susunan. Sistem tersusun atas load cell, penguat instrumentasi, penapis, pengubah analog ke digital 12-bit, mikrokontroler dan LCD. Sistem mikrokontroler juga dilengkapi dengan tombol-tombol sebagai media perantara memasukkan harga dasar untuk buah yang ditimbang. Mikrokontroler yang digunakan adalah ATmega16 sebagai pusat pengendali dan Liquid Crystal Display (LCD) 16 x 2 sebagai penampil.

Load cell digunakan sebagai sensor untuk mendeteksi berat dari beban yang diberikan. Kapasitas maksimal load cell yang digunakan adalah 5 kg. Sedangkan rangkaian penguat instrumentasi digunakan untuk menguatkan tegangan dari keluaran sensor karena tegangan keluaran sensor yang sangat kecil. Jenis penguat instrumentasi yang digunakan adalah seri AMP04.

Penapis lolos bawah (LPF) jenis Bessel orde-5 digunakan untuk meloloskan isyarat frekuensi rendah yang merupakan informasi tegangan hasil pengukuran beban. Jenis yang digunakan adalah MAX7419. Pengubah analog ke digital (ADC) digunakan untuk mengubah isyarat tegangan analog keluaran penguat instrumentasi menjadi data digital. Lebar data digitalnya adalah 12-bit dengan

pengubah seri ADS7822. Data digital hasil pengubahan tersebut selanjutnya akan diolah oleh mikrokontroler ATmega16.

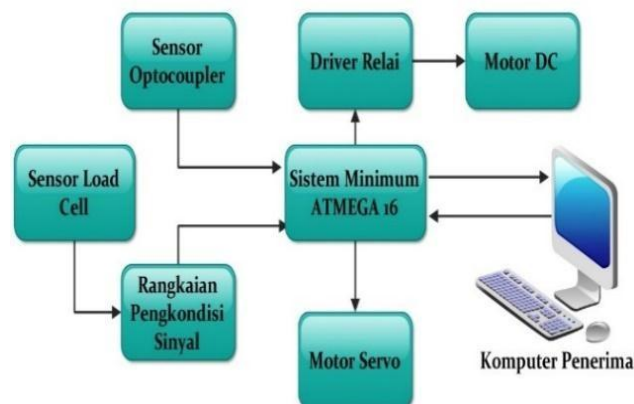
Perangkat lunak yang dikembangkan terdiri dari sebuah program untuk mengolah data penimbangan serta untuk kalkulasi harga yang ditetapkan. Komunikasi yang digunakan untuk menghubungkan ADC eksternal dengan mikrokontroler adalah komunikasi serial, SPI. Penapisan perangkat lunak data digital terkonversi dilakukan dengan menggunakan rata-rata bergerak (moving average, MA) dengan jumlah panjang data 10.

Harga dimasukkan menggunakan tombol. Ada empat tombol yang digunakan. Tombol pertama untuk menentukan nilai ratusan, tombol kedua untuk menetapkan nilai ribuan kemudian tombol ketiga untuk menetapkan nilai puluhan ribu. Tombol keempat adalah tombol untuk memulai kalkulasi, yaitu mengalikan harga yang sudah dimasukkan dengan berat. Program ini juga mengatur penaksiran harga total berdasarkan satuan mata uang rupiah terkecil yang ditetapkan, yaitu Rp 100.

2. Mirfan,( Agustus 2016) : melakukan penelitian tentang Mesin Penyaji Beras Secara Digital. Timbangan beras dirancang dengan menggunakan aluminium sebagai rangka dengan tinggi 40 cm dan lebar 40 cm dan akrilik yang memiliki dimensi yang tidak terlalu besar dan ringan sebagai papan untuk meletakkan komponen-komponen .

Pemilihan bahan ini didasarkan pada struktur yang kuat dan ringan. Adapun komponen-komponen seperti komponen push button, tombol on/of, keypad dan lcd disimpan pada bagian atas akrilik, power supply, mikrokontroler Arduino Mega 2560 motor servo dan sensor berat (load cell) ditempatkan pada sisi bawah rangka akrilik.

3. Arief Cipta Indra Rukmana, (April 2014) : melakukan penelitian tentang Aplikasi Sensor Load Cell pada Purwarupa Sistem Sortir Barang, Perancangan sistem pada penelitian ini terdiri dari perancangan perangkat keras (hardware) dan perangkat lunak (software). Perangkat keras dirancang terdiri dari belt conveyer, sistem pendeteksi benda, sisten timbang dan aktuator. Sistem timbang dan pendeteksi benda terdiri dari rangkaian sensor load cell dengan pengkondisi sinyal, sensor optocoupler yang terdiri dari bagian pengirim dan penerima serta rangkaian sistem minimum ATmega16 yang dilengkapi dengan LCD. Sedangkan aktuator terdiri dari motor servo dan driver relay untuk mengendalikan motor DC. Pada perancangan perangkat lunak terdiri dari perancangan aplikasi tertanam mikrokontroler menggunakan program BASCOM AVR dan perancangan antarmuka menggunakan program Microsoft Visual Basic 6.0. Diagram blok sistem secara keseluruhan dapat dilihat pada Gambar 3.



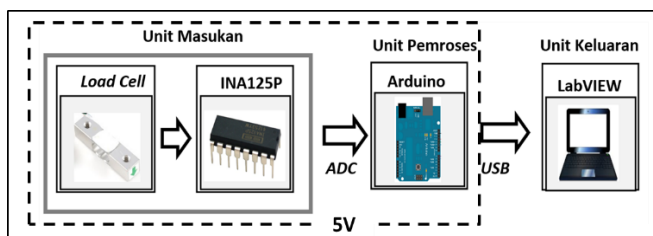
Gambar 2.1 Diagram blok rancangan sistem

Barang yang dibawa oleh belt conveyer akan lewat di antara bagian pengirim dan penerima sensor optocoupler menyebabkan cahaya infra merah dari bagian pengirim tidak sampai ke

phototransistor pada bagian penerima sehingga bagian penerima berlogika low sehingga motor penggerak conveyor berhenti dan barang yang dibawa tepat berada di atas sensor load cell untuk ditimbang beratnya. hasil timbang lalu dibandingkan dengan kisaran berat batas yang ditentukan pengguna dengan toleransi sebesar 10%. Apabila tidak sesuai, maka barang akan didorong oleh motor servo sehingga terpisah dari barang lain yang beratnya sesuai dengan keinginan pengguna.

4. Kamirul,(Desember 2015) : melaukan penelitian Rancang Bangun Data Logger Massa Menggunakan Load Cell, Sebagai unit pemroses digunakan mikrokontroller Arduino Uno R3 yang memiliki fasilitas ADC (AnalogDigital Converter). Selain itu Arduino Uno juga memiliki keluaran 5 VDC yang dapat digunakan untuk menyuplai tegangan ke load cell dan penguat INA125P dengan stabil. Untuk menampilkan hasil pembacaan, digunakan perangkat lunak LabVIEW (Laboratory Virtual Instrumentation Engineering Workbench). Karena aplikasinya yang luas dan kemampuannya dalam menyediakan paket khusus koneksi LabVIEW Interface For arduino (LIFA), maka LabVIEW merupakan salah satu perangkat lunak yang sesuai untuk menampilkan hasil pembacaan timbangan berbasis Arduino.

Dalam proses perancangannya, timbangan digital dibuat berdasarkan diagram blok dibawah ini.



Gambar 2.2 Diagram blok timbangan digital dengan load cell

Pengujian dilakukan dengan bervariasi nilai hambatan  $R_g$  pada rangkaian penguat. Variasi nilai  $R_g$  akan menghasilkan nilai penguatan yang bervariasi pula sesuai dengan persamaan (4). Selain itu, dengan bervariasi  $R_g$ , spesifikasi timbangan (NST dan batas ukur) juga akan mengalami pergeseran.

Pada penelitian ini, digunakan  $R_g = 10 \Omega$  sebagai hambatan tetap penguat INA125P. Nilai tersebut dipilih karena pada nilai  $R_g$  tersebut dihasilkan jangkauan pengukuran yang tinggi dan nilai sensitifitas yang tinggi pula. Hal ini dianggap sesuai untuk diimplementasikan pada jasa pengiriman barang berupa dokumen.

Nilai pembacaan yang dihasilkan oleh timbangan digital sudah sangat baik dan hampir mendekati nilai sesungguhnya (massa uji) dengan nilai kesalahan relatif rata-rata sebesar 0,46 %. Ini menunjukkan bahwa timbangan digital yang dibuat sudah mampu memberikan nilai bacaan yang valid dengan nilai kesalahan yang kecil. Oleh karena itu, timbangan digital berbasis load cell ini sudah layak untuk diimplementasikan untuk menimbang massa beban yang dengan jangkauan maksimal 330 gram.

Pada penelitian ini, timbangan digital diimplementasikan untuk jasa pengiriman barang berupa dokumen. Seperti yang kita ketahui bahwa meskipun sudah menggunakan timbangan digital, namun proses pencatatan masih dilakukan manual. Oleh karena itu, implementasi timbangan digital pada penelitian ini difungsikan untuk dapat mencatat data secara otomatis. Hal ini dapat mengurangi kesalahan pembacaan yang diakibatkan oleh pengguna (human error).

5. Try Utami Hidayani,(2012) melakukan penelitian rancang bangun timbangan buah digital dengan keluaran berat dan harga. Alat ini menggunakan sensor Load cell untuk mendeteksi berat benda yang akan ditimbang. Prinsip kerja alat ini adalah jika ada penambahan

berat, maka sensor akan mulai mendeteksi berat benda. kemudian keluaran dari sensor yang berupa perubahan resistansi ini akan diubah menjadi perubahan tegangan oleh rangkaian mikrokontroler. Rangkaian ADC yang terdapat pada mikrokontroler ini akan mengubah sinyal analog menjadi sinyal digital. Kemudian hasil konversi dari rangkaian ADC akan diproses oleh rangkaian mikrokontroler untuk ditampilkan ke display LCD sebagai data berat dalam satuan kilogram (Kg). Dari hasil penelitian dan pembahasan tentang alat timbangan digital dengan keluaran harga barang dapat disimpulkan bahwa:

- a. Piranti elektronik yang dibutuhkan dalam rangkaian timbangan digital ini adalah Mikrokontroler Atmega32, Sensor load cell, LCD, dan keypad. Alat ini dibuat dengan merangkai piranti-piranti elektronik yang menjadi suatu sistem yang dapat mendeteksi berat buah dan akan menampilkannya secara otomatis.
  - b. Secara keseluruhan alat yang dibuat, dapat bekerja dan berfungsi sebagaimana yang diharapkan, sehingga diharapkan dapat dimanfaatkan sebagai alat timbang digital yang mampu menampilkan berat secara otomatis. Mikrokontroler Atmega32 sebagai pengendali utama, cukup efisien karena membutuhkan perangkat keras yang sedikit serta kebutuhan sumber catu daya yang kecil.
  - c. Kemampuan padatimbangan ini berdasarkan pembebanan pada sensor load cell hingga pada berat beban maksimal yang dimiliki oleh sensor yaitu hanya 5 Kg.
6. Penelitian yang akan dilakukan oleh penulis berbeda dengan penelitian sebelumnya. Penulis akan membuat seperangkat alat dan software point of sale embedded system menggunakan RaspberryPi yang akan mengintegrasikan teknologi timbangan digital kedalam system point of sale yang dirancang khusus untuk menerima inputan

berat pakaian kemudian diproses untuk dihitung jumlah transaksi kemudian menyimpan data transaksi untuk kebutuhan laporan dan evaluasi usaha jasa laundry. Kelebihan dari system ini adalah efisiensi energy dan waktu dalam menyelesaikan transaksi jasa laundry.

Perbedaan mendasar dari system yang dirancang oleh penulis dengan yang telah ada adalah:

- a. System lama hasil berat timbangan digital yang terpisah dengan computer atau dokumen kemudian akan diketik manual kedalam computer atau dokumen
- b. Sedangkan system yang dibuat oleh penulis hasil timbangan sudah terintegrasi kedalam system jadi tidak diperlukan lagi ketik manual.

## **2.2 Teori Pendukung**

### **2.2.1 Timbangan digital**

Timbangan adalah suatu alat yang bisa dipakai untuk melakukan uji coba pengukuran berat dari suatu benda. timbangan bisa di artikan sebagai sebuah alat yang bisa dipakai untuk melakukan pengukuran berat dari suatu benda. Terdapat dua sistem kategori dalam timbangan yaitu timbangan dengan sistem mekanik/analog dan sistem elektronik/digital (Khakim, 2015). Timbangan manual, yaitu jenis timbangan biasa yang bekerja secara manual melalui perantara manusia yang sering digunakan dalam kehidupan sehari-hari. sedangkan Timbangan digital, yaitu jenis timbangan yang dapat bekerja secara elektronik dan otomatis dengan input arus listrik dan indikatornya berupa angka digital yang ditunjukkan pada layar LCD. Fungsi dari kedua jenis timbangan ini adalah sama, yaitu untuk menghitung berat suatu benda, masing-masing dari kedua jenis timbangan ini memiliki karakteristik dan tingkat akurasi pengukuran yang berbeda (Erlangga, 2011). Berikut ini merupakan Persamaan matematis suatu neraca pegas pada timbangan manual dinyatakan dalam:

$$k \cdot X = m \cdot g \quad (1)$$

Dimana  $k$  adalah konstanta pegas,  $X$  adalah defleksi,  $m$  adalah massa, dan  $g$  adalah gravitasi.

Timbangan digital tersedia berbagai model, merek, ukuran, dan model yang berbeda, dan biasanya datang dengan baterai dan bobot kalibrasi, bantalan timbangan, serta nampan. Timbangan digital setiap tipe produk yang berbeda, bermacam-macam dalam harga dan kualitas.



Gambar 2.3 Timbangan digital Komvensional

### 2.2.2 Sensor Load Cell

*Load Cell* adalah komponen utama pada sistem timbangan digital. Sensor *load cell* apabila diberi beban pada inti besi maka nilai resistansi di *strain gauganya* akan berubah yang dikeluarkan melalui empat kabel (Syaiful Kasim, 2013). Dua kabel sebagai eksitasi dan dua kabel lainnya sebagai sinyal keluaran ke control.

Secara umum load cell digunakan untuk menghitung massa dari suatu benda. Sebuah sensor load cell tersusun dari beberapa konduktor, strain gauge, dan jembatan wheatstone (Nuryanto, 2015).



Gambar 2.4 Load Cell

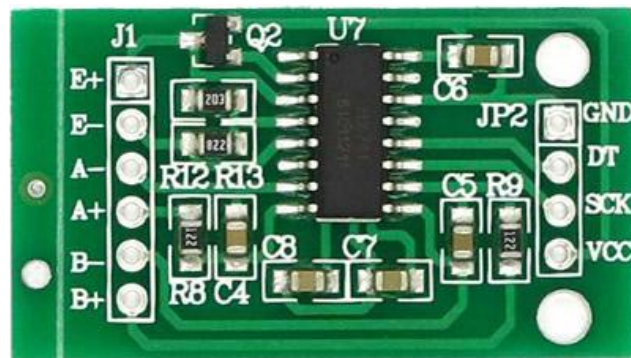
Sensor Load cell memiliki spesifikasi kerja sebagai berikut :

Tabel 2.1 Spesifikasi Sensor load cell .

|                        |                          |
|------------------------|--------------------------|
| Brand Name             | ZHIPU                    |
| Weight Scale Type      | LAB-40Kg                 |
| Series                 | B-B-B-2.0                |
| Material Type          | Metal                    |
| Specifics Type         | Vibration Sensor         |
| Specifics Output       | Analog Sensor            |
| Rated Capacity         | 40kg                     |
| Accuracy Class         | B                        |
| Rated Output           | 2.0±10% $mV/V$           |
| Zero Balance           | ±5% $R.O$                |
| Input Resistance       | 405±3 $\Omega$           |
| Output Resistance      | 350±3 $\Omega$           |
| Linearty Error         | ±0.02% $R.O$             |
| Repeatability Error    | ±0.015% $R.O$            |
| Hysteresis Error       | ±0.015% $R.O$            |
| Creep in 30 Min        | ±0.03% $R.O$             |
| Temp.Effect on Output  | ±0.03% $R.O/10^{\circ}C$ |
| Temp.Effect on Zero    | ±0.05% $R.O/10^{\circ}C$ |
| Compesanted Temp.Range | -10-+40 $^{\circ}C$      |
| Excitation, Recommeded | 5-12VDC                  |
| Excitation, Maximum    | 18VDC                    |
| Operation Temp,Range   | -20-+60 $^{\circ}C$      |
| Safe Overload          | 150% $R.C$               |
| Ultimate               | 200% $R.C$               |
| Insulation Resistance  | ≥2000M $\Omega(50VDC)$   |
| Cable Length           | ∅4mmX0.4m*               |
| Protection Class       | IP65                     |
| Size                   | 13mm X 25mm X 22mm       |

### 2.2.3 Modul IC Hx711

Hx711 adalah modul timbangan yang memiliki prinsip kerja mengkonversi bilangan yang terukur dalam perubahan resistansi dan mengkonversinya kedalam besaran tegangan melalui rangkaian yang ada. Modul melakukan komunikasi dengan komputer/mikrokontroler melalui TTL232. Kelebihan daripada modul Hx711 adalah struktur yang sederhana, mudah dalam penggunaan, hasil yang stabil dan reliable, memiliki sensitifitas yang tinggi dan mampu mengukur perubahan dengan cepat.



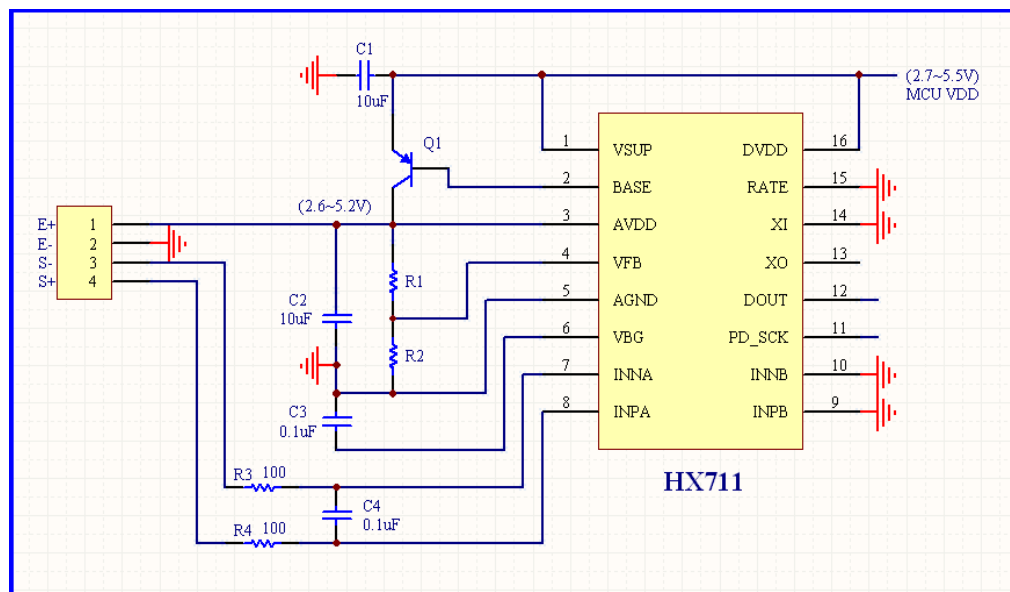
Gambar 2.5 Modul Hx711

HX711 biasanya digunakan pada bidang aerospace, mekanik, elektrik, kimia, konstruksi, farmasi dan lainnya, digunakan untuk mengukur gaya, gaya tekanan, perpindahan, gaya tarikan, torsi, dan percepatan. Spesifikasinya adalah sebagai dibawah berikut :

1. Differential input voltage:  $\pm 40\text{mV}$
2. Data accuracy: 24 bit (24 bit A / D converter chip.)
3. Refresh frequency: 80 Hz
4. Operating Voltage : 5V DC
5. Operating current : :  $< 10\text{ mA}$
6. Size:  $38\text{mm} \times 21\text{mm} \times 10\text{mm}$

7. Red wire to E+
8. Black wire to E-
9. Green wire to A-
10. White wire to A+

Modul HX711 memiliki presisi tinggi 24 ADC high gain input yang didesain untuk berbagai sensor berjenis Bridge. Dengan dua channel A dan B (fix gain 32) yang berkomunikasi secara multiplex, modul ini dapat di program untuk gain 128 atau 64 (20mV atau 40mV). Prinsip kerja dari modul HX711 ini yaitu sebagai penguat tegangan pada load cell pada saat load cell bekerja. HX711 presisi 24-bit analog to digital converter (ADC) (Khakim, 2015). Berikut gambar PCB modul Hx711.

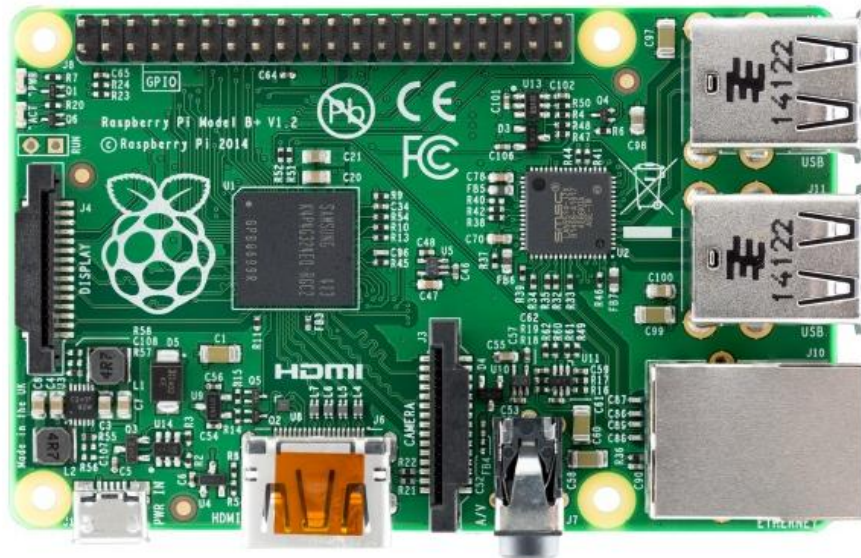


Gambar. 2.6 PCB Board Modul Hx711 (Single Layer)

## 2.2.4 Raspberry Pi

Raspberry Pi adalah komputer mikro berukuran seperti kartu kredit yang dikembangkan oleh Raspberry Pi Foundation, Inggris. Komputer single board ini dikembangkan dengan tujuan untuk mengajarkan dasardasar ilmu komputer dan pemrograman untuk siswa sekolah di seluruh dunia.

Meskipun mikrokontroler yang memiliki fisik seperti Arduino dimana lebih dikenal untuk proyek-proyek prototyping, tidak demikian dengan Raspberry Pi yang sangat berbeda dari mikrokontroler kebanyakan, dan sebenarnya, lebih seperti komputer daripada Arduino.



Gambar 2.7 Tampilan raspberry pi 3 tampak dari atas

Raspberry Pi terdiri dari banyak bagian perangkat keras yang penting dengan beberapa fungsi yang penting. Bagian utama dari Raspberry Pi adalah *processor* nya. Setiap Raspberry Pi memiliki *BCM2835 Chip Broadcom* yang mewujudkan suatu CPU inti

ARM1176JZF-S. Chip ini memiliki *clock speed* 700MHz dan merupakan sistem 32-bit (Baskoro, 2014).

Raspberry Pi merupakan sebuah komputer berukuran mini sebesar kartu kredit dengan harga relative murah. Raspberry memiliki dua model yaitu Raspberry model A dan model B. Perbedaan diantara keduanya terletak pada keberadaan Ethernet yang absen pada model A dan jumlah port USB yang menjadi dua kalinya pada model B (Baskoro, 2014).

| Raspberry Pi 3 GPIO Header |                                    |  |                                    |      |
|----------------------------|------------------------------------|--|------------------------------------|------|
| Pin#                       | NAME                               |  | NAME                               | Pin# |
| 01                         | 3.3v DC Power                      |  | DC Power 5v                        | 02   |
| 03                         | GPIO02 (SDA1 , I <sup>2</sup> C)   |  | DC Power 5v                        | 04   |
| 05                         | GPIO03 (SCL1 , I <sup>2</sup> C)   |  | Ground                             | 06   |
| 07                         | GPIO04 (GPIO_GCLK)                 |  | (TXD0) GPIO14                      | 08   |
| 09                         | Ground                             |  | (RXD0) GPIO15                      | 10   |
| 11                         | GPIO17 (GPIO_GEN0)                 |  | (GPIO_GEN1) GPIO18                 | 12   |
| 13                         | GPIO27 (GPIO_GEN2)                 |  | Ground                             | 14   |
| 15                         | GPIO22 (GPIO_GEN3)                 |  | (GPIO_GEN4) GPIO23                 | 16   |
| 17                         | 3.3v DC Power                      |  | (GPIO_GEN5) GPIO24                 | 18   |
| 19                         | GPIO10 (SPI_MOSI)                  |  | Ground                             | 20   |
| 21                         | GPIO09 (SPI_MISO)                  |  | (GPIO_GEN6) GPIO25                 | 22   |
| 23                         | GPIO11 (SPI_CLK)                   |  | (SPI_CE0_N) GPIO08                 | 24   |
| 25                         | Ground                             |  | (SPI_CE1_N) GPIO07                 | 26   |
| 27                         | ID_SD (I <sup>2</sup> C ID EEPROM) |  | (I <sup>2</sup> C ID EEPROM) ID_SC | 28   |
| 29                         | GPIO05                             |  | Ground                             | 30   |
| 31                         | GPIO06                             |  | GPIO12                             | 32   |
| 33                         | GPIO13                             |  | Ground                             | 34   |
| 35                         | GPIO19                             |  | GPIO16                             | 36   |
| 37                         | GPIO26                             |  | GPIO20                             | 38   |
| 39                         | Ground                             |  | GPIO21                             | 40   |

Rev. 2  
29/02/2016

www.element14.com/RaspberryPi

Gambar 2.8 GPIO raspberry pi 3 Model B

Spesifikasi Raspberry tipe A dan tipe B.

Tabel 2.2 Spesifikasi Raspberry Model A dan Model B.

| Fitur Teknis          | Model A  | Model B       |
|-----------------------|--|---------------|
| SoC (System on Chip)  | Broadcom BCM2835                               |               |
| CPU                   | 700 MHz Low Power ARM1176JZ-F                  |               |
| GPU                   | Dual core VideoCore IV multimedia Co-processor |               |
| Memory                | 256MB SDRAM                                    | 512MB SDRAM   |
| USB2.0                | 1  | 2             |
| Video Out             | Composite RCA (PAL and NTSC), HDM              |               |
| Audio Out             | 3.5mm jack, HDMI                               |               |
| Storage               | SD/MMC/SDIO card slot                          |               |
| Network               | No port  | RJ45 Ethernet |
| Peripheral Connectors | 8xGPIO, UART, I2C bus, SPI bus                 |               |
| Power Source          | 8xGPIO, UART, I2C bus, SPI bus                 |               |

### 2.2.5 Bisnis Laundry

Bisnis laundry merupakan usaha penyedia jasa layanan dalam menangani masalah pakaian kotor. Layanan binatu telah menjadi gaya hidup modern dari masyarakat kota. Hal ini menimbulkan usaha atau gerai binatu terus tumbuh setiap tahunnya. Umumnya proses layanan laundry di Indonesia memiliki prosedur yang sama. Pertama, pakaian kotor didaftarkan pelanggan ke layanan laundry. Pihak laundry membersihkan pakaian yang terdiri dari cuci, jemur, setrika, pengemasan. Terakhir, pelanggan mengambil pakaian dan membayar layanan laundry.

Bisnis jasa Laundry merupakan bisnis yang memberikan jasa kepada pihak konsumen atau kepada organisasi bisnis lain. Prospek layanan jasa pencucian laundry ini cukup luas apabila dengan dukungan kemampuan

manajemen mengidentifikasi dan antisipasi kebutuhan konsumen. Bisnis laundry ini akan terus berkembang karena mencuci merupakan kebutuhan pokok, selama semua orang masih menggunakan baju atau pakaian lainnya, bisnis laundry masih akan tetap hidup.

Target pasar mulai dari mahasiswa, kost, rumah tangga ,industri, perhotelan, rumah makan, perkantoran dan segala bisnis yang berkaitan dengan konveksi. Pasar ini akan terus berkembang seiring berkembangnya sebuah penduduk atau sebuah kota.

Tahapan-tahapan penerimaan order dapat dirinci sebagai berikut :

1. Mencatatat nama dan nomor telepon pelanggan.
2. Penerimaan Pakaian yang akan di Laundry dari Konsumen kepada Pihak Laundry
3. Menimbang dan menghitung jumlah pakaian.pakaian yang akan di Laundry.
4. Kemudian menghitung jumlah berat pakaian dikalikan dengan tariff laundry kiloan yang telah ditentukan.
5. Membuatkan Nota sebagai tanda terima pakaian.
6. Mencuci pakaian dilanjutkan dengan pengeringan pakaian
7. Menyimpan pakaian konsumen tersebut ke dalam 1 keranjang cuci tersendiri yang ditandai dengan copy bon atau nomor Nota.
8. Dilanjutkan pengambilan pakaian umumnya dilakukan dalam jangkan waktu 1,2 atau 3 hari pemilik pakaian akan datang mengambil cucian atau diantarkan.

### **2.2.6 Point Of Sale**

*Point of Sale* atau yang biasa disingkat POS yaitu, merupakan kegiatan yang berorientasi pada transaksi penjualan serta sistem yang membantu proses perhitungan transaksi. Setiap POS terdiri dari hardware

dan software dimana kedua komponen tersebut digunakan untuk setiap proses transaksi untuk memudahkan dalam perhitungan jumlah transaksi dan memudahkan pemilik usaha dalam mengevaluasi laba-rugi transaksi.

POS software dan hardware merupakan satu kesatuan komponen utama dari sistem POS. Hardware POS dibutuhkan untuk menjalankan fungsinya, membantu proses perhitungan dan membuat tanda terima transaksi penjualan untuk pelanggan. Sedangkan POS software merupakan interface yang menampilkan informasi hasil proses perhitungan transaksi. Dalam hal pemilihan hardware dan software menyesuaikan dengan lingkungan kerja, seperti yang akan digunakan oleh penulis pada usaha jasa laundry dalam penelitian ini adalah Raspberry Pi dan Timbangan Digital yang terintegrasi dengan POS software, yang merupakan bagian terpenting untuk mempercepat proses perhitungan berat pakaian dan proses pelayanan transaksi pembayaran pada usaha jasa laundry.

## **2.2.7 Bahasa Pemrograman pada Raspberry Pi**

### **a. Bahasa pemrograman Python**

Python adalah bahasa pemrograman interpretatif multiguna dengan filosofi perancangan yang berfokus pada tingkat keterbacaan kode (Noprianto, 2002). Python diklaim sebagai bahasa yang menggabungkan kapabilitas, kemampuan, dengan sintaksis kode yang sangat jelas, dan dilengkapi dengan fungsionalitas pustaka standar yang besar serta komprehensif.

Python mendukung multi paradigma pemrograman, utamanya; namun tidak dibatasi; pada [pemrograman berorientasi objek](#), pemrograman imperatif, dan pemrograman fungsional. Salah satu fitur yang tersedia pada python adalah sebagai [bahasa pemrograman dinamis](#) yang dilengkapi dengan manajemen memori otomatis. Seperti

halnya pada bahasa pemrograman dinamis lainnya, python umumnya digunakan sebagai [bahasa skrip](#) meski pada praktiknya penggunaan bahasa ini lebih luas mencakup konteks pemanfaatan yang umumnya tidak dilakukan dengan menggunakan bahasa skrip. Python dapat digunakan untuk berbagai keperluan pengembangan perangkat lunak dan dapat berjalan di berbagai platform [sistem operasi](#).

Saat ini kode python dapat dijalankan di berbagai platform [sistem operasi](#), beberapa di antaranya adalah [Linux/Unix](#), [Windows](#), [Mac OS X](#), [Java Virtual Machine](#), [OS/2](#), [Amiga](#), [Palm](#) dan [Symbian](#)

Python didistribusikan dengan beberapa lisensi yang berbeda dari beberapa versi. Lihat sejarahnya di [Python Copyright](#). Namun pada prinsipnya Python dapat diperoleh dan dipergunakan secara bebas, bahkan untuk kepentingan komersial. Lisensi Python tidak bertentangan baik menurut definisi [Open Source](#) maupun [General Public License \(GPL\)](#).

Python dapat digunakan untuk membangun aplikasi-aplikasi yang berjalan pada banyak fungsi (Noprianto, 2002). Diantaranya adalah sebagai berikut :

1. Pengembangan Web dan Internet. Python menyediakan dukungan terhadap pengembangan web dan internet, seperti : penulisan skrip Common Gateway Internet (CGI), pengembangan frameworks seperti Django dan TurboGears, solusi terkini seperti Zope, sistem manajemen konten web dengan Plone. Python juga mendukung secara penuh HTML dan XML, pemrosesan e-mail, pemrosesan RSS feeds serta banyak protokol Internet.

2. Akses terhadap Database. Antarmuka OpenDatabase Connectivity (ODBC) untuk MySQL, Oracle, Microsoft SQL Server,

PostgreSQL, SybaseODBC dan database lainnya yang bebas untuk diunduh. Selain itu tersedia Application Programming Interface (API) dan obyek-obyek untuk database seperti ZODB dan Durus.

3. Pengembangan Graphical User Interface (GUI) pada Desktop. Python menyediakan pustakapustaka pengembangan untuk Desktop, seperti Tk, wxWidgets, GTK+, Qt, kelas-kelas dari Microsoft Foundation melalui perluasan-perluasan win32, Delphi.

4. Keperluan Perhitungan Scientific dan Numeric. Python secara luas digunakan untuk komputasi scientific dan numeris seperti Bioinformatika, kursus Python dalam Bioinformatika serta Fisika.

5. Pengembangan Aplikasi Pendidikan. Python merupakan bahasa pemrograman yang dapat digunakan untuk keperluan pembelajaran, tingkat pengenalan, maupun tingkat lanjutan. Komunitas pendidikan dengan peminatan khusus yang menggunakan Python merupakan salah satu tempat pengembang pemula untuk memulai mengenal Python. PyBiblio merupakan aplikasi untuk pendidikan yang berhubungan dengan banyak sumber yang berbeda. Kelas yang diadakan oleh Software Carpentry, didanai oleh yayasan perangkat lunak Python (PSF).

6. Pemrograman Jaringan Komputer. Sebagai tambahan dari dukungan Python terhadap web dan internet, Python menyediakan dukungan untuk tingkat pemrograman jaringan komputer yang lebih rendah, seperti : antar muka socket jaringan, TwistedPython yang merupakan framework untuk pemrograman jaringan komputer secara asinkron.

7. Pengembangan Perangkat Lunak. Python seringkali digunakan untuk mendukung pengembang perangkat lunak, seperti pembuatan kendali dan manajemen pengembangan, pengetestan dan fungsi lainnya. Aplikasi Python yang seringkali digunakan untuk pengembangan perangkat lunak adalah : Scons untuk membangun kendali, Buildbot dan Apache Gump untuk kompilasi dan pengetestan yang otomatis secara terus menerus, Roundup atau Trac untuk penelusuran bug dan manajemen proyek serta Roster Integrated Development Environments (IDE)

8. Game dan Grafik 3 Dimensi. Python secara luas digunakan untuk pengembangan game, baik untuk keperluan komersil, maupun untuk keperluan hobby. Beberapa aplikasi dan dukungan Python untuk pengembangan game adalah : Python Games yang terdapat pada lingkungan desktop Linux/Unix, PyGame dan PyKyra yang merupakan framework untuk pengembangan game, pustaka-pustaka rendering 3 dimensi yang tersedia untuk Python, PyWeek yang merupakan kontes pengembangan game yang dilangsungkan hampir setiap tahun. Sebagai informasi, game editing software Blender yang berjalan diatas Linux menggunakan Python untuk scripting aplikasi.

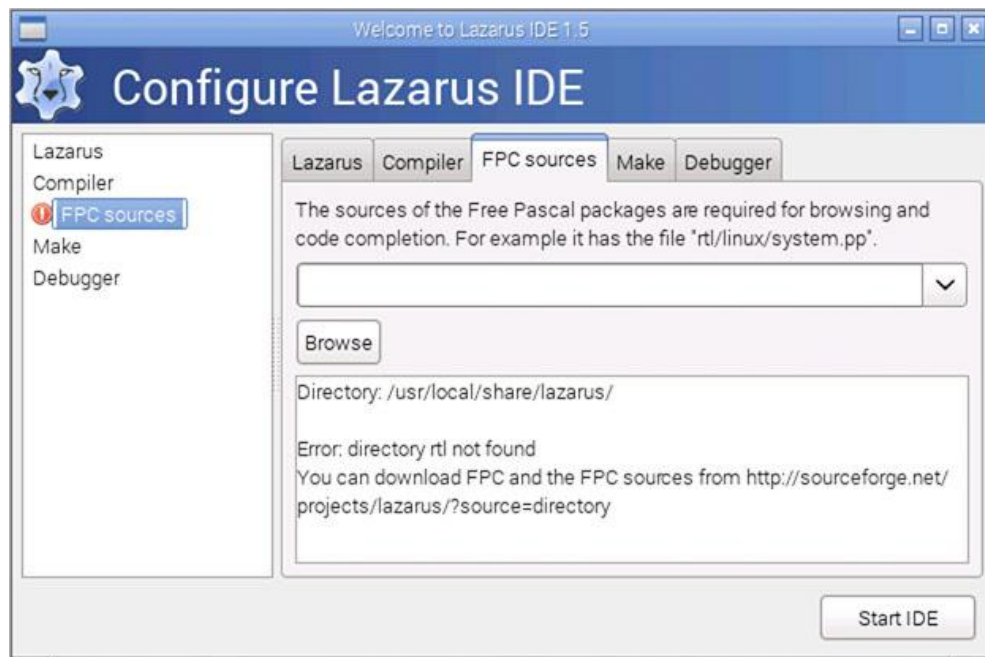
#### **b. Bahasa Pemrograman Pascal**

Pascal adalah [bahasa pemrograman](#) yang pertama kali di buat oleh Profesor [Niklaus Wirth](#), seorang anggota International Federation of Information Processing (IFIP) pada tahun 1971. Dengan mengambil nama dari [matematikawan Perancis, Blaise Pascal](#), yang pertama kali menciptakan mesin penghitung, Profesor Niklaus Wirth membuat bahasa Pascal ini sebagai alat bantu untuk mengajarkan konsep pemrograman [komputer](#) kepada mahasiswanya. Selain itu, Profesor

Niklaus Wirth membuat Pascal juga untuk melengkapi kekurangan-kekurangan bahasa pemrograman yang ada pada saat itu.

### 2.2.8 Lazarus

Lazarus adalah sebuah IDE (Integrated Development Environment), lingkungan pengembangan perangkat lunak yang terintegrasi sehingga pembuatan software menjadi Rapid, dapat diselesaikan dalam waktu singkat. Lazarus bersifat open source, tersedia untuk banyak platform terutama Linux, Windows dan Macintosh. Bahasa pemrograman yang dijadikan landasan dalam Lazarus adalah Pascal.



Gambar 2.9 Configurasi Lazarus IDE

## **BAB III**

### **METODELOGI PENELITIAN**

#### **4.1 Metode Penelitian**

Dalam melakukan penelitian digunakan metode yaitu *Research and Development (R&D)* atau Penelitian dan Pengembangan. Penelitian ini bertujuan untuk mendisain dan membuat sistem point of sale jasa laundry dengan tahap tahap sebagai berikut:

#### **4.2 Analisis Data**

Data keluaran, berupa semua data yang dihasilkan dalam proses pengolahan data, terdiri dari berat cucian , harga paket cucian , data pelanggan, daftar pakaian jasa laundry, transaksi pakaian masuk, dan pengambilan pakaian yang dibutuhkan pada bagian system point of sale laundry.

#### **4.3 Analisis Dokumen**

- a. Dokumen transaksi, merupakan dokumen yang berisikan informasi harga paket, berat cucian, total harga jasa cucian, tanggal pakaian masuk dan pengambilan pakaian.
- b. Dokumen pelanggan, merupakan dokumen yang berisikan daftar pelanggan jasa laundry setiap periode tertentu.
- c. Dokumen jasa pakaian, merupakan dokumen yang berisikan informasi pakaian yang di tempat usaha jasa laundry tersebut.

#### **4.4 Analisis Teknologi**

Teknologi yang digunakan saat ini pada usaha jasa laundry sudah banyak menggunakan komputer tetapi hanya berfungsi untuk mengetik dan mengolah data secara konvensional, komputer yang digunakan belum maksimal dan boros daya. Hal ini terlihat adanya

ketidak efisienan dalam proses kerja sehari-hari. Teknologi yang digunakan sangat berpengaruh pada informasi yang akan disajikan.

Saat ini telah banyak digunakan perangkat mikrokontroler yang lebih sederhana dibandingkan dengan Raspberry Pi dan dikhususkan sebagai komunikator perangkat konverter analog dan digital. artinya mikrokontroler sangat memungkinkan untuk berkomunikasi dengan perangkat sensor terutama sensor yang hanya memproses data pada kasus tertentu.

Dalam penelitian ini akan digunakan Raspberry Pi sebagai perangkat mini computer yang kemudian diintegrasikan dengan sensor load cell yang akan memproses data beban cucian dan ditampilkan pada layar LCD 7 inch, dilanjutkan dengan perhitungan harga berat cucian untuk disimpan ke database dan diprint. Raspberry Pi adalah mini komputer yang dalam memanfaatkan GPIO juga memiliki keterbatasan efektivitas dan efisiensi yang sangat bergantung dengan spesifikasi perangkat dan data yang diproses.

Dengan adanya keterbatasan ini maka beberapa hal dalam penelitian ini perlu diperhatikan yaitu mengoptimalkan penggunaan library pemrograman dan desain antar muka pengguna tanpa mengurangi tingkat keakurasian system dalam menyelesaikan masalah dalam penelitian ini.

#### **4.5 Analisis PIECES**

Analisis PIECES adalah singkatan dari Performance Information Economy Control, Efficiency and Service, penjelasan analisis tersebut dapat dilihat pada penjelasnya tabel dibawah ini.

Tabel 3.1 Deskripsi Metode analisis PIECES

| Analisis           | Sisten Konvensional  | Sistem Baru yang diinginkan   |
|--------------------|--|---|
| <i>Performance</i> | -Jumlah beban kerja lebih besar<br>-pelayanan transaksi pelanggan membutuhkan waktu yang lama  | -Beban kerja lebih ringan sehingga semangat kerja tinggi.<br>-waktu untuk melayani pelanggan cukup singkat            |
| <i>Information</i> | Pencarian data pelanggan dan transaksi cukup lama karena proses pencarian datanya berupa arsip-arsip yang dibuat secara manual belum danya software khusus | -dengan memasukkan noor anggota atau transaksi secara otomatis dapatlangsung diketahui                                |
| <i>Ecomomic</i>    | -Penggunaan tinta dan alat tulis alinnya untuk menyimpan dokumen sangat boros karena terjadi kesalahan tidak dapat digunakan lagi                          | -Dengan system ini akan menghemat waktu dan penggunaan kertas, alat tulis, jika terjadi kesalahan masih dapat diedit. |
| <i>Control</i>     | -Kontrol terhadap dokmen yang kurang teliti akan terjadi kesalahan data  | -data lebih akurat karena aotu koreksi dari system computer.  |
| <i>Effeciency</i>  | -Sumber daya yang digunakan lebih besar  | -Sumber daya yang dibutuhkan lebih kecil  |
| <i>Services</i>    | -Pelanggan harus menunggu lama dalam antrian transaksi karena mencatat kedalam dikument  | -pelayanan terhadap pelanggan lebih cepat karena sudah terintegrasi dengan system                                     |

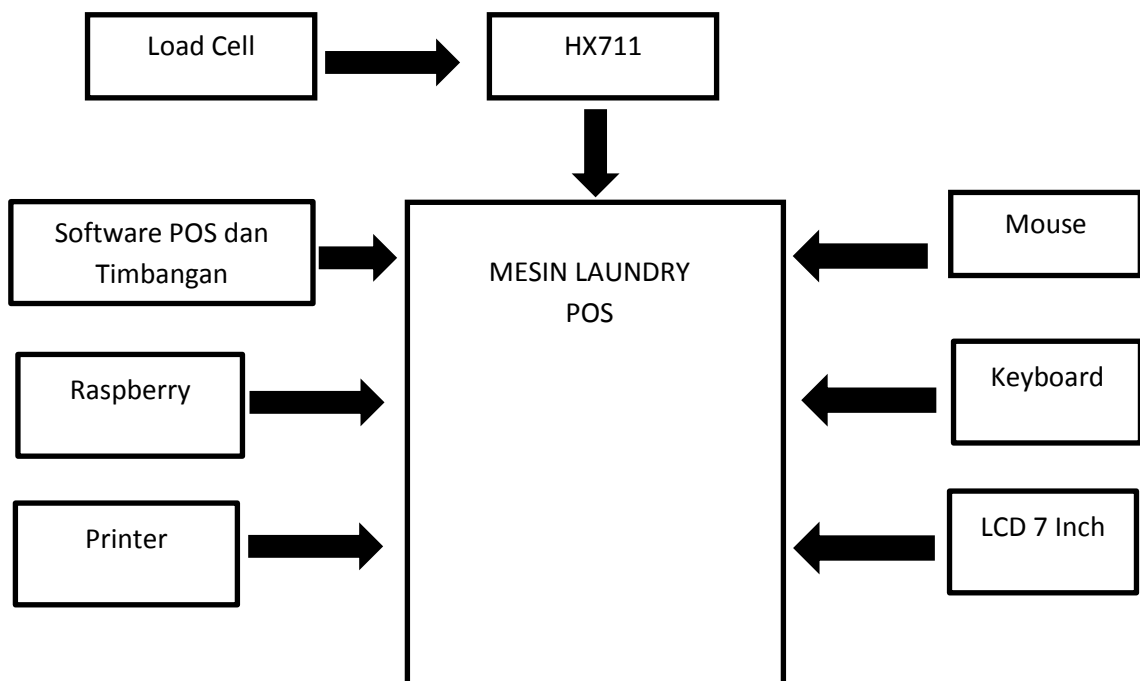
#### 4.6 Konfigurasi Sistem Secara Umum

Secara umum sistem kerja dari proses Alat POS Laundry ini adalah menimbang berat pakaian dan menghitung berdasarkan pengelompokan jenis pakaian. Proses perhitungan dilakukan oleh Mesin

POS Laundry dengan cara menimbang pakaian dengan menaruh pakaian diatas plat mesin POS. Pada bagian input terdiri dari sensor load cell untuk menimbang berat pakaian dan menggunakan RaspberryPi sebagai kontrol.

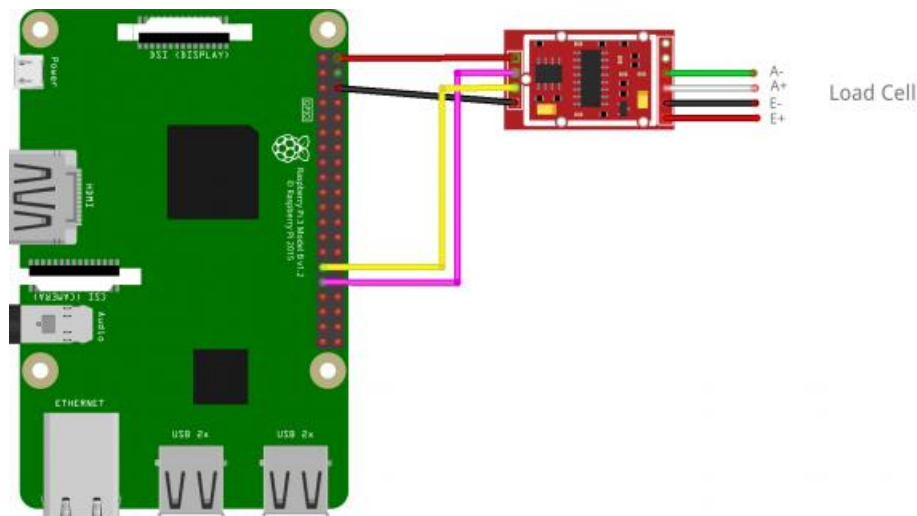
Pada bagian output terhubung pada LCD 7 inch untuk display berat pakaian dan harga, serta terhubung ke printer untuk melihat hasil cetakan pada mesin POS. Sebagai kendali utama pada sistem ini adalah sensor berat, Raspberry PI dan Software Laundry, adapau sensor berat yang digunakan adalah load cell.

Fungsi Software adalah untuk menghitung berat pakaian yang akan dicuci. Sedangkan fungsi dari raspberry PI untuk menjalankan aplikasi POS dengan dukungan bahasa pemrograman, yang menyimpan hasil berat kedalam database dan mengirim output hasil perhitungan ke layar LCD. Blok diagram dari perencanaan ini dapat dilihat pada gambar di bawah ini.



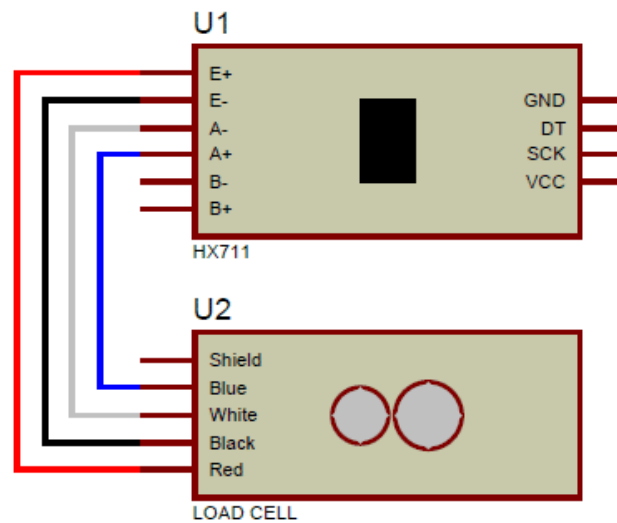
Gambar 3.1 Blok diagram sistem

Pada gambar 3.1 diatas dapat dilihat bahwa laundry point of sale terdiri dari beberapa perangkat yang kemudian disatukan menjadi sebuah alat dimana alat tersebut diberi nama mesin laundry point of sale.



Gambar 3.2 Rangkaian GPIO raspberry pi 3 dengan Modul Hx711

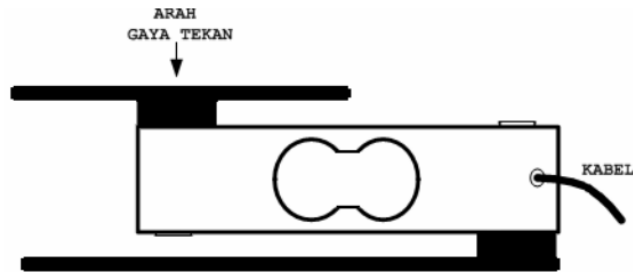
Raspi memiliki fungsi yaitu sebagai tempat database, pusat human machine interface dan pusat pengendalian perangkat lunak. Raspi tidak mempunyai clock (real time clock) internal, sehingga ditambahkan clock eksternal. Konfigurasi RTC eksternal dengan raspi adalah dengan menghubungkan pin-pin RTC dengan pin GPIO raspi. Raspi membutuhkan power supply dengan tegangan 5 Volt dan arus  $\pm 2.0$  A.



Gambar 3.3 Rangkaian Load Cell dengan Modul Hx711

Rangkaian Sensor Timbangan Sensor load cell dihubungkan dengan modul HX711. Load cell mempunyai empat kabel penghubung yaitu VCC (kabel merah), Ground (kabel hitam), Data + (kabel putih), dan Data – (kabel hijau). Modul HX711 mempunyai empat pin sebagai masukan untuk raspberry pi. Pin-pin tersebut adalah Vcc, SCK, DT, dan Ground. Gambar 3.3 merupakan rancangan rangkaian elektrik dari sensor timbangan.

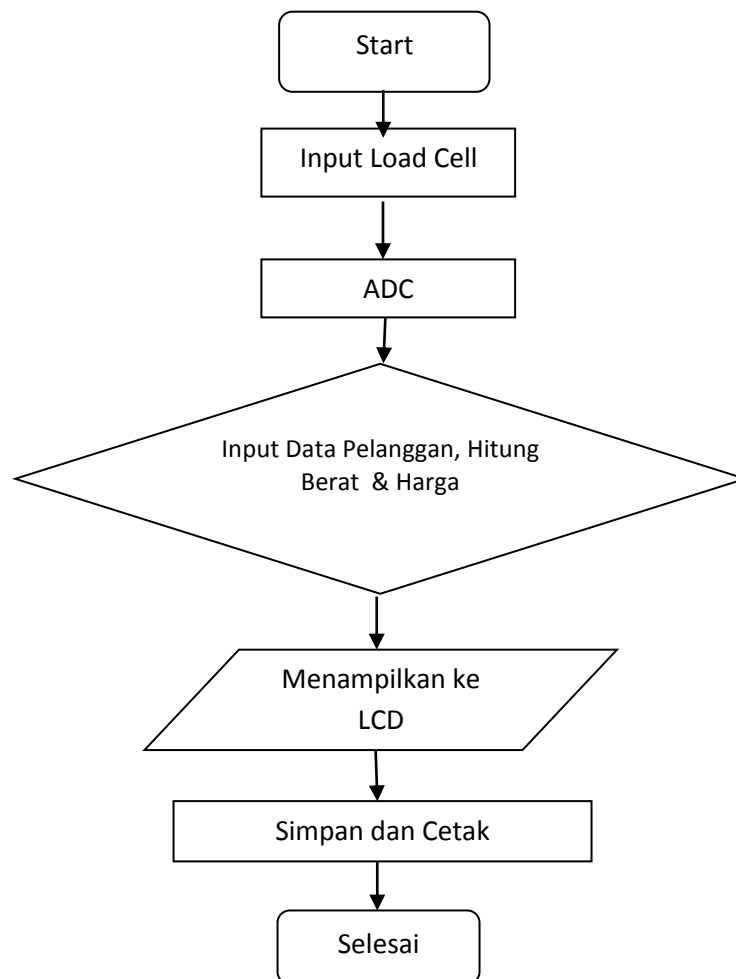
Pada saat load cell digunakan dengan diberi alas di bawahnya dan tempat di atasnya seperti ditunjukkan pada Gambar 3.4 Pada saat dalam kondisi tidak ada beban tegangan keluaran dari load cell tersebut adalah 0v.



Gambar 3.4 Teknik pemasangan load cell

#### 4.7 Tahapan Perencanaan Pada penelitian

Pada penelitian ini sebelum membuat program sebagai kontrol sistem, maka perlu dibuat flowchart sistem untuk mempermudah proses pembuatan program, adapun flowchart yang dirancang adalah sebagai berikut:



Gambar 3.5 Flowchart sistem

Keterangan gambar 3.5 Inialisasi Sensor Load cell sebagai perangkat pengukur berat cucian, yang terhubung ke ADC yaitu mengkonversi sinyal analog ke dalam bentuk besaran yang merupakan rasio perbandingan sinyal input dan tegangan referensi kemudian hasilnya menjadi input bagi raspberry pi, lalu sistem raspberry pi mengkonversi menjadi besaran satuan berat serta menampilkan berat beban cucian hasil dari sensor load cell selanjutnya menghitung harga total cucian dengan mengalikan harga paket cucian yang diinput oleh user dengan berat cucian

kemudian menyimpan dan data ke database dan mencetak dalam bentuk bukti transaksi.

### **Penjelasan Algoritma**

1. Mulai
2. Inisialisasi dan Pemberian beban sensor load cell
3. Proses konversi nilai analog ke data digital
4. Input nilai harga paket dan menghitung total harga
5. Tampilkan ke LCD
6. Simpan hasil perhitungan total harga ke database dan mencetak bukti transaksi
7. Selesai

## BAB IV HASIL DAN PEMBAHASAN

### 4.1 Hasil Perancangan Alat

Dari Hasil penelitian yang dilakukan menghasilkan perancangan alat laundry point of sale yang terintegrasi dengan timbangan digital menggunakan raspberry pi dan kemampuan system menghitung harga dan berat cucian untuk digunakan sebagai perangkat alat penghitung transaksi pada laundry. Perancangan software sebagai interface system yang memiliki tingkat keterbatasan terhadap memory perangkat keras yang digunakan yaitu raspberry pi dan graphic user interface, perancangan alat ini dilakukan dengan mengintegrasikan perangkat keras yaitu raspberry pi, printer dan sensor load cell, dengan dukungan perangkat lunak yaitu bahasa pemrograman python, pascal, database mysql, selanjutnya ke proses desain alat dan menyelesaikan studi kasus antar muka aplikasi berbasis GUI dan pembahasan mekanisme perhitungan berat dan harga cucian kemudian disimpan ke database mysql yang selanjutnya dicetak sebagai bukti transaksi dan data laporan transaksi.

Melalui hasil penelitian yang diperoleh perancangan alat , seperti disajikan pada gambar 4.1 berikut ini:

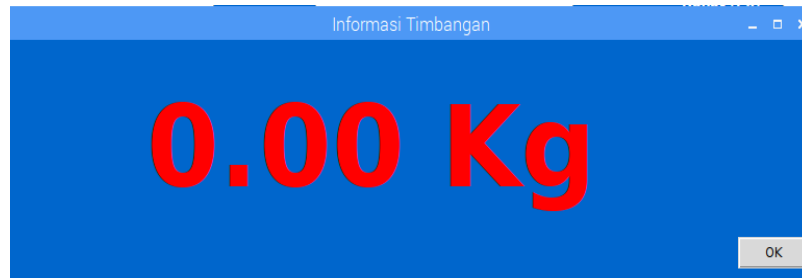


Gambar 4.1 Hasil Perancangan Alat



Gambar 4.2 Tampilan Halaman depan laundry point of sale

Alat dirancang untuk menghitung berat cucian yang ingin dicuci pada tempat usaha jasa laundry, perubahan satuan berat adalah 1 Dekagram x 20 kg, penelitian ini berada di lingkungan jasa laundry dimana rata-rata berat orderan cucian berada pada kisaran 1- 20 kg maka memberikan toleransi perubahan satuan berat dari 1 dekagram sampai maksimal 20 kg, jika satuan berat berada pada 10 gram maka akan membulatkan 1 dekagram begitu pula seterusnya jika berat berada pada 10 dekagram maka akan dibulatkan menjadi satuan hektogram hingga satuan kilogram. Berikut tampilan gambar pengukuran berat pada timbangan serta batas minimum yang tampil pada form aplikasi point of sale pada raspberry pi dengan monitor LCD 7 inchi, seperti gambar diberikut ini:



Gambar 4.3 Pengukuran berat pada timbangan

User dapat melakukan penyimpanan data dengan mengklik tombol ok dan mencetak hasil perhitungan berat cucian dan harga cucian kemudian menutup aplikasi.

## 4.2 Pengujian Sistem

Pada tahapan pengujian sistem ini, yaitu pengujian fungsional untuk menguji kinerja sistem secara keseluruhan, pengujian ini berfokus pada fungsi-fungsi setiap perangkat yang telah dirancang dalam sistem yang dibangun. Menguji dan mengevaluasi akurasi system point of sale tersebut secara langsung dengan menjalankan program sekaligus menguji komponen lainnya ketika telah disatukan.

### 4.2.1 Pengujian Performa Raspberry Pi saat akses GPIO secara continiu menggunakan bahasa pemrograman Python berbasis Grafic User Interface

Pada sistem ini Raspberry Pi yang digunakan adalah Raspberry Pi 3 Model B Memory 1024MB, alat ini difungsikan untuk membaca inputan dari sensor Load Cell dengan menggunakan bahasa pemrograman python tanpa GUI dan dengan dukungan library GUI TKinter, sebagai kesatuan perangkat alat ini yaitu software timbangan dan Poin of sale berbasis raspberry pi dengan GUI.



| No | Alat Ukur Konvensional<br>(kg) | Alat Ukur Point of<br>Sale (kg) | Selisih (%) |
|----|--------------------------------|---------------------------------|-------------|
| 1  | 1 kg                           | 1 kg                            | 0 %         |
| 2  | 2 kg                           | 2 kg                            | 0 %         |
| 3  | 3 kg                           | 3 kg                            | 0 %         |
| 4  | 4 kg                           | 4 kg                            | 0 %         |
| 5  | 5 kg                           | 5 kg                            | 0 %         |

#### 4.3 Pengujian black box

Pengujian dengan menggunakan metode blackbox, yaitu pengujian yang berorientasi pada mekanisme eksternal pada sebuah sistem dan hanya berfokus pada keluaran yang dihasilkan sebagai respon dari pelaksanaan sebuah kondisi yang diinginkan pada pengujian dengan metode blackbox. Berikut hasil pengujian

##### 4.3.1 Pengujian hasil perhitungan total harga cucian

Pengujian ini dilakukan bertujuan untuk melihat sejauh mana kemampuan system menghasilkan output perhitungan harga cucian jasa laundry yang valid. Dalam pengujian ini ditentukan 4 paket harga kiloan jasa cucian yang masing-masing memiliki jumlah harga tertentu berdasarkan jasa yang dipilih oleh pelanggan yaitu paket ekspres dengan harga 10.000/kg paket satu hari dengan harga 7.000/kg, paket dua hari dengan harga 5,000/kg dan paket 3 hari dengan harga 4.000/kg. Selanjutnya harga paket akan dikalikan dengan jumlah berat cucian pelanggan yang menghasilkan nilai total harga jasa cucian.

Berikut tabel validasi tingkat keakuratan sistem point of sale dalam menghitung total harga cucian.

Tabel 4.2 Hasil Validasi tingkat keakuratan perhitungan total harga

| Nota | Tanggal    | Nama Pelanggan | Berat Cucian/Kg | Harga Paket/Rp | Hasil perhitung Point of sale | Hasil Validasi |
|------|------------|----------------|-----------------|----------------|-------------------------------|----------------|
| 501  | 28/08/2017 | Zuhrizal       | 3.45            | 5,000          | 17,250                        | Benar          |
| 502  | 28/08/2017 | Ummi           | 2.18            | 7,000          | 15,260                        | Benar          |
| 503  | 28/08/2017 | Rezky          | 5.27            | 4,000          | 21,080                        | Benar          |
| 504  | 28/08/2017 | Faizah         | 5.45            | 4,000          | 21,800                        | Benar          |
| 505  | 28/08/2017 | Wahyudi        | 3.16            | 5,000          | 15,800                        | Benar          |
| 506  | 28/08/2017 | Juani          | 6.05            | 7,000          | 42,350                        | Benar          |
| 507  | 28/08/2017 | Ilham          | 1.78            | 4,000          | 7,120                         | Benar          |
| 508  | 28/08/2017 | Muhaimim       | 3.56            | 4,000          | 14,240                        | Benar          |
| 509  | 28/08/2017 | Ridha          | 2.63            | 5,000          | 13,150                        | Benar          |

#### 4.3.2 Pengujian hasil penyimpanan database mysql

Pengujian ini dilakukan bertujuan untuk melihat sejauh mana kemampuan akurasi pembacaan dari load cell sehingga system dikategorikan berjalan normal dengan adanya perangkat lunak database mysql berjalan pada perangkat keras raspberry pi dengan menyesuaikan berdasarkan pengujian dilapangan. Sebagaimana gambar dibawa

```
Database changed
mysql> select * from jasa_cucian;
+-----+-----+-----+-----+-----+-----+
| id_jasa_cucian | barcode | nama_cucian | satuan | harga_cucian | harga_setrika |
+-----+-----+-----+-----+-----+-----+
| 3 | 1 | Bedcover Besar | PCS | 25000.00 | 0.00 |
| 4 | 2 | Bedcover Sedang | PCS | 20000.00 | 0.00 |
| 5 | 3 | Bedcover Kecil | PCS | 15000.00 | 0.00 |
| 6 | 4 | Baju Pesta | PCS | 20000.00 | 0.00 |
| 7 | 5 | Jas Satu Pasang | PCS | 25000.00 | 0.00 |
| 8 | 6 | Jas | PCS | 15000.00 | 0.00 |
| 9 | 7 | Boneka Jumbo | PCS | 25000.00 | 0.00 |
| 10 | 8 | Boneka Besar | PCS | 20000.00 | 0.00 |
| 11 | 9 | Boneka Sedang | PCS | 15000.00 | 0.00 |
| 12 | 10 | Boneka Kecil | PCS | 10000.00 | 0.00 |
+-----+-----+-----+-----+-----+-----+
```

Gambar 4.5 Hasil Penyimpanan data Harga Cucian

## 4.4 Implementasi Bahasa Pemrograman

Pada flowchart telah dijelaskan tahapan-tahapan deteksi berat cucian, berikut implementasi fungsi kode program pada raspberry pi dengan menggunakan bahasa pemrograman python dengan dukungan GUI TKinter khusus untuk raspberry pi.

```
import RPi.GPIO as gpio

from Tkinter import*

import time

gpio.setwarnings(False)

gpio.setmode(gpio.BCM)

DT =27

SCK=18

sample=0

kg=0

gpio.setup(SCK, gpio.OUT)

def readCount():

    i=0
```

```
for i in range(24):
    gpio.output(SCK,1)
    Count=Count<<1
    gpio.output(SCK,0)
    if gpio.input(DT) == 0:
        Count=Count+1
    gpio.output(SCK,1)
    Count=Count^0x800000
    gpio.output(SCK,0)
    #gpio.output(23,0)
    return Count
def quit():
    global kg
    print kg
    exit()
root = Tk()
sample= readCount()
def ulangi():
    global sample
    global kg
    count= readCount()
    g=0 #1.000 kg
    dag=0 #100 kg
    ons=0 #10 kg
    g=int((count-sample)/106)
```

```
    dag=g/float(10.0)

    kg=int(dag)/float(100.0)

    lbl["text"] = '{:.0f}'.format(g) + " gram"

    label["text"] = '{:.2f}'.format(kg) + " Kg"

    root.after(1500,ulangi)

root.after(2000,ulangi)

root.title("Informasi Timbangan")

root.geometry('780x215+100+150') #(width = 400, height = 200)

frame1=Frame(root)

frame1.grid()

lbl=Label(root,text="",font="Sans 20 bold",width=10,background='#0066CC')

lbl.grid(row=0,column=0,columnspan=5,sticky='EWNS')

label=Label(root,text="0.00 Kg",font="Sans 75
bold",width=10,backgroun='#0066CC' ,fg='red')

label.grid(row=1,column=0,columnspan=3,sticky='EWNS')

lbl1=Label(root,text="",font="Sans 10 bold",width=10,background='#0066CC')

lbl1.grid(row=2,column=0,columnspan=5,sticky='EWNS')

quitButton = Button(root, text = " OK ", command = quit)

quitButton.grid(row=3,column=4,columnspan=1,sticky='EWNS')

lbl2=Label(root,text="",font="Sans 20 bold",width=10,background='#0066CC')

lbl2.grid(row=4,column=0,columnspan=5,sticky='EWNS')

root.configure(background='#0066CC')

root.mainloop()
```

## **BAB V**

### **KESIMPULAN DAN SARAN**

Bab ini merupakan bab penutup yang berisi kesimpulan dari hasil penelitian dan analisis dari penelitian ini serta saran bagi pihak yang terkait berkenaan dengan pembuatan “Laundry Point Of Sale Embedded System Menggunakan Raspberry Pi”.

#### **5.1. Kesimpulan**

Berdasarkan evaluasi hasil pengujian perangkat Laundry Point Of Sale ini maka penulis dapat membuat kesimpulan sebagai berikut:

1. Mesin Laundry Point Of Sale telah berhasil direalisasikan dan dapat bekerja sesuai dengan tujuan awal yaitu membuat timbangan digital kemudian mengintegrasikan dengan software point of sale sebagai solusi para pengusaha laundry
2. Mesin Laundry Point Of Sale telah berhasil menghitung berat cucian kemudian menampilkan ke LCD sebagai acuan harga transaksi jasa laundry
3. Sistem mampu menampilkan berbagai data-data laporan transaksi laundry.

#### **5.2 Saran**

Berdasarkan pengalaman yang diperoleh selama perancangan dan pembuatan alat ini, ada beberapa saran yang bermanfaat untuk mengembangkan dan menyempurnakan hasil karya berikutnya:

1. Dalam pengembangan alat selanjutnya diharapkan untuk menambahkan fitur agar alat ini mampu terkoneksi ke jaringan internet atau media koneksi lainnya untuk mengirim informasi berat pada timbangan ke perangkat lain.
2. Persamaan untuk kalibrasi nilai berat dalam Gram sebaiknya menggunakan pendekatan yang lebih sensitif untuk meningkatkan kinerja pembacaan nilai berat dalam Gram.
3. Menambahkan jangkauan kapasitas timbangan dan kecepatan dalam proses pembacaan data.

### Daftar Pustaka

- R. Arif Tri Rahmawanto, 2014 : dengan penelitian Pengembangan Timbangan Buah Digital Berbasis Mikrokontroler Atmega16, ISSN 1412-9612 .
- Mirfan, 2016 : melakukan penelitian tentang Mesin Penyaji Beras Secara Digital, Agustus 2016 ,ISSN: 2087-1716
- Arief Cipta Indra Rukmana, April 2014 : melakukan penelitian tentang Aplikasi Sensor Load Cell pada Purwarupa Sistem Sortir Barang. (ISSN: 2088-3714 )
- Kamirul, Desember 2015 : melaukan penelitian Rancang Bangun Data Logger Massa Menggunakan Load Cell, ISBN : 978-602-19655-9-7)
- Try Utami Hidayani,2012 melakukan penelitian rancang bangun timbangan buah digital dengan keluaran berat dan harga.
- Baskoro, I. T., Darjat dan Sudjadi. 2014. *Perancangan Pengontrolan Nyala Lampu dan Kipas Angin pada Sebuah Ruangan Menggunakan Raspberry PI Model B dengan Web GUI*. Transient, Vol. 3, No. 4, Desember 2014, ISSN: 2302-9927, Hal: 567-571.
- Afdali, M. Daud., M. & Raihan Putri. (2017). Perancangan Alat Ukur Digital untuk Tinggi dan Berat Badan dengan Output Suara berbasis Arduino UNO .

- Kamirul., H. Syahwanti., A. Nelvi & M.S. Hendro. (2015). Rancang Bangun Data Logger Massa Menggunakan Load Cell . Prosiding Seminar Kontribusi Fisika
- Syaiful Kasim, 2013, Sensor & Aktuator penerbit Kementrian pendidikan dan kebudayaan.
- Erlangga, W.B, 2011, Rancang Bangun Timbangan Digital Dengan Pemilihan Jenis Buah, Jurusan Teknik Elektro, Universitas Negeri Malang, Malang.
- Noprianto, "Python dan Pemrograman Linux", Yogyakarta: Penerbit ANDI, 2002.
- Khakim, A. L. (2015). Rancang Bangun Alat Timbang Digital Berbasis AVR Tipe Atmega32. Tugas Akhir. Semarang: Universitas Negeri Semarang.
- Nuryanto, R. (2015). Pengukur Berat dan Tinggi Badan Ideal Berbasis Arduino. Karya Ilmiah Program Sarjana. Surakarta: Universitas Muhammadiyah Surakarta.

# LAMPIRAN

## LAMPIRAN

### File view\_paket.pas

```

unit view_paket;
{$mode objfpc}{$H+}
interface
uses
  Classes, SysUtils, sqldb, db, FileUtil, Forms, Controls, Graphics, Dialogs,
  ExtCtrls, DBGrids, StdCtrls;
type
  { TViewPaket }
  TViewPaket = class(TForm)
    Button1: TButton;
    Button2: TButton;
    Button3: TButton;
    Button4: TButton;
    DataSource1: TDataSource;
    DBGrid1: TDBGrid;
    Panel1: TPanel;
    SQLQuery1: TSQLQuery;
    procedure Button1Click(Sender: TObject);
    procedure Button2Click(Sender: TObject);
    procedure Button3Click(Sender: TObject);
    procedure Button4Click(Sender: TObject);
    procedure FormCreate(Sender: TObject);
    procedure FormShow(Sender: TObject);
    procedure Panel1Click(Sender: TObject);
  private
    { private declarations }
  public
    { public declarations }
  end;
var
  ViewPaket: TViewPaket;
implementation
  uses form_paket;
  {$R *.lfm}
  { TViewPaket }
  procedure TViewPaket.FormShow(Sender: TObject);
  begin
    SQLQuery1.Active:=true;
  end;
  procedure TViewPaket.Button4Click(Sender: TObject);
  begin
    Close;
  end;
  procedure TViewPaket.Button1Click(Sender: TObject);
  begin
    FormPaket.Editnm_paket.Text:="";
    FormPaket.Editharga_paket.Text:="";
    FormPaket.Edithari.Text:="";
    FormPaket.Editid_paket.Text:="";
  end;

```

```

    FormPaket.Caption:='Tambah Data Paket';
    FormPaket.ShowModal;
    DBGrid1.DataSource.DataSet.Refresh ;
end;
procedure TViewPaket.Button2Click(Sender: TObject);
begin
    if (DBGrid1.DataSource.DataSet.RecordCount >0) and
(DBGrid1.DataSource.DataSet.Fields[0].AsString<>") then
    begin
        FormPaket.Editid_paket.Text:= DBGrid1.DataSource.DataSet.Fields[0].AsString ;
        FormPaket.Editnm_paket.Text:= DBGrid1.DataSource.DataSet.Fields[1].AsString ;
        FormPaket.Editharga_paket.Text:= DBGrid1.DataSource.DataSet.Fields[2].AsString ;
        FormPaket.Edithari.Text:= DBGrid1.DataSource.DataSet.Fields[3].AsString ;
        FormPaket.Caption:='Ubah Data Paket';
        FormPaket.ShowModal;
        DBGrid1.DataSource.DataSet.Refresh;
    end;
end;
procedure TViewPaket.Button3Click(Sender: TObject);
var
    myYes, myNo: TMsgDlgBtn;
    myButs: TMsgDlgButtons;
begin
    myYes:= mbYes;
    myNo:= mbNo;
    myButs:= [myYes, myNo];
    if (DBGrid1.DataSource.DataSet.RecordCount >0) then
    begin
        if MessageDlg('Hapus Data '+DBGrid1.DataSource.DataSet.Fields[1].AsString+'?',
mtConfirmation, myButs, 0) = mrYes then
        begin
            // hapus temp detail

            DBGrid1.DataSource.DataSet.Delete;
            SQLQuery1.Active:=true;
        end ;
    end
    else
    begin
        MessageDlg('Silahkan pilih data yang ingin dihapus',mtError,[mbOK],0);
    end;

end;
end;
end.
File view_pelanggan.lfm
object ViewPelanggan: TViewPelanggan
    Left = 0
    Height = 534
    Top = 36
    Width = 1022

```

```
BorderStyle = bsSingle
Caption = 'Data Member'
ClientHeight = 534
ClientWidth = 1022
OnCreate = FormCreate
OnShow = FormShow
LCLVersion = '1.2.4.0'
object DBGrid1: TDBGrid
  Left = 0
  Height = 494
  Top = 40
  Width = 1022
  Align = alClient
  AlternateColor = clMenuHighlight
  AutoFillColumns = True
  Color = clHotLight
  Columns = <
    item
      Title.Caption = 'Nama'
      Width = 165
      FieldName = 'nama_pelanggan'
    end
    item
      Title.Caption = 'Pengenal'
      Width = 165
      FieldName = 'kartu_pengenal'
    end
    item
      Title.Caption = 'Kelamin'
      Width = 165
      FieldName = 'kelamin'
    end
    item
      Title.Caption = 'Hp/Telp'
      Width = 165
      FieldName = 'telp_pelanggan'
    end
    item
      Title.Caption = 'Deposit'
      Width = 165
      FieldName = 'deposit_pelanggan'
    end
    item
      Title.Caption = 'Alamat'
      Width = 170
      FieldName = 'alamat_pelanggan'
    end
  end>
DataSource = DataSource1
DefaultRowHeight = 30
FixedColor = clNavy
FixedHotColor = clHighlight
```

```
Font.Color = clWhite
Font.Height = -19
Font.Style = [fsBold]
ParentFont = False
TabOrder = 0
TitleFont.Color = clWhite
TitleFont.Height = -19
TitleFont.Style = [fsBold]
end
object Panel1: TPanel
  Left = 0
  Height = 40
  Top = 0
  Width = 1022
  Align = alTop
  ClientHeight = 40
  ClientWidth = 1022
  Color = clHotLight
  Font.Height = -19
  ParentColor = False
  ParentFont = False
  TabOrder = 1
  OnClick = Panel1Click
object Button1: TButton
  Left = 1
  Height = 38
  Top = 1
  Width = 255
  Align = alLeft
  Caption = 'Tambah'
  Color = clMenuHighlight
  Font.Height = -13
  Font.Name = 'Sans'
  ParentBidiMode = False
  OnClick = Button1Click
  ParentFont = False
  TabOrder = 0
end
object Button2: TButton
  Left = 256
  Height = 38
  Top = 1
  Width = 255
  Align = alLeft
  Caption = 'Ubah'
  Color = clMenuHighlight
  Font.Height = -13
  Font.Name = 'Sans'
  OnClick = Button2Click
  ParentFont = False
  TabOrder = 1
```

```

end
object Button3: TButton
  Left = 511
  Height = 38
  Top = 1
  Width = 255
  Align = alLeft
  Caption = 'Hapus'
  Color = clMenuHighlight
  Font.Height = -13
 OnClick = Button3Click
  ParentFont = False
  TabOrder = 2
end
object Button4: TButton
  Left = 766
  Height = 38
  Top = 1
  Width = 255
  Align = alLeft
  Caption = 'Tutup'
  Color = clMenuHighlight
  Font.Height = -13
  OnClick = Button4Click
  ParentFont = False
  TabOrder = 3
end
end
object DataSource1: TDataSource
  DataSet = SQLQuery1
  left = 248
  top = 184
end
object SQLQuery1: TSQLQuery
  FieldDefs = <>
  Database = DataModule1.MySQL56Connection1
  Transaction = DataModule1.SQLTransaction1
  SQL.Strings = (
    'select * from pelanggan;'
  )
  Params = <>
  left = 208
  top = 130
end
end
file view_pelanggan.pas
unit view_pelanggan;
{$mode objfpc}{$H+}
interface
uses
  Classes, SysUtils, db, sqldb, FileUtil, Forms, Controls, Graphics, Dialogs,

```

```

ExtCtrls, DBGrids, StdCtrls;
type
{ TViewPelanggan }
TViewPelanggan = class(TForm)
  Button1: TButton;
  Button2: TButton;
  Button3: TButton;
  Button4: TButton;
  DataSource1: TDataSource;
  DBGrid1: TDBGrid;
  Panel1: TPanel;
  SQLQuery1: TSQLQuery;
  procedure Button1Click(Sender: TObject);
  procedure Button2Click(Sender: TObject);
  procedure Button3Click(Sender: TObject);
  procedure Button4Click(Sender: TObject);
  procedure FormCreate(Sender: TObject);
  procedure FormShow(Sender: TObject);
  procedure Panel1Click(Sender: TObject);
  procedure Timer1Timer(Sender: TObject);
private
  { private declarations }
public
  { public declarations }
end;
var
  ViewPelanggan: TViewPelanggan;
implementation
  uses form_Pelanggan;
{$R *.lfm}
{ TViewPelanggan }
procedure TViewPelanggan.FormShow(Sender: TObject);
begin
  SQLQuery1.Active:=TRUE;
end;
procedure TViewPelanggan.Button1Click(Sender: TObject);
begin
  FormPelanggan.Editid_pelanggan.Text:= "";
  FormPelanggan.Memoalamat.Text:= "";
  FormPelanggan.Editnama_pelanggan.Text:= "";
  FormPelanggan.Editkartu_pengenal.Text:= "";
  FormPelanggan.ComboBox1.ItemIndex:=0;
  FormPelanggan.Editdeposit.Text:= "";
  FormPelanggan.Edittelp .Text:= "";
  FormPelanggan.Caption:='Tambah Data Member';
  FormPelanggan.ShowModal;
  DBGrid1.DataSource.DataSet.Refresh
end;
procedure TViewPelanggan.Button2Click(Sender: TObject);
//var
// FormPelanggan: TFormPelanggan;

```

```

begin
    if      (DBGrid1.DataSource.DataSet.RecordCount      >0)      and
           (DBGrid1.DataSource.DataSet.Fields[0].AsString<>"") then
    begin
        FormPelanggan.Editid_pelanggan.Text:=
            DBGrid1.DataSource.DataSet.Fields[0].AsString ;
        FormPelanggan.Editkartu_pengenal.Text:=
            DBGrid1.DataSource.DataSet.Fields[1].AsString ;
        FormPelanggan.Editnama_pelanggan.Text:=
            DBGrid1.DataSource.DataSet.Fields[2].AsString ;
        FormPelanggan.Edittelp.Text:= DBGrid1.DataSource.DataSet.Fields[4].AsString ;
        FormPelanggan.Memoalamat.Text:=DBGrid1.DataSource.DataSet.Fields[3].AsString;
        FormPelanggan.Editdeposit.Text:= DBGrid1.DataSource.DataSet.Fields[6].AsString ;

        if DBGrid1.DataSource.DataSet.Fields[7].AsString='Laki-laki' then
            FormPelanggan.ComboBox1.ItemIndex:=0
        else
            FormPelanggan.ComboBox1.ItemIndex:=1;

        FormPelanggan.Caption:='Ubah Data Member';
        FormPelanggan.ShowModal;
        DBGrid1.DataSource.DataSet.Refresh;
    end;
end;
procedure TViewPelanggan.Button3Click(Sender: TObject);
var
    myYes, myNo: TMsgDlgBtn;
    myButs: TMsgDlgButtons;
begin
    myYes:= mbYes;
    myNo:= mbNo;
    myButs:= [myYes, myNo];
    if (DBGrid1.DataSource.DataSet.RecordCount >0) then
    begin
        if MessageDlg('Hapus Data '+DBGrid1.DataSource.DataSet.Fields[1].AsString+?',
            mtConfirmation, myButs, 0) = mrYes then
        begin
            // hapus temp detail

            DBGrid1.DataSource.DataSet.Delete;
            SQLQuery1.Active:=true;
        end ;
    end
    else
    begin
        MessageDlg('Silahkan pilih data yang ingin dihapus',mtError,[mbOK],0);
    end;
end;
procedure TViewPelanggan.Button4Click(Sender: TObject);
begin

```

```

    Close;
end;
end.
File view_pegawai.lfm
object ViewPewangi: TViewPewangi
  Left = 0
  Height = 534
  Top = 36
  Width = 1022
  Caption = 'ViewPewangi'
  ClientHeight = 534
  ClientWidth = 1022
  OnCreate = FormCreate
  OnShow = FormShow
  LCLVersion = '1.2.4.0'
  object DBGrid1: TDBGrid
    Left = 0
    Height = 494
    Top = 40
    Width = 1022
    Align = alClient
    AutoFillColumns = True
    Color = clWindow
    Columns = <
      item
        Title.Caption = 'Nama Pewangi'
        Width = 995
        FieldName = 'nama_pewangi'
      end>
    DataSource = DataSource1
    DefaultRowHeight = 25
    Font.Height = -16
    ParentFont = False
    TabOrder = 0
    TitleFont.Height = -16
  end
  object Panel1: TPanel
    Left = 0
    Height = 40
    Top = 0
    Width = 1022
    Align = alTop
    ClientHeight = 40
    ClientWidth = 1022
    Color = clHotLight
    Font.Height = -19
    ParentColor = False
    ParentFont = False
    TabOrder = 1
    OnClick = Panel1Click
  end
  object Button1: TButton

```

```
Left = 24
Height = 25
Top = 8
Width = 90
Caption = 'Tambah'
Color = clMenuHighlight
Font.Height = -13
Font.Name = 'Sans'
ParentBidiMode = False
OnClick = Button1Click
ParentFont = False
TabOrder = 0
end
object Button2: TButton
Left = 125
Height = 25
Top = 8
Width = 90
Caption = 'Ubah'
Color = clMenuHighlight
Font.Height = -13
Font.Name = 'Sans'
OnClick = Button2Click
ParentFont = False
TabOrder = 1
end
object Button3: TButton
Left = 226
Height = 25
Top = 8
Width = 90
Caption = 'Hapus'
Color = clMenuHighlight
Font.Height = -13
OnClick = Button3Click
ParentFont = False
TabOrder = 2
end
object Button4: TButton
Left = 326
Height = 25
Top = 8
Width = 90
Caption = 'Tutup'
Color = clMenuHighlight
Font.Height = -13
OnClick = Button4Click
ParentFont = False
TabOrder = 3
end
end
end
```

```

object DataSource1: TDataSource
  DataSet = SQLQuery1
  left = 316
  top = 119
end
object SQLQuery1: TSQLQuery
  FieldDefs = <>
  Database = DataModule1.MySQL56Connection1
  Transaction = DataModule1.SQLTransaction1
  SQL.Strings = (
    'select * from pewangi;'
  )
  Params = <>
  left = 421
  top = 189
end
end
file view_rak.lfm
object ViewRak: TViewRak
  Left = 13
  Height = 534
  Top = 36
  Width = 1009
  Caption = 'ViewRak'
  ClientHeight = 534
  ClientWidth = 1009
  OnShow = FormShow
  LCLVersion = '1.2.4.0'
object DBGrid1: TDBGrid
  Left = 0
  Height = 494
  Top = 40
  Width = 1009
  Align = alClient
  AutoFillColumns = True
  Color = clWindow
  Columns = <
    item
      Title.Caption = 'Nama Rak'
      Width = 982
      FieldName = 'nm_rak'
    end>
  DataSource = DataSource1
  DefaultRowHeight = 25
  Font.Height = -16
  ParentFont = False
  TabOrder = 0
  TitleFont.Height = -16
end
object Panel1: TPanel
  Left = 0

```

```
Height = 40
Top = 0
Width = 1009
Align = alTop
ClientHeight = 40
ClientWidth = 1009
Color = clHotLight
Font.Height = -19
ParentColor = False
ParentFont = False
TabOrder = 1
OnClick = Panel1Click
object Button1: TButton
  Left = 24
  Height = 25
  Top = 8
  Width = 90
  Caption = 'Tambah'
  Color = clMenuHighlight
  Font.Height = -13
  Font.Name = 'Sans'
  ParentBidiMode = False
  OnClick = Button1Click
  ParentFont = False
  TabOrder = 0
end
object Button2: TButton
  Left = 125
  Height = 25
  Top = 8
  Width = 90
  Caption = 'Ubah'
  Color = clMenuHighlight
  Font.Height = -13
  Font.Name = 'Sans'
  OnClick = Button2Click
  ParentFont = False
  TabOrder = 1
end
object Button3: TButton
  Left = 226
  Height = 25
  Top = 8
  Width = 90
  Caption = 'Hapus'
  Color = clMenuHighlight
  Font.Height = -13
  OnClick = Button3Click
  ParentFont = False
  TabOrder = 2
end
```

```

object Button4: TButton
  Left = 326
  Height = 25
  Top = 8
  Width = 90
  Caption = 'Tutup'
  Color = clMenuHighlight
  Font.Height = -13
  OnClick = Button4Click
  ParentFont = False
  TabOrder = 3
end
end
object DataSource1: TDataSource
  DataSet = SQLQuery1
  left = 316
  top = 119
end
object SQLQuery1: TSQLQuery
  FieldDefs = <>
  Database = DataModule1.MySQL56Connection1
  Transaction = DataModule1.SQLTransaction1
  SQL.Strings = (
    'select * from rak;'
  )
  Params = <>
  left = 243
  top = 219
end
end
file view_rak.pas
unit view_rak;
{$mode objfpc}{$H+}
interface
uses
  Classes, SysUtils, db, sqldb, FileUtil, Forms, Controls, Graphics, Dialogs,
  DBGrids, ExtCtrls, StdCtrls;
type
  { TViewRak }
  TViewRak = class(TForm)
    Button1: TButton;
    Button2: TButton;
    Button3: TButton;
    Button4: TButton;
    DataSource1: TDataSource;
    DBGrid1: TDBGrid;
    Panel1: TPanel;
    SQLQuery1: TSQLQuery;
    procedure Button1Click(Sender: TObject);
    procedure Button2Click(Sender: TObject);
    procedure Button3Click(Sender: TObject);

```

```

    procedure Button4Click(Sender: TObject);
    procedure FormShow(Sender: TObject);
    procedure Panel1Click(Sender: TObject);
private
    { private declarations }
public
    { public declarations }
end;
var
    ViewRak: TViewRak;
implementation
    uses form_rak;
{$R *.lfm}
{ TViewRak }
procedure TViewRak.Button4Click(Sender: TObject);
begin
    Close;
end;
procedure TViewRak.FormShow(Sender: TObject);
begin
    SQLQuery1.Active:=TRUE;
end;
procedure TViewRak.Button1Click(Sender: TObject);
begin
    FormRak.EditidRak.Text:='' ;
    FormRak.EditRak.Text:= '' ;
    FormRak.Caption:='Tambah Data Rak';
    FormRak.ShowModal;
    DBGrid1.DataSource.DataSet.Refresh;
end;
procedure TViewRak.Button2Click(Sender: TObject);
//var
//FormWarna: TFormWarna;
begin
    //FormWarna := TFormWarna.Create(Self);
    if (DBGrid1.DataSource.DataSet.RecordCount >0) and
        (DBGrid1.DataSource.DataSet.Fields[0].AsString<>'') then
    begin
        FormRak.EditidRak.Text:= DBGrid1.DataSource.DataSet.Fields[0].AsString ;
        FormRak.EditRak.Text:= DBGrid1.DataSource.DataSet.Fields[1].AsString ;
        FormRak.Caption:='Ubah Data Rak';
        FormRak.ShowModal;
        DBGrid1.DataSource.DataSet.Refresh;
    end;
end;
procedure TViewRak.Button3Click(Sender: TObject);
var
    myYes, myNo: TMsgDlgBtn;
    myButs: TMsgDlgButtons;
begin
    myYes:= mbYes;

```

```

myNo:= mbNo;
myButs:= [myYes, myNo];
if (DBGrid1.DataSource.DataSet.RecordCount >0) then
begin
  if MessageDlg('Hapus Data '+DBGrid1.DataSource.DataSet.Fields[1].AsString+'?',
    mtConfirmation, myButs, 0) = mrYes then
  begin
    // hapus temp detail
    DBGrid1.DataSource.DataSet.Delete;
    SQLQuery1.Active:=true;
  end ;
end
else
begin
  MessageDlg('Silahkan pilih data yang ingin dihapus',mtError,[mbOK],0);
end;
end;
end.

```

#### **File view\_transaksi.lfm**

```
object ViewTransaksi: TViewTransaksi
```

```

  Left = 0
  Height = 523
  Top = 11
  Width = 1022
  BiDiMode = bdRightToLeft
  Caption = 'Data Transaksi'
  ClientHeight = 523
  ClientWidth = 1022
  Color = 13395456
  OnShow = FormShow
  ParentBiDiMode = False
  Position = poMainFormCenter
  LCLVersion = '1.2.4.0'
  WindowState = wsMaximized
object DBGrid1: TDBGrid

```

```

  Left = 0
  Height = 435
  Top = 88
  Width = 1022
  Align = alClient
  AlternateColor = 14120960
  AutoFillColumns = True
  Color = 13395456
  Columns = <
  item
    Title.Caption = 'Nota'
    Width = 142
    FieldName = 'id_transaksi_cucian'
  end
  item
    Title.Caption = 'Pelanggan'

```

```
        Width = 142
        FieldName = 'nama_pelanggan'
    end
    item
        Title.Caption = 'Berat'
        Width = 142
        FieldName = 'berat_cucian'
    end
    item
        Title.Caption = 'Paket'
        Width = 142
        FieldName = 'nm_paket'
    end
    item
        Title.Caption = 'Masuk'
        Width = 142
        FieldName = 'tgl_masuk'
    end
    item
        Title.Caption = 'Selesai'
        Width = 142
        FieldName = 'tgl_selesai'
    end
    item
        Title.Caption = 'Pewangi'
        Width = 143
        FieldName = 'nama_pewangi'
    end>
DataSource = DataSource1
DefaultRowHeight = 25
FixedColor = clNavy
FixedHotColor = clHighlight
Font.Color = clWhite
Font.Height = -19
Font.Style = [fsBold]
ParentBiDiMode = False
ParentFont = False
TabOrder = 0
TitleFont.Color = clWhite
TitleFont.Height = -19
TitleFont.Style = [fsBold]
end
object Panel1: TPanel
    Left = 0
    Height = 88
    Top = 0
    Width = 1022
    Align = alTop
    ClientHeight = 88
    ClientWidth = 1022
    Color = 13395456
```

```
Font.Height = -19
ParentColor = False
ParentFont = False
TabOrder = 1
OnClick = Panel1Click
object Button1: TButton
  Left = 8
  Height = 40
  Top = 8
  Width = 196
  Caption = 'Tambah'
  Color = clMenuHighlight
  Font.Height = -15
  Font.Name = 'Sans'
  ParentBidiMode = False
  OnClick = Button1Click
  ParentFont = False
  TabOrder = 0
end
object Button2: TButton
  Left = 416
  Height = 40
  Top = 8
  Width = 204
  Caption = 'Ubah'
  Color = clMenuHighlight
  Font.Height = -15
  Font.Name = 'Sans'
  ParentFont = False
  TabOrder = 1
end
object Button3: TButton
  Left = 624
  Height = 40
  Top = 8
  Width = 204
  Caption = 'Hapus'
  Color = clMenuHighlight
  Font.Height = -15
  ParentFont = False
  TabOrder = 2
end
object Button4: TButton
  Left = 832
  Height = 40
  Top = 8
  Width = 188
  Caption = 'Tutup'
  Color = clMenuHighlight
  Font.Height = -15
  OnClick = Button4Click
```

```
    ParentFont = False
    TabOrder = 3
end
object Button5: TButton
    Left = 208
    Height = 40
    Top = 8
    Width = 204
    Caption = 'Selesai'
    Color = clMenuHighlight
    Font.Height = -15
    Font.Name = 'Sans'
    ParentFont = False
    TabOrder = 4
end
object Label1: TLabel
    Left = 8
    Height = 20
    Top = 56
    Width = 27
    Caption = 'Cari'
    Font.Color = clWhite
    Font.Height = -15
    Font.Style = [fsBold]
    ParentColor = False
    ParentFont = False
end
object ComboBox1: TComboBox
    Left = 48
    Height = 37
    Top = 52
    Width = 364
    Font.Height = -19
    Font.Style = [fsBold]
    ItemHeight = 0
    OnChange = ComboBox1Change
    ParentBidiMode = False
    ParentFont = False
    TabOrder = 5
    Text = 'ComboBox1'
end
object Edit1: TEdit
    Left = 416
    Height = 37
    Top = 52
    Width = 412
    Font.Height = -19
    OnChange = Edit1Change
    ParentFont = False
    TabOrder = 6
    Text = 'Edit1'
```

```
end
object Button6: TButton
  Left = 832
  Height = 35
  Top = 50
  Width = 188
  Caption = 'Cari'
  Color = clMenuHighlight
  Font.Height = -15
  OnClick = Button4Click
  ParentFont = False
  TabOrder = 7
end
end
object DataSource1: TDataSource
  DataSet = SQLQuery1
  left = 280
  top = 192
end
object SQLQuery1: TSQLQuery
  FieldDefs = <
    item
      Name = 'id_transaksi_cucian'
      DataType = ftLargeint
      Precision = -1
      Size = 0
    end
    item
      Name = 'id_user'
      DataType = ftInteger
      Precision = -1
      Size = 0
    end
    item
      Name = 'id_user_selesai'
      DataType = ftInteger
      Precision = -1
      Size = 0
    end
    item
      Name = 'id_user_keluar'
      DataType = ftInteger
      Precision = -1
      Size = 0
    end
    item
      Name = 'id_user_edit'
      DataType = ftInteger
      Precision = -1
      Size = 0
    end
  end
```

```
item
  Name = 'nama_kasir'
  DataType = ftString
  Precision = -1
  Size = 50
end
item
  Name = 'tgl_masuk'
  DataType = ftDate
  Precision = -1
  Size = 0
end
item
  Name = 'tgl_selesai'
  DataType = ftDate
  Precision = -1
  Size = 0
end
item
  Name = 'tgl_diambil'
  DataType = ftDate
  Precision = -1
  Size = 0
end
item
  Name = 'jam_masuk'
  DataType = ftTime
  Precision = -1
  Size = 0
end
item
  Name = 'metode_pembayaran'
  DataType = ftFixedChar
  Precision = -1
  Size = 8
end
item
  Name = 'kode_transfer'
  DataType = ftString
  Precision = -1
  Size = 50
end
item
  Name = 'total_satuan'
  DataType = ftFMTBcd
  Precision = -1
  Size = 2
end
item
  Name = 'total_kiloan'
  DataType = ftFMTBcd
```

```
Precision = -1
Size = 2
end
item
  Name = 'sub_total'
  DataType = ftFMTBcd
  Precision = -1
  Size = 2
end
item
  Name = 'diskon_uang'
  DataType = ftFMTBcd
  Precision = -1
  Size = 2
end
item
  Name = 'grand_total'
  DataType = ftFMTBcd
  Precision = -1
  Size = 2
end
item
  Name = 'uang_bayar'
  DataType = ftFMTBcd
  Precision = -1
  Size = 2
end
item
  Name = 'sisa_bayar'
  DataType = ftFMTBcd
  Precision = -1
  Size = 2
end
item
  Name = 'uang_kembali'
  DataType = ftFMTBcd
  Precision = -1
  Size = 2
end
item
  Name = 'keterangan'
  DataType = ftString
  Precision = -1
  Size = 200
end
item
  Name = 'status_cucian'
  DataType = ftFixedChar
  Precision = -1
  Size = 8
end
```

```
item
  Name = 'berat_cucian'
  DataType = ftBCD
  Precision = -1
  Size = 3
end
item
  Name = 'nama_pewangi'
  DataType = ftString
  Precision = -1
  Size = 50
end
item
  Name = 'id_paket'
  DataType = ftSmallint
  Precision = -1
  Size = 0
end
item
  Name = 'nm_paket'
  DataType = ftString
  Precision = -1
  Size = 50
end
item
  Name = 'harga_paket'
  DataType = ftBCD
  Precision = -1
  Size = 2
end
item
  Name = 'nm_rak'
  DataType = ftString
  Precision = -1
  Size = 50
end
item
  Name = 'id_pelanggan'
  DataType = ftInteger
  Precision = -1
  Size = 0
end
item
  Name = 'nama_pelanggan'
  DataType = ftString
  Precision = -1
  Size = 50
end
item
  Name = 'telp_pelanggan'
  DataType = ftString
```

```

    Precision = -1
    Size = 50
end>
Database = DataModule1.MySQL56Connection1
Transaction = DataModule1.SQLTransaction1
SQL.Strings = (
  'select * from transaksi_cucian limit 10;'
)
Params = <>
left = 114
top = 211
end
end
file view_transaksi.pas
unit view_transaksi;
{$mode objfpc}{$H+}
interface
uses
  Classes, SysUtils, db, sqldb, FileUtil, Forms, Controls, Graphics, Dialogs,
  DBGrids, ExtCtrls, StdCtrls;
type
  { TViewTransaksi }
  TViewTransaksi = class(TForm)
    Button1: TButton;
    Button2: TButton;
    Button3: TButton;
    Button4: TButton;
    Button5: TButton;
    Button6: TButton;
    ComboBox1: TComboBox;
    DataSource1: TDataSource;
    DBGrid1: TDBGrid;
    Edit1: TEdit;
    Label1: TLabel;
    Panel1: TPanel;
    SQLQuery1: TSQLQuery;
    procedure Button1Click(Sender: TObject);
    procedure Button4Click(Sender: TObject);
    procedure ComboBox1Change(Sender: TObject);
    procedure Edit1Change(Sender: TObject);
    procedure FormShow(Sender: TObject);
    procedure Panel1Click(Sender: TObject);
  private
    { private declarations }
  public
    { public declarations }
    procedure FormVar(iSession:TStrings );
  end;
var
  ViewTransaksi: TViewTransaksi;
  indexSession: TStrings;

```

```

implementation
  uses transaksi_masuk;
{$R *.lfm}
{ TViewTransaksi }
procedure TViewTransaksi.Button4Click(Sender: TObject);
begin
  Close;
end;
procedure TViewTransaksi.Button1Click(Sender: TObject);
begin
  FormTransaksiMasuk. FormVar(indexSession);
  FormTransaksiMasuk.Show;
end;
procedure TViewTransaksi.FormShow(Sender: TObject);
begin
  SQLQuery1.Active:=true;
  color:=$00CC6600;
  DBGrid1.AlternateColor:=$00D77800;;
  DBGrid1.Color:=$00CC6600;;
  DBGrid1.FixedHotColor:=$00D77800;;
end;
procedure TViewTransaksi.FormVar(iSession:TStrings );
begin
  indexSession := TStringList.Create;
  indexSession.values['id_user'] := iSession.Values['id_user'] ;
  indexSession.values['user'] := iSession.Values['user'];
  indexSession.values['nama'] :=iSession.Values['nama'] ;
  indexSession.values['level'] := iSession.Values['level'] ;
  indexSession.values['alamat_perusahaan'] := iSession.Values['alamat_perusahaan'] ;
  indexSession.values['nama_perusahaan'] :=
  iSession.Values['nama_perusahaan'] ;
  indexSession.values['telp_perusahaan'] := iSession.Values['telp_perusahaan'] ;
end;
end.

```

**File view\_user.lfm**

```

object ViewUser: TViewUser
  Left = 0
  Height = 534
  Top = 36
  Width = 1022
  BorderStyle = bsSingle
  Caption = 'ViewUser'
  ClientHeight = 534
  ClientWidth = 1022
  OnCreate = FormCreate
  OnShow = FormShow
  Position = poDesktopCenter
  LCLVersion = '1.2.4.0'
  object DBGrid1: TDBGrid
    Left = 0
    Height = 486

```

```

Top = 48
Width = 1022
Align = alClient
AlternateColor = clMenuHighlight
AutoFillColumns = True
Color = clHotLight
Columns = <
  item
    Title.Caption = 'Nama'
    Width = 331
    FieldName = 'nama'
  end
  item
    Title.Caption = 'Username'
    Width = 331
    FieldName = 'user'
  end
  item
    Title.Caption = 'Level'
    Width = 333
    FieldName = 'level'
  end>
DataSource = DataSource1
DefaultRowHeight = 25
FixedColor = clNavy
FixedHotColor = clHighlight
Font.Color = clWhite
Font.Height = -19
Font.Style = [fsBold]
ParentFont = False
TabOrder = 0
TitleFont.Color = clWhite
TitleFont.Height = -19
TitleFont.Style = [fsBold]
end
object Panel1: TPanel
  Left = 0
  Height = 48
  Top = 0
  Width = 1022
  Align = alTop
  ClientHeight = 48
  ClientWidth = 1022
  Color = clHotLight
  Font.Height = -19
  ParentColor = False
  ParentFont = False
  TabOrder = 1
  object Button1: TButton
    Left = 8
    Height = 33

```

```
Top = 8
Width = 255
Caption = 'Tambah'
Color = clMenuHighlight
Font.Height = -13
Font.Name = 'Sans'
ParentBidiMode = False
OnClick = Button1Click
ParentFont = False
TabOrder = 0
end
object Button2: TButton
Left = 264
Height = 33
Top = 8
Width = 255
Caption = 'Ubah'
Color = clMenuHighlight
Font.Height = -13
Font.Name = 'Sans'
OnClick = Button2Click
ParentFont = False
TabOrder = 1
end
object Button3: TButton
Left = 520
Height = 34
Top = 8
Width = 255
Caption = 'Hapus'
Color = clMenuHighlight
Font.Height = -13
OnClick = Button3Click
ParentFont = False
TabOrder = 2
end
object Button4: TButton
Left = 776
Height = 34
Top = 8
Width = 239
Caption = 'Tutup'
Color = clMenuHighlight
Font.Height = -13
OnClick = Button4Click
ParentFont = False
TabOrder = 3
end
end
object DataSource1: TDataSource
DataSet = SQLQuery1
```

```

    left = 216
    top = 136
end
object SQLQuery1: TSQLQuery
    FieldDefs = <>
    Database = DataModule1.MySQL56Connection1
    Transaction = DataModule1.SQLTransaction1
    SQL.Strings = (
        'select * from user;'
    )
    Params = <>
    left = 187
    top = 230
end
end
file view_user.pas
unit view_user;
{$mode objfpc}{$H+}
interface
uses
    Classes, SysUtils, db, sqldb, FileUtil, Forms, Controls, Graphics, Dialogs,
    ExtCtrls, DBGrids, StdCtrls;
type
    { TViewUser }
    TViewUser = class(TForm)
        Button1: TButton;
        Button2: TButton;
        Button3: TButton;
        Button4: TButton;
        DataSource1: TDataSource;
        DBGrid1: TDBGrid;
        Panel1: TPanel;
        SQLQuery1: TSQLQuery;
        procedure Button1Click(Sender: TObject);
        procedure Button2Click(Sender: TObject);
        procedure Button3Click(Sender: TObject);
        procedure Button4Click(Sender: TObject);
        procedure FormCreate(Sender: TObject);
        procedure FormShow(Sender: TObject);
        procedure Timer1Timer(Sender: TObject);
    private
        { private declarations }
    public
        { public declarations }
    end;
var
    ViewUser: TViewUser;
implementation
    uses form_user;
{$R *.lfm}
{ TViewUser }

```

```

procedure TViewUser.FormShow(Sender: TObject);
begin
    SQLQuery1.Active:=TRUE;
end;
procedure TViewUser.Button1Click(Sender: TObject);
begin
    FormUser.Editiduser.Text:="";
    FormUser.Editnama.Text:="";
    FormUser.Editpass.Text:="";
    FormUser.Edituser.Text:="";
    FormUser.Caption:='Tambah Data User';
    FormUser.ShowModal;
    DBGrid1.DataSource.DataSet.Refresh;
end;
procedure TViewUser.Button2Click(Sender: TObject);
var
    FormUser: TFormUser;
begin
    FormUser := TFormUser.Create(Self);

    if (DBGrid1.DataSource.DataSet.RecordCount >0) and
        (DBGrid1.DataSource.DataSet.Fields[0].AsString<>") then
    begin

        FormUser.Editiduser.Text:= DBGrid1.DataSource.DataSet.Fields[0].AsString ;
        FormUser.Editnama.Text:= DBGrid1.DataSource.DataSet.Fields[3].AsString ;
        FormUser.Edituser.Text:= DBGrid1.DataSource.DataSet.Fields[1].AsString ;
        FormUser.Editpass.Text:= DBGrid1.DataSource.DataSet.Fields[2].AsString ;

        if DBGrid1.DataSource.DataSet.Fields[4].AsString='kasir' then
            FormUser.ComboBox1.ItemIndex:=0
        else
            FormUser.ComboBox1.ItemIndex:=1;

        FormUser.Caption:='Ubah Data User';
        FormUser.ShowModal;
        DBGrid1.DataSource.DataSet.Refresh;
    end;
end;
procedure TViewUser.Button3Click(Sender: TObject);
var
    myYes, myNo: TMsgDlgBtn;
    myButs: TMsgDlgButtons;
begin
    myYes:= mbYes;
    myNo:= mbNo;
    myButs:= [myYes, myNo];
    if (DBGrid1.DataSource.DataSet.RecordCount >0) then
    begin
        if MessageDlg('Hapus Data '+DBGrid1.DataSource.DataSet.Fields[1].AsString+'?',
            mtConfirmation, myButs, 0) = mrYes then

```

```

begin
    // hapus temp detail

    DBGrid1.DataSource.DataSet.Delete;
    SQLQuery1.Active:=true;
end ;
end
else
begin
    MessageDlg('Silahkan pilih data yang ingin dihapus',mtError,[mbOK],0);
end;
end;
procedure TViewUser.Button4Click(Sender: TObject);
begin
    Close;
end;
end.

```

#### **File view\_warna.lfm**

```

object ViewWarna: TViewWarna
    Left = 0
    Height = 508
    Top = 62
    Width = 1022
    BorderStyle = bsSingle
    Caption = 'ViewWarna'
    ClientHeight = 508
    ClientWidth = 1022
    OnCreate = FormCreate
    OnShow = FormShow
    Position = poScreenCenter
    LCLVersion = '1.2.4.0'
    object DBGrid1: TDBGrid
        Left = 0
        Height = 460
        Top = 48
        Width = 1022
        Align = alClient
        AlternateColor = clMenuHighlight
        AutoFillColumns = True
        Color = clHotLight
        Columns = <
            item
                Title.Caption = 'Nama Warna'
                Width = 995
                FieldName = 'warna'
            end>
        DataSource = DataSource1
        DefaultRowHeight = 25
        FixedColor = clNavy
        FixedHotColor = clHighlight
        Font.Color = clWhite
    end
end

```

```
Font.Height = -19
Font.Style = [fsBold]
ParentFont = False
TabOrder = 0
TitleFont.Color = clWhite
TitleFont.Height = -19
TitleFont.Style = [fsBold]
end
object Panel1: TPanel
  Left = 0
  Height = 48
  Top = 0
  Width = 1022
  Align = alTop
  ClientHeight = 48
  ClientWidth = 1022
  Color = clHotLight
  Font.Height = -19
  ParentColor = False
  ParentFont = False
  TabOrder = 1
 OnClick = Panel1Click
object Button1: TButton
  Left = 1
  Height = 46
  Top = 1
  Width = 255
  Align = alLeft
  Caption = 'Tambah'
  Color = clMenuHighlight
  Font.Height = -13
  Font.Name = 'Sans'
  ParentBidiMode = False
  OnClick = Button1Click
  ParentFont = False
  TabOrder = 0
end
object Button2: TButton
  Left = 766
  Height = 46
  Top = 1
  Width = 255
  Align = alLeft
  Caption = 'Ubah'
  Color = clMenuHighlight
  Font.Height = -13
  Font.Name = 'Sans'
  OnClick = Button2Click
  ParentFont = False
  TabOrder = 1
end
```

```

object Button3: TButton
  Left = 256
  Height = 46
  Top = 1
  Width = 255
  Align = alLeft
  Caption = 'Hapus'
  Color = clMenuHighlight
  Font.Height = -13
  OnClick = Button3Click
  ParentFont = False
  TabOrder = 2
end
object Button4: TButton
  Left = 511
  Height = 46
  Top = 1
  Width = 255
  Align = alLeft
  Caption = 'Tutup'
  Color = clMenuHighlight
  Font.Height = -13
  OnClick = Button4Click
  ParentFont = False
  TabOrder = 3
end
end
object DataSource1: TDataSource
  DataSet = SQLQuery1
  left = 288
  top = 91
end
object SQLQuery1: TSQLQuery
  FieldDefs = <>
  Database = DataModule1.MySQL56Connection1
  Transaction = DataModule1.SQLTransaction1
  SQL.Strings = (
    'select * from warna;'
  )
  Params = <>
  left = 262
  top = 232
end
end
file view_warna.pas
unit view_warna;
{$mode objfpc}{$H+}
interface
uses
  Classes, SysUtils, mysql55conn, sqldb, db, FileUtil, Forms, Controls,
  Graphics, Dialogs, ExtCtrls, DBGrids, StdCtrls, Buttons;

```

```

type
{ TViewWarna }
TViewWarna = class(TForm)
  Button1: TButton;
  Button2: TButton;
  Button3: TButton;
  Button4: TButton;
  DataSource1: TDataSource;
  DBGrid1: TDBGrid;
  Panel1: TPanel;
  SQLQuery1: TSQLQuery;
  procedure Button1Click(Sender: TObject);
  procedure Button2Click(Sender: TObject);
  procedure Button3Click(Sender: TObject);
  procedure Button4Click(Sender: TObject);
  procedure FormCreate(Sender: TObject);
  procedure FormShow(Sender: TObject);
  procedure Panel1Click(Sender: TObject);
  procedure Timer1Timer(Sender: TObject);
private
  { private declarations }
public
  { public declarations }
end;
var
  ViewWarna: TViewWarna;
implementation
{$R *.lfm}
uses form_warna;
{ TViewWarna }
  procedure TViewWarna.Button4Click(Sender: TObject);
begin
  Close;;
end;
  procedure TViewWarna.Button3Click(Sender: TObject);
var
  myYes, myNo: TMsgDlgBtn;
  myButs: TMsgDlgButtons;
begin
  myYes:= mbYes;
  myNo:= mbNo;
  myButs:= [myYes, myNo];

  if (DBGrid1.DataSource.DataSet.RecordCount >0) then
begin
  if MessageDlg('Hapus Data '+DBGrid1.DataSource.DataSet.Fields[1].AsString+?',
  mtConfirmation, myButs, 0) = mrYes then
begin
  // hapus temp detail
  DBGrid1.DataSource.DataSet.Delete;
  SQLQuery1.Active:=true;

```

```

        end ;
    end
    else
    begin
        MessageDlg('Silahkan pilih data yang ingin dihapus',mtError,[mbOK],0);
    end;
end;
procedure TViewWarna.Button1Click(Sender: TObject);
begin
    FormWarna.EditWarna.Text:="";
    FormWarna.Editidwarna.Text:="";
    FormWarna.Caption:='Tambah Data Warna';
    FormWarna.ShowModal;
    DBGrid1.DataSource.DataSet.Refresh;
end;
procedure TViewWarna.Button2Click(Sender: TObject);
//var
//FormWarna: TFormWarna;
begin
    //FormWarna := TFormWarna.Create(Self);

    if      (DBGrid1.DataSource.DataSet.RecordCount      >0)          and
        (DBGrid1.DataSource.DataSet.Fields[0].AsString<>") then
    begin
        FormWarna.Editidwarna.Text:= DBGrid1.DataSource.DataSet.Fields[0].AsString ;
        FormWarna.Editwarna.Text:= DBGrid1.DataSource.DataSet.Fields[1].AsString ;
        FormWarna.Caption:='Ubah Data warna';
        FormWarna.ShowModal;
        DBGrid1.DataSource.DataSet.Refresh;
    end;
end;
procedure TViewWarna.FormShow(Sender: TObject);
begin
    SQLQuery1.Active:=true;
end;

end.
File view_lainlain.lfm
object ViewLainLain: TViewLainLain
    Left = 312
    Height = 513
    Top = 131
    Width = 909
    Caption = 'ViewLainLain'
    ClientHeight = 513
    ClientWidth = 909
    Color = clHotLight
    OnCreate = FormCreate
    LCLVersion = '1.6.4.0'
    object Button1: TButton

```

```

    Left = 24
    Height = 25
    Top = 16
    Width = 211
    Caption = 'Rak'
    OnClick = Button1Click
    TabOrder = 0
end
object Button2: TButton
    Left = 24
    Height = 25
    Top = 48
    Width = 211
    Caption = 'Pewangi'
    OnClick = Button2Click
    TabOrder = 1
end
object Button3: TButton
    Left = 24
    Height = 25
    Top = 80
    Width = 211
    Caption = 'Profil Perusahaan'
    OnClick = Button3Click
    TabOrder = 2
end
object Button4: TButton
    Left = 24
    Height = 25
    Top = 112
    Width = 211
    Caption = 'P'
    TabOrder = 3
end
end
file view_lainlain.pas
unit view_lainlain;

{$mode objfpc}{$H+}

interface

uses
    Classes, SysUtils, FileUtil, Forms, Controls, Graphics, Dialogs, StdCtrls,
    ExtCtrls;

type

    { TViewLainLain }

    TViewLainLain = class(TForm)

```

```

    Button1: TButton;
    Button2: TButton;
    Button3: TButton;
    Button4: TButton;
    procedure Button1Click(Sender: TObject);
    procedure Button2Click(Sender: TObject);
    procedure Button3Click(Sender: TObject);
    procedure FormCreate(Sender: TObject);
private
    { private declarations }
public
    { public declarations }
end;

var
    ViewLainLain: TViewLainLain;

implementation
    uses view_pewangi,view_rak,form_perusahaan;
    {$R *.lfm}

    { TViewLainLain }
    procedure TViewLainLain.Button1Click(Sender: TObject);
    begin
        ViewRak.ShowModal;
    end;
    procedure TViewLainLain.Button2Click(Sender: TObject);
    begin
        ViewPewangi.ShowModal;
    end;
    procedure TViewLainLain.Button3Click(Sender: TObject);
    begin
        FormPerusahaan.ShowModal;
    end;
    procedure TViewLainLain.FormCreate(Sender: TObject);
    begin
    end;
end.

File view_laporan.lfm
object ViewLaporan: TViewLaporan
    Left = 281
    Height = 395
    Top = 69
    Width = 495
    Caption = 'Laporan'
    ClientHeight = 395
    ClientWidth = 495
    Color = clActiveCaption
    OnCreate = FormCreate
    Position = poMainFormCenter
    LCLVersion = '1.2.4.0'

```

```
object GroupBox1: TGroupBox
  Left = -120
  Height = 344
  Top = -88
  Width = 464
  Caption = 'Pilih Laporan'
  ClientHeight = 322
  ClientWidth = 462
  Font.Color = clWhite
  ParentFont = False
  TabOrder = 0
  object Button1: TButton
    Left = 120
    Height = 40
    Top = 8
    Width = 208
    Caption = 'Laporan Transaksi Harian'
    Font.Color = clBlack
    OnClick = Button1Click
    ParentFont = False
    TabOrder = 0
  end
end
object Button2: TButton
  Left = 136
  Height = 41
  Top = 288
  Width = 208
  Caption = 'Laporan Item Cucian'
  OnClick = Button1Click
  TabOrder = 1
end
object Button3: TButton
  Left = 136
  Height = 41
  Top = 160
  Width = 208
  Caption = 'Grafik Transaksi Harian'
  OnClick = Button3Click
  TabOrder = 2
end
object Button4: TButton
  Left = 136
  Height = 40
  Top = 96
  Width = 208
  Caption = 'Laporan Transaksi Bulanan'
  OnClick = Button1Click
  TabOrder = 3
end
object Button5: TButton
```

```

    Left = 136
    Height = 41
    Top = 224
    Width = 208
    Caption = 'Grafik Transaksi Harian'
    OnClick = Button1Click
    TabOrder = 4
end
end
file view_laporan.pas
unit view_laporan;

{$mode objfpc}{$H+}

interface

uses
    Classes, SysUtils, FileUtil, Forms, Controls, Graphics, Dialogs, StdCtrls,
    Buttons, ExtCtrls, LR_Class, LR_DBSet, sqldb;

type
    { TViewLaporan }
    TViewLaporan = class(TForm)
        Button1: TButton;
        Button2: TButton;
        Button3: TButton;
        Button4: TButton;
        Button5: TButton;
        GroupBox1: TGroupBox;
        procedure Button1Click(Sender: TObject);
        procedure Button3Click(Sender: TObject);
        procedure FormCreate(Sender: TObject);
    private
        { private declarations }
    public
        { public declarations }
    end;
var
    ViewLaporan: TViewLaporan;
implementation
{$R *.lfm}

{ TViewLaporan }
    uses lap_transaksi,view_grafik;
    procedure TViewLaporan.Button1Click(Sender: TObject);
begin
    LapTransaksi.Show ;
end;
    procedure TViewLaporan.Button3Click(Sender: TObject);
begin
    ViewGrafik.Show ;

```

```

end;
end.
File view_pakaian.lfm
object ViewPakaian: TViewPakaian
  Left = 0
  Height = 534
  Top = 36
  Width = 1022
  Caption = 'Data Barang/Cucian'
  ClientHeight = 534
  ClientWidth = 1022
  OnCreate = FormCreate
  OnShow = FormShow
  Position = poDesktopCenter
  LCLVersion = '1.2.4.0'
  object DBGrid1: TDBGrid
    Left = 0
    Height = 486
    Top = 48
    Width = 1022
    Align = alClient
    AlternateColor = clMenuHighlight
    AutoFillColumns = True
    Color = clHotLight
    Columns = <
      item
        Title.Caption = 'Kode'
        Width = 199
        FieldName = 'barcode'
      end
      item
        Title.Caption = 'Nama Cucian'
        Width = 199
        FieldName = 'nama_cucian'
      end
      item
        Title.Caption = 'Satuan'
        Width = 199
        FieldName = 'satuan'
      end
      item
        Title.Caption = 'Harga Kiloan/Unit'
        Width = 199
        FieldName = 'harga_cucian'
      end
      item
        Title.Caption = 'Harga Setrika'
        Width = 199
        FieldName = 'harga_setrika'
      end>
    DataSource = DataSource1
  end
end

```

```
DefaultRowHeight = 30
FixedColor = clNavy
FixedHotColor = clHighlight
Font.Color = clWhite
Font.Height = -19
Font.Style = [fsBold]
ParentFont = False
TabOrder = 0
TitleFont.Color = clWhite
TitleFont.Height = -19
TitleFont.Style = [fsBold]
end
object Panel1: TPanel
  Left = 0
  Height = 48
  Top = 0
  Width = 1022
  Align = alTop
  ClientHeight = 48
  ClientWidth = 1022
  Color = clHotLight
  Font.Height = -19
  ParentColor = False
  ParentFont = False
  TabOrder = 1
  OnClick = Panel1Click
object Button1: TButton
  Left = 1
  Height = 46
  Top = 1
  Width = 255
  Align = alLeft
  Caption = 'Tambah'
  Color = clMenuHighlight
  Font.Height = -13
  Font.Name = 'Sans'
  ParentBidiMode = False
  OnClick = Button1Click
  ParentFont = False
  TabOrder = 0
end
object Button2: TButton
  Left = 256
  Height = 46
  Top = 1
  Width = 255
  Align = alLeft
  Caption = 'Ubah'
  Color = clMenuHighlight
  Font.Height = -13
  Font.Name = 'Sans'
```

```

OnClick = Button2Click
ParentFont = False
TabOrder = 1
end
object Button3: TButton
Left = 511
Height = 46
Top = 1
Width = 255
Align = allLeft
Caption = 'Hapus'
Color = clMenuHighlight
Font.Height = -13
OnClick = Button3Click
ParentFont = False
TabOrder = 2
end
object Button4: TButton
Left = 766
Height = 46
Top = 1
Width = 255
Align = allLeft
Caption = 'Tutup'
Color = clMenuHighlight
Font.Height = -13
OnClick = Button4Click
ParentFont = False
TabOrder = 3
end
end
object DataSource1: TDataSource
DataSet = SQLQuery1
left = 276
top = 212
end
object SQLQuery1: TSQLQuery
FieldDefs = <>
Database = DataModule1.MySQL56Connection1
Transaction = DataModule1.SQLTransaction1
SQL.Strings = (
  'select * from jasa_cucian;'
)
Params = <>
left = 217
top = 204
end
end
file view_pakaian.pas
unit view_pakaian;
{$mode objfpc}{$H+}

```

```

interface
uses
  Classes, SysUtils, sqldb, db, FileUtil, Forms, Controls, Graphics, Dialogs,
  Menus, ExtCtrls, DBGrids, StdCtrls;
type
  { TViewPakaian }
  TViewPakaian = class(TForm)
    Button1: TButton;
    Button2: TButton;
    Button3: TButton;
    Button4: TButton;
    DataSource1: TDataSource;
    DBGrid1: TDBGrid;
    Panel1: TPanel;
    SQLQuery1: TSQLQuery;
    procedure Button1Click(Sender: TObject);
    procedure Button2Click(Sender: TObject);
    procedure Button3Click(Sender: TObject);
    procedure Button4Click(Sender: TObject);
    procedure FormCreate(Sender: TObject);
    procedure FormShow(Sender: TObject);
    procedure Panel1Click(Sender: TObject);
  private
    { private declarations }
  public
    { public declarations }
  end;
var
  ViewPakaian: TViewPakaian;
implementation
  uses form_cucian;
  {$R *.lfm}
  { TViewPakaian }
  procedure TViewPakaian.FormCreate(Sender: TObject);
  begin
  end;
  procedure TViewPakaian.FormShow(Sender: TObject);
  begin
    SQLQuery1.Active:=true;
  end;
  procedure TViewPakaian.Button4Click(Sender: TObject);
  begin
    Close;
  end;
  procedure TViewPakaian.Button1Click(Sender: TObject);
  begin
    FormCucian.Editid_jasa_cucian.Text:= "";
    FormCucian.Editbarcode.Text:= "";
    FormCucian.Editnama_cucian.Text:= "";
    FormCucian.ComboBox1.ItemIndex:=0;
    FormCucian.Editharga_cucian.Text:= "";
  end;

```

```

FormCucian.Editharga_setrika.Text:="";
FormCucian.Caption:='Tambah Data Barang';
FormCucian.ShowModal;
DBGrid1.DataSource.DataSet.Refresh ;
end;
procedure TViewPakaian.Button2Click(Sender: TObject);
begin
if      (DBGrid1.DataSource.DataSet.RecordCount      >0)      and
(DBGrid1.DataSource.DataSet.Fields[0].AsString<>") then
begin
FormCucian.Editid_jasa_cucian.Text:= DBGrid1.DataSource.DataSet.Fields[0].AsString ;
FormCucian.Editbarcode.Text:= DBGrid1.DataSource.DataSet.Fields[1].AsString ;
FormCucian.Editnama_cucian.Text:= DBGrid1.DataSource.DataSet.Fields[2].AsString ;
FormCucian.Editharga_cucian.Text:= DBGrid1.DataSource.DataSet.Fields[4].AsString ;
FormCucian.Editharga_setrika.Text:= DBGrid1.DataSource.DataSet.Fields[5].AsString ;
if DBGrid1.DataSource.DataSet.Fields[3].AsString='KG' then
FormCucian.ComboBox1.ItemIndex:=0
else
FormCucian.ComboBox1.ItemIndex:=1;
FormCucian.Caption:='Ubah Data Barang';
FormCucian.ShowModal;
DBGrid1.DataSource.DataSet.Refresh;
end;
end;
procedure TViewPakaian.Button3Click(Sender: TObject);
var
myYes, myNo: TMsgDlgBtn;
myButs: TMsgDlgButtons;
begin
myYes:= mbYes;
myNo:= mbNo;
myButs:= [myYes, myNo];
if (DBGrid1.DataSource.DataSet.RecordCount >0) then
begin
if MessageDlg('Hapus Data '+DBGrid1.DataSource.DataSet.Fields[1].AsString+?',
mtConfirmation, myButs, 0) = mrYes then
begin
// hapus temp detail
DBGrid1.DataSource.DataSet.Delete;
SQLQuery1.Active:=true;
end ;
end
else
begin
MessageDlg('Silahkan pilih data yang ingin dihapus',mtError,[mbOK],0);
end;
end;
end.
File view_paket.lfm
object ViewPaket: TViewPaket
Left = 0

```

```

Height = 534
Top = 36
Width = 1022
Caption = 'Data Paket'
ClientHeight = 534
ClientWidth = 1022
Color = clHotLight
OnCreate = FormCreate
OnShow = FormShow
Position = poDesktopCenter
LCLVersion = '1.2.4.0'
object DBGrid1: TDBGrid
  Left = 0
  Height = 480
  Top = 54
  Width = 1022
  Align = alClient
  AlternateColor = clMenuHighlight
  AutoFillColumns = True
  Color = clHotLight
  Columns = <
    item
      Title.Caption = 'Nama Paket'
      Width = 331
      FieldName = 'nm_paket'
    end
    item
      Title.Caption = 'Durasi Hari'
      Width = 331
      FieldName = 'hari'
    end
    item
      Title.Caption = 'Harga'
      Width = 333
      FieldName = 'harga_paket'
    end>
  DataSource = DataSource1
  DefaultRowHeight = 30
  FixedColor = clNavy
  FixedHotColor = clHighlight
  Font.Color = clWhite
  Font.Height = -19
  Font.Style = [fsBold]
  ParentFont = False
  TabOrder = 0
  TitleFont.Color = clWhite
  TitleFont.Height = -19
  TitleFont.Style = [fsBold]
end
object Panel1: TPanel
  Left = 3

```

```
Height = 48
Top = 3
Width = 1016
Align = alTop
BorderSpacing.Left = 3
BorderSpacing.Top = 3
BorderSpacing.Right = 3
BorderSpacing.Bottom = 3
ChildSizing.ShrinkHorizontal = crsHomogenousChildResize
ChildSizing.Layout = cclLeftToRightThenTopToBottom
ClientHeight = 48
ClientWidth = 1016
Color = clHotLight
Font.Height = -19
ParentColor = False
ParentFont = False
TabOrder = 1
OnClick = Panel1Click
object Button1: TButton
  Left = 1
  Height = 46
  Top = 1
  Width = 255
  Align = alLeft
  BorderSpacing.Right = 3
  Caption = 'Tambah'
  Color = clMenuHighlight
  Font.Height = -13
  Font.Name = 'Sans'
  ParentBidiMode = False
  OnClick = Button1Click
  ParentFont = False
  TabOrder = 0
end
object Button2: TButton
  Left = 775
  Height = 46
  Top = 1
  Width = 255
  Align = alLeft
  BorderSpacing.Right = 3
  Caption = 'Ubah'
  Color = clMenuHighlight
  Font.Height = -13
  Font.Name = 'Sans'
  OnClick = Button2Click
  ParentFont = False
  TabOrder = 1
end
object Button3: TButton
  Left = 517
```

```
Height = 46
Top = 1
Width = 255
Align = alLeft
BorderSpacing.Right = 3
Caption = 'Hapus'
Color = clMenuHighlight
Font.Height = -13
OnClick = Button3Click
ParentFont = False
TabOrder = 2
end
object Button4: TButton
Left = 259
Height = 46
Top = 1
Width = 255
Align = alLeft
BorderSpacing.Right = 3
Caption = 'Tutup'
Color = clMenuHighlight
Font.Height = -13
OnClick = Button4Click
ParentFont = False
TabOrder = 3
end
end
object DataSource1: TDataSource
DataSet = SQLQuery1
left = 272
top = 256
end
object SQLQuery1: TSQLQuery
FieldDefs = <>
Database = DataModule1.MySQL56Connection1
Transaction = DataModule1.SQLTransaction1
SQL.Strings = (
  'select * from paket;'
)
Params = <>
left = 222
top = 223
end
end
```