**LAPORAN KEGIATAN**

**KERJA PRAKTEK**

***PENGEMBANGAN APLIKASI BERBASIS ANDROID***

***“BAKULPULSA”***

PT. AIRMEDIA PERSADA

Ruko Permai lantai 2-3, Jalan Magelang Km. 4, Yogyakarta

Periode : 25 Juni – 25 Juli 2014

****

Oleh :

Mahendra Harsa Wardhana 5111100045

Muhammad Yunus Bahari 5111100079

**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI INFORMASI**

**INSTITUT TEKNOLOGI SEPULUH NOPEMBER**

**SURABAYA**

**2014**

**LAPORAN KEGIATAN**

**KERJA PRAKTEK**

***PENGEMBANGAN APLIKASI BERBASIS ANDROID***

***“BAKULPULSA”***

PT. AIRMEDIA PERSADA

Ruko Permai lantai 2-3, Jalan Magelang Km. 4, Yogyakarta

Periode : 25 Juni – 25 Juli 2014

****

Oleh :

Mahendra Harsa Wardhana 5111100045

Muhammad Yunus Bahari 5111100079

**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI INFORMASI**

**INSTITUT TEKNOLOGI SEPULUH NOPEMBER**

**SURABAYA**

**2014**

**LEMBAR PENGESAHAN I**

**Judul :**

**Lokasi : PT. AIRMEDIA PERSADA**

**Periode : 25 Juni – 25 Juli 2014**

**Yogyakarta, 25 Juli 2014**

**Mengetahui,**

**Pembimbing Lapangan**

**LEMBAR PENGESAHAN II**

**Judul : Aplikasi Android “BakulPulsa”**

**Lokasi : PT. AIRMEDIA PERSADA**

**Periode : 25 Juni – 25 Juli 2014**

**Surabaya, 25 Juli 2014**

**Mengetahui,**

**Dosen Pembimbing**

**Dr. Tohari Ahmad, S.Kom., MIT.**

**NIP: 132 306 296**

# ABSTRAK

Teknologi informasi saat ini sedang berkembang dengan pesat sehingga banyak yang membutuhkan teknologi informasi untuk memenuhi ataupun membantu pekerjaan sehari-hari.

Salah satu kemajuan teknologi yang dapat membantu pekerjaan sehari-hari adalah *Cloud Storage*. *Cloud Storage* banyak dimanfaatkan untuk menyimpan data tanpa perlu membawa data tersebut secara fisik namun data tersebut tetap tersedia kapanpun dibutuhkan dengan hanya berbekal koneksi internet.

*“BakulPulsa”* adalah aplikasi berbasis android yang menggunakan *Cloud Storage* sebagai penyimpan data transaksi penjualan pulsa handphone yang dibuat oleh penulis untuk memberikan kemudahan dalam proses transaksi oleh kios pulsa handphone.

**Kata kunci :**

**Android, Java, XML, MySQL, PHP, XAMPP, eclipse-android, BakulPulsa**

KATA PENGANTAR

Puji syukur kami sampaikan kepada Tuhan YME karena berkat rahmat dan berkat-Nya lah kami dapat menyelesaikan kegiatan Kerja Praktik di PT. AIRMEDIA PERSADA Yogjakarta.

Kerja Praktik ini dilaksanakan untuk memenuhi salah satu mata kuliah wajib di Institut Teknologi Sepuluh Nopember Fakultas Teknologi Informasi Jurusan Teknik Informatika.Dengan selesainya kerja praktik ini diharapkan apa yang kami lakukan dapat memberikan manfaat untuk pembaca dan seluruh pihak yang terlibat dalam proses Kerja Praktik.

Melalui laporan ini kami ingin menyampaikan rasa terima kasih kepada semua pihak yang telah membantu kami hingga terlaksananya kegiatan Kerja Praktik ini. Ucapan terima kasih ini kami sampaikan sebesar-besarnya kepada :

1. Tuhan YME yang telah memberikan limpahan rahmat-Nya kepada penulis sehingga dapat menyelesaikan seluruh rangkaian proses Kerja Praktik.
2. Keluarga tercinta yang telah memberikan dukungan penuh kepada penulis.
3. Ibu Dr. Eng. Nanik Suciati S.Kom, M.Kom sebagai Ketua Jurusan Teknik Informatika Institut Teknologi Sepuluh Nopember.
4. Bapak Dr. Radityo Anggoro, S.Kom., M.Eng.Sc sebagai Koordinator Kerja Praktik .
5. Bapak Dr. Tohari Ahmad, S.Kom., MIT. sebagai Dosen Pembimbing Kegiatan Kerja Praktik .
6. Bapak Muhammad Irfan Ashshidiq, M.Kom selaku Pembimbing Kerja Praktek dari PT. AIRMEDIA PERSADA.

Apabila terdapat kesalahan dan kekurangan dalam penulisan laporan Kerja Praktik ini penulis memohon maaf yang sebesar-besarnya. Saran dan kritik penulis harapkan sebagai perbaikan dan pembelajaran di kemudian hari. Semoga laporan Kerja Praktik ini dapat memberikan manfaat yang sebaik-baiknya.

Surabaya, 25 Agustus 2014

Penulis

DAFTAR ISI

[ABSTRAK 6](#_Toc416790944)

[KATA PENGANTAR 8](#_Toc416790945)

[DAFTAR ISI 10](#_Toc416790946)

[DAFTAR GAMBAR 14](#_Toc416790947)

[BAB I PENDAHULUAN 18](#_Toc416790948)

[1.1. Latar Belakang 18](#_Toc416790949)

[1.2. Permasalahan 18](#_Toc416790950)

[1.3. Tujuan dan Manfaat Project 19](#_Toc416790951)

[1.4. Batasan masalah 19](#_Toc416790952)

[1.5. Metodologi 19](#_Toc416790953)

[1.6. Waktu Pelaksanaan 20](#_Toc416790954)

[1.7. Sistematika Penulisan 20](#_Toc416790955)

[BAB II PROFIL PERUSAHAAN 22](#_Toc416790956)

[2.1. Tentang Perusahaan 22](#_Toc416790957)

[2.2. Visi dan Misi AirMedia [1] 22](#_Toc416790958)

[2.2.1. Visi 22](#_Toc416790959)

[2.2.2. Misi 23](#_Toc416790960)

[2.3. Tim Inti AirMedia [1] 23](#_Toc416790961)

[BAB III TINJAUAN PUSTAKA 24](#_Toc416790962)

[3.1. ObjectOriented Programming, Basis Data, Model View Controller 24](#_Toc416790963)

[3.1.1. Object Oriented Programming 24](#_Toc416790964)

[3.1.2. Basis Data 24](#_Toc416790965)

[3.1.3. Model View Controller 24](#_Toc416790966)

[3.2. Teknologi yang Digunakan 25](#_Toc416790967)

[3.2.1. Android 25](#_Toc416790968)

[3.2.2. Web Server Apache 25](#_Toc416790969)

[3.2.3. PHP 25](#_Toc416790970)

[3.2.4. MySQL 26](#_Toc416790971)

[3.2.5. HTML 26](#_Toc416790972)

[3.2.6. CSS 26](#_Toc416790973)

[3.3. Aplikasi Perangkat Lunak yang Digunakan 27](#_Toc416790974)

[3.3.1. XAMPP 27](#_Toc416790975)

[3.3.2. Mozilla firefox 27](#_Toc416790976)

[3.3.3. Eclipse Studio 28](#_Toc416790977)

[BAB IV ANALISIS DAN PERANCANGAN SISTEM 30](#_Toc416790978)

[4.1. Deskripsi Umum Sistem 30](#_Toc416790979)

[4.2. Analisis Kebutuhan 30](#_Toc416790980)

[4.3. Perancangan Fungsionalistas 31](#_Toc416790981)

[4.3.1. Diagram Kelas 31](#_Toc416790982)

[4.3.2. Diagram Kasus Penggunaan 32](#_Toc416790983)

[4.4. Perancangan Basis Data 33](#_Toc416790984)

[Berikut merupakan desain dari database dengan menggunakan *Conceptual Data Model* (CDM) dan *Physical Data Model* (PDM). 33](#_Toc416790985)

[4.4.1. CDM 33](#_Toc416790986)

[4.4.2. PDM 35](#_Toc416790987)

[4.5. Perancangan Antarmuka Aplikasi Android 36](#_Toc416790988)

[4.5.1. Tampilan Halaman “Drawable Menu” 36](#_Toc416790989)

[4.5.2. Tampilan Halaman “Home” 36](#_Toc416790990)

[4.5.3. Tampilan Halaman “InputSaldo” 37](#_Toc416790991)

[4.5.4. Tampilan Halaman “Add Customer” 39](#_Toc416790992)

[4.5.5. Tampilan Halaman “Show Customer” 40](#_Toc416790993)

[4.5.6. Tampilan Halaman “Transaction” 41](#_Toc416790994)

[4.5.7. Tampilan Halaman “History” 43](#_Toc416790995)

[4.6. Perancangan Antarmuka Aplikasi Web 44](#_Toc416790996)

[4.6.1. Tampilan Halaman Utama 44](#_Toc416790997)

[4.6.2. Tampilan Halaman Manajemen Saldo 44](#_Toc416790998)

[4.6.3. Tampilan Halaman Manajemen Pelanggan 45](#_Toc416790999)

[4.6.4. Tampilan Halaman Manajemen Dealer Pulsa 45](#_Toc416791000)

[4.6.5. Tampilan Halaman Manajemen Nomor Telepon 46](#_Toc416791001)

[4.6.6. Tampilan Halaman Manajemen Paket Pengisian 46](#_Toc416791002)

[4.6.7. Tampilan Halaman Manajemen Transaksi 47](#_Toc416791003)

[4.6.8. Menu Reporting 47](#_Toc416791004)

[BAB V IMPLEMENTASI 50](#_Toc416791005)

[5.1 Pembuatan User Interface Aplikasi 51](#_Toc416791006)

[5.1.1 Kelas AboutFragment.java 51](#_Toc416791007)

[5.1.2 Kelas CustomerFragmentAdd.java 52](#_Toc416791008)

[5.1.3 Kelas CustomerFragmentShow.java 53](#_Toc416791009)

[5.1.4 Kelas HistoryFragment.java 54](#_Toc416791010)

[5.1.5 Kelas HomeFragment.java 55](#_Toc416791011)

[5.1.6 Kelas SaldoFragment.java 56](#_Toc416791012)

[5.1.7 Kelas TransactionFragment.java 57](#_Toc416791013)

[5.1.8 Kelas NavDrawerListAdapter.java 58](#_Toc416791014)

[5.1.9 Kelas NavDrawerItem.java 60](#_Toc416791015)

[5.1.10 Kelas MainActivity.java 61](#_Toc416791016)

[BAB VI UJI COBA DAN EVALUASI 68](#_Toc416791017)

[5.1 Lingkungan Uji Coba 68](#_Toc416791018)

[5.2 Membuat Dealer Baru 69](#_Toc416791019)

[5.3 Membuat Pelanggan Baru 70](#_Toc416791020)

[5.4 Menambah Nomor pelanggan 71](#_Toc416791021)

[5.5 Menambah Paket Pengisian 72](#_Toc416791022)

[5.6 Membangkitkan XML Saldo 74](#_Toc416791023)

[5.7 Membangkitkan XML Transaksi Terakhir 75](#_Toc416791024)

[5.8 Membangkitkan XML Pencarian Pelanggan 76](#_Toc416791025)

[5.9 Membangkitkan XML Seluruh Transaksi 77](#_Toc416791026)

[5.10 Membangkitkan XML Keuntungan Per Bulan 79](#_Toc416791027)

[BAB VII PENUTUP 80](#_Toc416791028)

[7.1 Kesimpulan 80](#_Toc416791029)

[7.2 Saran 80](#_Toc416791030)

[DAFTAR PUSTAKA 82](#_Toc416791031)

[LAMPIRAN 84](#_Toc416791032)

# DAFTAR GAMBAR

[Gambar 1 Gambar *Class Diagram* aplikasi BakulPulsa 18](#_Toc408223793)

[Gambar 2 Gambar use case diagram 19](#_Toc408223794)

[Gambar 3 Gambar CDM database aplikasi 20](#_Toc408223795)

[Gambar 4 Gambar PDM database aplikasi 21](#_Toc408223796)

[Gambar 5 Gambar tampilan drawable menu 22](#_Toc408223797)

[Gambar 6 Gambar tampilan halaman Home 23](#_Toc408223798)

[Gambar 7 Gambar tampilan halaman InputSaldo 24](#_Toc408223799)

[Gambar 8 Gambar tampilan Add Customer 25](#_Toc408223800)

[Gambar 9 Gambar tampilan Show Customer 26](#_Toc408223801)

[Gambar 10 Gambar tampilan halaman Transaction bagian 1 27](#_Toc408223802)

[Gambar 11 Gambar tampilan halaman Transaction bagian 2 28](#_Toc408223803)

[Gambar 12 Gambar tampilan halaman History 29](#_Toc408223804)

[Gambar 13 Gambar tampilan Halaman Utama 30](#_Toc408223805)

[Gambar 14 Gambar tampilan halaman Manajemen Saldo 30](#_Toc408223806)

[Gambar 15 Gambar tampilan halaman Manajemen Pelanggan 31](#_Toc408223807)

[Gambar 16 Gambar tampilan Majemen Delaer Pulsa 31](#_Toc408223808)

[Gambar 17 Gambar tampilan Manajemen Nomor Telepon 32](#_Toc408223809)

[Gambar 18 Gambar tampilan halaman Manajemen Paket Pengisian 33](#_Toc408223810)

[Gambar 19 Gambar tampilan Halaman Manajemen Transaksi 33](#_Toc408223811)

[Gambar 20 Gambar tampilan Menu Reporting 34](#_Toc408223812)

[Gambar 21 Gambar contoh XML yang akan dikirim 34](#_Toc408223813)

[Gambar 22 Gambar potongan kode sumber Class AboutFragment.java 36](file:///C:\Users\Yunus\Downloads\Laporan%20Kerja%20Praktek.docx#_Toc408223814)

[Gambar 23 Gambar potongan kode sumber Class CustomerFragment.java 37](file:///C:\Users\Yunus\Downloads\Laporan%20Kerja%20Praktek.docx#_Toc408223815)

[Gambar 24 Gambar potongan kode sumber Class CustomerFragmentShow.java 38](file:///C:\Users\Yunus\Downloads\Laporan%20Kerja%20Praktek.docx#_Toc408223816)

[Gambar 25 Gambar potongan kode sumber Class HistoryFragment.java 39](file:///C:\Users\Yunus\Downloads\Laporan%20Kerja%20Praktek.docx#_Toc408223817)

[Gambar 26 Gambar potongan kode sumber Class HomeFragment.java 40](file:///C:\Users\Yunus\Downloads\Laporan%20Kerja%20Praktek.docx#_Toc408223818)

[Gambar 27 Gambar potongan kode sumber Class SaldoFragment.java 41](file:///C:\Users\Yunus\Downloads\Laporan%20Kerja%20Praktek.docx#_Toc408223819)

[Gambar 28 Gambar potongan kode sumber Class TransactionFragment.java 42](file:///C:\Users\Yunus\Downloads\Laporan%20Kerja%20Praktek.docx#_Toc408223820)

[Gambar 29 Gambar potongan kode sumber Class NavDrawerListAdapter.java 44](file:///C:\Users\Yunus\Downloads\Laporan%20Kerja%20Praktek.docx#_Toc408223821)

[Gambar 30 Gambar potongan kode sumber Class NavDrawerItem.java 45](file:///C:\Users\Yunus\Downloads\Laporan%20Kerja%20Praktek.docx#_Toc408223822)

[Gambar 31 Gambar potongan kode sumber Class MainActivity.java 51](file:///C:\Users\Yunus\Downloads\Laporan%20Kerja%20Praktek.docx#_Toc408223823)

[Gambar 32 Gambar tampilan membuatan Dealer Baru 54](#_Toc408223824)

[Gambar 33 Gambar tampilan notifikasi perubahan database dealer baru 54](#_Toc408223825)

[Gambar 34 Gambar membuat pelanggan baru 55](#_Toc408223826)

[Gambar 35 Gambar tampilan notifikasi perubahan database pelanggan 55](#_Toc408223827)

[Gambar 36 Gambar tampilan menambahkan nomor pelanggan 56](#_Toc408223828)

[Gambar 37 Gambar tampilan notifikasi perubahan database nomor pelanggan 56](#_Toc408223829)

[Gambar 38 Gambar tampilan penambahan paket pengisian 57](#_Toc408223830)

[Gambar 39 Gambar tampilan notifikasi perubahan database paket pengisian 57](#_Toc408223831)

[Gambar 40 Gambar tampilan menu pada web 59](#_Toc408223832)

[Gambar 41 Gambar XML dari menu balance yang dikirim 59](#_Toc408223833)

[Gambar 42 Tampilan Menu Report di Web 60](#_Toc408223834)

[Gambar 43 Hasil Pembangkitan XML Transaksi Terakhir 60](#_Toc408223835)

[Gambar 44 Menu Report Pencarian Pelanggan 61](#_Toc408223836)

[Gambar 45 Hasil Pembangkitan XML Pencarian Pelanggan 61](#_Toc408223837)

[Gambar 46 Menu Report Seluruh Transaksi 62](#_Toc408223838)

[Gambar 47 Hasil Pembangkitan XML Seluruh Transaksi 63](#_Toc408223839)

[Gambar 48 Menu Report Keuntungan Perbulan 64](#_Toc408223840)

[Gambar 49 Hasil Pembangkitan XML Keuntungan Perbulan 64](#_Toc408223841)

# BAB I PENDAHULUAN

Bab pendahuluan menjelaskan segala hal yang berkaitan dengan latar belakang, permasalahan, tujuan dan manfaat projek, batasan masalah, waktu pelaksanaan, dan sistematika penulisan yang digunakan dalam Kerja Praktik, dan sistematika penulisan.

* 1. Latar Belakang

Penggunaan alat tulis ataupun jurnal dalam proses transaksi jual beli sangat bagus untuk melakukan manajemen keuangan dan statistik oleh penjual. Namun banyak dari pelaku transaksi jual beli yang masih menggunakan pencatatan secara manual suatu proses transaksi yang dilakukan.

Salah satu kelemahan dari penggunaan proses manual ini ialah kurangnya fleksibilitas dalam pencatatan, dikarenakan setiap terjadi proses transaksi maka penjual akan membutuhkan catatan ataupun jurnal transaksi. Hal ini tentu saja tidak efisien dan mungkin saja dapat menyebabkan kesalahan baik karena tidak teliti ataupun cara pencatatan proses transaksi yang tidak mudah.

Dengan permasalahan inilah penulis dan pembimbing Kerja Praktik merencanakan membuat sebuah aplikasi yang bisa digunakan untuk mempermudah proses pencatatan transaksi jual beli. Terutama untuk penjual yang memiliki banyak transaksi, sebagai contoh adalah distributor pulsa handphone. Sehingga aplikasi yang dibuat dapat digunakan untuk mempermudah penjual untuk melakukan bisnis yang dilakukan dengan lebih mudah dan efisien.

* 1. Permasalahan

Dari hasil diskusi yang telah dilakukan dengan pembimbing Kerja Praktik didapatkan beberapa permasalahan dalam proses pembuatan aplikasi yakni

1. Apa *platform* terbaik yang digunakan untuk melakukan pembuatan aplikasi?
2. Bagaimana aplikasi yang dibuat dapat menyimpan data dari transaksi ke *Cloud Storage*?
   1. Tujuan dan Manfaat Project

Tujuan dan manfaat Kerja Praktik ini diantaranya adalah :

1. Memenuhi mata kuliah Kerja Praktik sebagai syarat untuk mendapatkan gelar Sarjana Teknik.
2. Mendapatkan pengalaman kerja sebagai bekal ketika lulus nanti untuk terjun di dunia kerja.
3. Menerapkan teori–teori yang telah dipelajari selama kuliah dalam dunia kerja.
4. Mengetahui dan lebih mendalami proses pembuatan aplikasi dengan *platform* android
5. Memperdalam bahasa pemrograman PHP sebagai penyedia servis untuk pertukaran data antara android dan system database MySQL.
   1. Batasan masalah

Pada pelaksanaan proses Kerja Praktik ini penulis mengajukan tema Android dan PHP untuk aplikasi penjualan pulsa yang kemudian disebut “BakulPulsa”. Pelaksanaan Kerja Praktik yakni pada periode 25 Juni – 25 Juli 2014. Dalam pengerjaannya penulis menggunakan bahasa Java dan XML pada sisi aplikasi Android, PHP untuk servis penghubung database dan aplikasi Android serta database MySQL untuk menyimpan data.

* 1. Metodologi

Untuk menyelesaikan permasalahan yang diberikan perusahaan, kami menggunakan beberapa metodologi, yaitu :

1. Wawancara dan bimbingan langsung

Selama Kerja Praktik, dalam menentukan bentuk struktur dari sistem yang akan dibuat, penyusun melakukan bimbingan langsung dengan pembimbing lapangan.

1. Studi literatur

Mencari informasi dari sumber–sumber yang terkait dari internet dan buku.

* 1. Waktu Pelaksanaan

Kerja Praktik ini dilaksanakan selama tanggal 25 Juni 2014 s.d. 25 Juli 2014 pada Kantor PT. AIRMEDIA PERSADA Yogyakarta.

* 1. Sistematika Penulisan

Laporan Kerja Praktik ini terbagi menjadi 5 bab dengan rincian sebagai berikut :

**BAB 1 : PENDAHULUAN**

Berisi pendahuluan yang terdiri dari latar belakang, permasalahan, tujuan dan manfaat projek, batasan masalah, waktu pelaksanaan, dan sistematika penulisan yang digunakan dalam Kerja Praktik, dan sistematika penulisan.

**BAB II : PROFIL PERUSAHAAN**

Berisi tentang gambaran umum dari tempat Kerja Praktik yaitu PT. AirMedia Persada, mulai dari sejarah, perkembangan perusahaan, dan tim inti perusahaan.

**BAB III : TINJAUAN PUSTAKA**

Bab ini berisi dasar teori dari metode/teknologi yang digunakan dalam meyelesaikan proyek Kerja Praktik.

**BAB IV : ANALISIS DAN PERANCANGAN**

Pada bab ini dijelaskan mengenai desain antarmuka aplikasi serta desain basis data yang digunakan dalam aplikasi.

**BAB V : IMPLEMENTASI**

Bab ini menjelaskan tentang implementasi *source code* yang telah dibuat maupun yang diperbaiki sehingga proses yang sebelumnya tidak dapat berfungsi dengan baik dapat digunakan.

**BAB VI : UJI COBA DAN EVALUASI**

Pada bab ini menjelaskan tentang proses uji coba fungsi-fungsi *software* dan hasil evaluasinya.

**BAB VII : KESIMPULAN DAN SARAN**

Bab ini berisi kesimpulan dan saran dari proses pelaksanaan Tugas Praktik.

# BAB II PROFIL PERUSAHAAN

Bab ini berisi tentang gambaran umum dari tempat Kerja Praktik yaitu Kantor PT. AIRMEDIA PERSADA.

* 1. Tentang Perusahaan

**PT. AirMedia Persada** adalah perusahaan perangkat lunak yang berkutat dalam pelayanan tentang teknologi informasi dan perancangan sistem informasi. Sistem informasi akademik berbasis web adalah spesialisasi dari perusahaan ini karena telah banyak terbukti dari pekerjaan yang diselesaikan dan produk yang dihasilkan. Beberapa produk dan jasa yang di tangani oleh PT. AirMedia Persada adalah [1]:

1. Sistem Informasi Akademik
2. Sistem Informasi Perpustakaan
3. Sistem Informasi Kepegawaian
4. Sistem Informasi Keuangan
5. Sistem Informasi Pengelolaan Aset
6. Desain Web/Pembuatan Web Profil
7. Pengadaan Hardware Networking/Jaringan
8. Pelatihan di bidang Teknologi Informasi.

Sudah lebih dari 10 tahun PT. AirMedia Persada menangani berbagai macam pesanan perangkat lunak dan dari berbagai macam segmen pemesan mulai dari BUMN, perusahaan, instansi pemerintahan, institusi pendidikan hingga pesanan personal.

* 1. Visi dan Misi AirMedia [1]
     1. Visi

AirMedia Persada bertekad untuk menjadi sebuah Perusahaan Software dan Layanan Teknologi Informasi, dengan Kualifikasi dan Kompetensi Internasional, serta berorientasi Bisnis secara Profesional.

* + 1. Misi
* Mengembangkan Industri Teknologi Informasi dengan orientasi memasyarakatkan Penggunaan Teknologi Informasi, serta berorientasi Bisnis dan Pelayanan yang Profesional.
* Mengakomodasi Potensi Sumber Daya Manusia dan mengembangkan Peluang Bisnis dalam   bidang Teknologi Informasi yang ada.
* Memberikan Kontribusi bagi Perkembangan & Kemajuan Teknologi Informasi.
  1. Tim Inti AirMedia [1]
* Sri Bagus Respati Nugroho, M.T. **(ITS Project Manager)**
* Muhammad Irfan Ashshidiq, M.Kom.**(Lead Systems Analyst & Developer)**
* Aries Kadarisman, S.T.**(Senior Systems Developer)**
* Andri Kusuma Budi, S.T.**(Lead Data & Report Officer)**
* Ifwan Chaniago, S.T.**(Client Service Supervisor*)***
* Wahyu Agustine, M.M.**(Client Service Officer)**
* Ali Rozikan**(Administration Officer)**

# BAB III TINJAUAN PUSTAKA

Bab ini berisi dasar teori dari metode/teknologi yang digunakan dalam meyelesaikan project Kerja Praktik.

* 1. ObjectOriented Programming, Basis Data, Model View Controller
     1. Object Oriented Programming

*Object Oriented Programming* (OOP) adalah satu paradigma untuk pemrograman, alasan mengapa ditirapkan OOP disni karena memudahkan kita untuk melakukan pekerjaan dengan tim selain itu OOP memiliki berbagai fitur seperti *class*, *instances, abstraction, methods,* dan lain-lain yang berguna untuk mempercepat proses pembuatan aplikasi dan melakukan pengecekan performa aplikasi.

* + 1. Basis Data

Basis data digunakan sebagai dasar dalam merancang model penyimpanan data kedalam DBMS (*Database Management System)*, diantara teknik perancangan model penyimpanan yang digunakan adalah perancangan CDM (*Conceptual Data Model*) dan PDM (*Physical Data Model*). CDM merupakan gambaran umum dari entitas utama dari proses bisnis yang dirancang dan relasi antar entitas tersebut [2]. Sedangkan PDM merupakan penjabaran dari CDM dimana terdapat detail atribut dari entitas dan gambaran bagaimana data tersebut disimpan dalam DBMS [3].

* + 1. Model View Controller

*Framework* yang digunakan sudah mendukung arsitektur berbentuk *Model View Controller (MVC*). Hal ini memudahkan *programmer* dalam mengembangkan aplikasi karena file termanajemen dengan baik, logika pemrograman ada di *controller*, penanganan data ada di model, dan *view* menangani tampilan aplikasi ke pengguna [4].

* 1. Teknologi yang Digunakan
     1. Android

Android adalah sistem operasi turunan dari unix yang memiliki spesialisasi penggunaan pada perangkat *mobile*. Pada perkembangannya android terus dikembangkan untuk digunakan ke berbagai perangkat diantaranya mobil pintar, televisi, kulkas, dan berbagai perangkat lainnya, dengan kontribusi dari berbagai pengembang linux O*penSource*, pengembang perangkat keras, pengembang perangkat lunak dan provider pendamping android telah menjadi salah satu Sistem Operasi yang sangat cepat berkembang [5].

Android berjalan pada Dalvik yakni proses *virtual machine* pada Android. Bahasa pemrograman java yang digunakan akan dikompilasi menjadi kode *byte Java Virtual Machine* yang kemudian akan diterjemahkan menjadi kode *byte* Dalvik dan disimpan pada berkas .dex (*Dalvik EXecutable*) [6].

* + 1. Web Server Apache

*Apache* adalah *webserver* yang dapat dijalankan di banyak sistem operasi yang berguna untuk melayani dan memfungsikan situs web. Protokol yang digunakan untuk melayani fasilitas web menggunakan HTTP [7].

* + 1. PHP

PHP adalah salah satu bahasa pemrograman yang berjalan pada sebuah *web server* dan berfungsi sebagai pengolah data pada sebuah *server*. Dengan menggunakan PHP sebuah *website* akan lebih interaktif dan dinamis [8].

Keunggulan PHP [8]

* Tingkat akses PHP yang lebih cepat dan memiliki tingkat keamanan yang tinggi karena kode dieksekusi di *server* dan *client* hanya menerima hasil pemrosesan kode berupa halaman HTML.
* PHP memiliki tingkat pengembanganyang cepat, sehingga selalu mengikuti perkembangan teknologi internet.
* PHP mampu berjalan di beberepa *server* yang ada, misalnya Apache, Microsoft IIS, PWS, AOLserver, phttpd, dan Xitami.
* PHP juga mendukung akses ke beberapa *database* yang sudah ada, baik yang bersifat *free*/gratis ataupun komersial. *Database* itu antara lain MySQL, PosgreSQL, mSQL, Informix, MicrosoftSQL server.
  + 1. MySQL

MySQL dalah sebuah perangkat lunak Pembuat *database* yang bersifat terbuka atau *open source* dan berjalan disemua *platform* baik Linux maupun Windows, MySQL merupakan program pengakses *database* yang bersifat *network* sehingga dapat digunakan untuk aplikasi *Multi User* (Pengguna Banyak).

* + 1. HTML

HTML adalah, (*HyperText Markup Language*) sebuah bahasa standar yang digunakan oleh *browser Internet* untuk membuat halaman dan dokumen pada sebuah Web yang kemudian dapat diakses dan dibaca layaknya sebuah artikel [9].

* + 1. CSS

*Cascading Style Sheet (CSS)* merupakan salah satu bahasa pemrograman web  untuk mengendalikan beberapa komponen dalam sebuah web sehingga akan lebih terstruktur dan seragam. Sama halnya *styles* dalam aplikasi pengolahan kata seperti Microsoft Word yang dapat mengatur beberapa *style*, misalnya *heading, bodytext, footer, images,* dan *style* lainnya untuk dapat digunakan bersama-sama dalam beberapa file. Pada umumnya CSS dipakai untuk memformat tampilan halaman web yang dibuat dengan bahasa HTML dan XHTML [10].

* 1. Aplikasi Perangkat Lunak yang Digunakan
     1. XAMPP

XAMPP adalah sebuah *software web server* *apache* yang didalamnya sudah tersedia *database server* *mysql* dan *support* *php programming*. XAMPP merupakan *software* yang mudah digunakan, gratis dan mendukung instalasi di Linux dan Windows. Keuntungan lainnya adalah cuma menginstal satu kali sudah tersedia Apache Web Server, MySQL Database Server, PHP Support (PHP 4 dan PHP 5) dan beberapa module lainnya. Hanya bedanya kalau yang versi untuk Windows sudah dalam bentuk instalasi grafis dan yang Linux dalam bentuk file terkompresi tar.gz. Kelebihan lain yang berbeda dari versi untuk Windows adalah memiliki fitur untuk mengaktifkan sebuah server secara grafis, sedangkan Linux masih berupa perintah-perintah di dalam console.

* + 1. Mozilla firefox

*Browser* yang berguna untuk melihat tampilan web pada aplikasi yang dikembangkan. Tools lain yang digunakan pada *browser* firefox ini adalah firebug. Firebug berguna dalam melakukan *debug* dan *test client script* seperti javascript dan jquery. Terdapat fitur *console* yang berguna melihat *error* apa yang terjadi pada *client script* dan baris berapa *error* itu berada. Selain itu firebug digunakan untuk melihat CSS dan HTML yang kita buat sudah sesuai dengan yang diinginkan atau belum.

* + 1. Eclipse Studio

Eclipse adalah IDE (*Integrated Development Environtment*) yang digunakan sebagai media pembuatan program bakul pulsa. Eclipse menunjang berbagai bahasa pemrograman dan disertai berbagai macam plug-in, diantara banyak bahasa pemrograman yang ditunjang eclipse adalah java yang digunakan sebagai bahasa pemrograman aplikasi android.

# BAB IV ANALISIS DAN PERANCANGAN SISTEM

* 1. Deskripsi Umum Sistem

Aplikasi *BakulPulsa* merupakan sistem informasi berbasis android dan *web service* yang digunakan untuk mengorganisir penjualan pulsa pada tingkat eceran. Pengguna bias memasukkan transaksi, mengecek saldo akhir, mengecek transaksi terakhir, mengecek konsumen yang belum membayar, memeasukkan dan mencari nomor dan nama konsumen, serta dapat melihat laporan secara online.

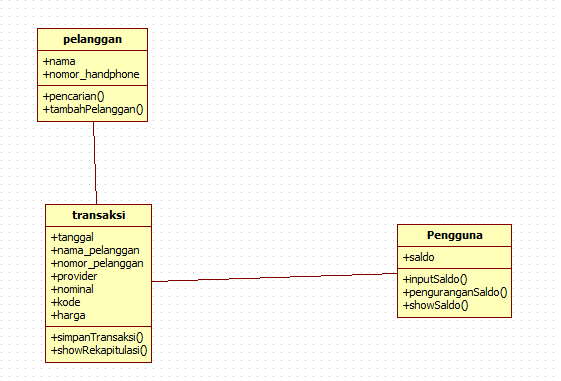
* 1. Analisis Kebutuhan

Kebutuhan fungsional pengguna dalam transaksi jual beli pulsa tingkat pengecer adalah sebagai berikut :

* Sistem dapat menampilkan saldo terakhir.
* Sistem dapat menghitung saldo akhir dengan otomatis.
* Sistem dapat menampilkan transaksi terakhir.
* Sistem dapat menambahkan data pelanggan.
* Sistem dapat melakukan pencarian berdasarkan nomor handphone atau nama pelanggan dan menampilkannya.
* Sistem dapat menambahkan data transaksi.
* Sistem dapat menyimpan data transaksi.
* Sistem dapat melihat rekapan transaksi berdasarkan tahun.
* Sistem dapat melihat laporan dari web.

Sedangkan untuk kebutuhan non-fungsional adalah sebagai berikut :

* *User friendly*, aplikasi ini dapat dengan mudah digunakan oleh pengguna.
  1. Perancangan Fungsionalistas
     1. Diagram Kelas

****

Gambar 1 Gambar Class Diagram aplikasi BakulPulsa

Gamber 1 adalah rancangan diagram kelas untuk aplikasi *BakulPulsa* ini. Terdapat tiga kelas, yaitu kelas pelanggan, transaksi, dan kelas pengguna. Ketiga kelas tersebut memiliki hubungan asosiasi.

* Kelas Pelanggan

Kelas pelanggan berisikan fungsi-fungsi yang berhubungan dengan pelanggan, seperti pencarian dara pelanggan dan menambahkan pelanggan.

* Kelas Transaksi

Kelas yang berisi fungsi-fungsi yang berhubungan dengan transaksi, seperti menyimpan data transaksi dan memperlihatkan rekapitulasi transaksi.

* Kelas Pengguna

Kelas pengguna berisi fungsi-fungsi untuk keperluan mengelolah data pengguna, seperti menginput saldo, fungsi pengurangan saldo,dan memperlihatkan saldo.

* + 1. Diagram Kasus Penggunaan

Berikut adalah usecase diagram dari aplikasi *BakulPulsa*

****

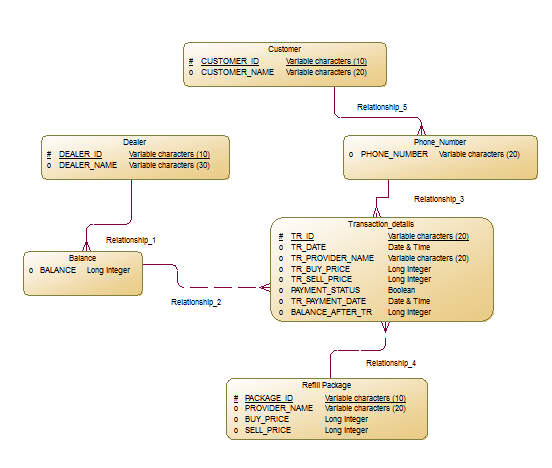
Gambar 2 Gambar use case diagram

Seperti pada Gambar 2, pengguna dapat melakukan tujuh buah proses. Pada proses pertama, pengguna dapat melihat saldo terakhir. Saldo akan diperbaharui datanya secara otomatis setelah transaksi terjadi. Pada proses kedua, pengguna dapat melihat transasi yang paling terakhir dilakukan. Pada proses ketiga, pengguna dapat menambahakan data pelanggan ke dalam aplikasi. Data pelanggan akan disimpan dapat dilihat salah satunya dengan proses keempat, yaitu mencari data pelanggan. Pada proses kelima, pengguna dapat melihat transaksi yang telah dilakukan. Pada proses keenam, pengguna dapat melihat seluruh laporan dari transaksi yang telah dilaksanakan. Pada proses ketujuh, pengguna dapat menambahkan data transaksi yang telah dilakukan.

* 1. Perancangan Basis Data

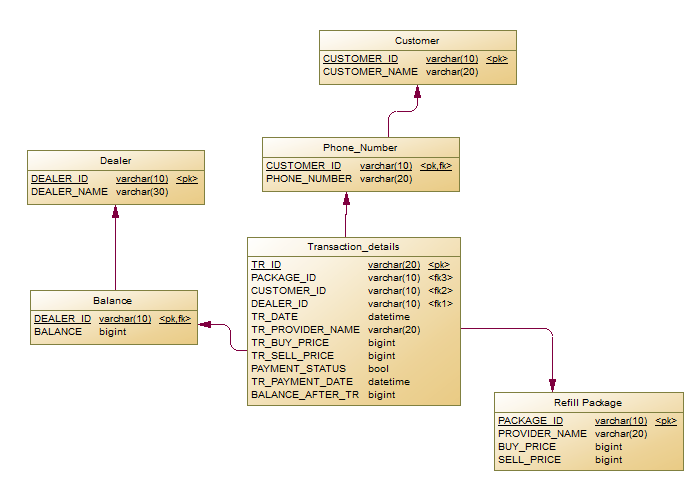
Berikut merupakan desain dari database dengan menggunakan *Conceptual Data Model* (CDM) dan *Physical Data Model* (PDM).

* + 1. CDM



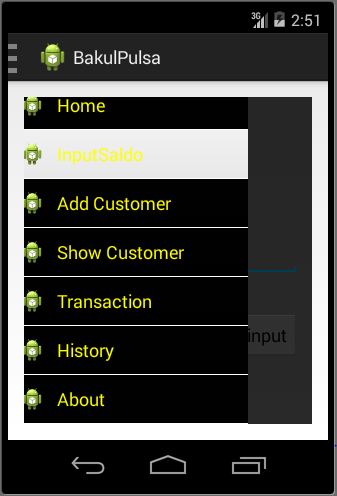
Gambar 3 Gambar CDM database aplikasi

* + 1. PDM



Gambar 4 Gambar PDM database aplikasi

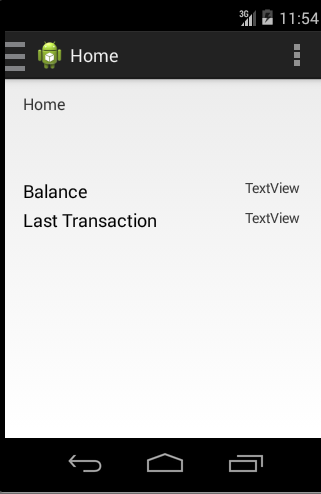
* 1. Perancangan Antarmuka Aplikasi Android
     1. Tampilan Halaman “Drawable Menu”



Gambar 5 Gambar tampilan drawable menu

Gambar 5 merupakan tampilan dari drawable menu, dan pada masing-masing bagian merupakan halaman yang berbeda sesuai nama yang tertera.

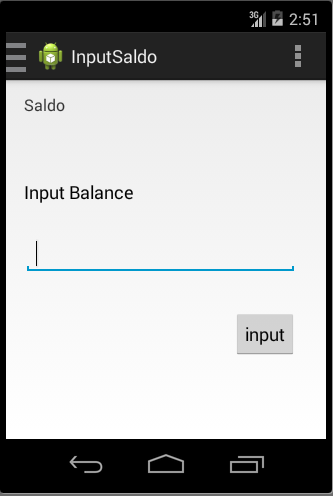
* + 1. Tampilan Halaman “Home”



Gambar 6 Gambar tampilan halaman Home

Pada gambar 6, “Balance” merupakan sisa saldo yang dimiliki oleh user, dan “Last Transaction” merupakan tanggal terakhir pelanggan melakukan transaksi.

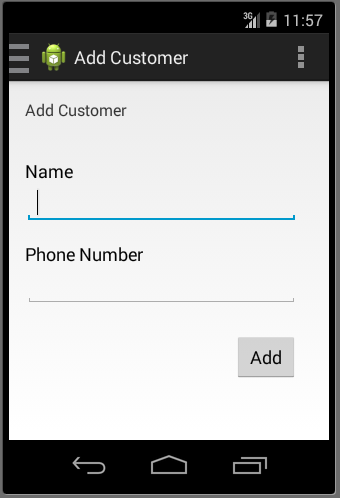
* + 1. Tampilan Halaman “InputSaldo”



Gambar 7 Gambar tampilan halaman InputSaldo

Gambar 7 merupakan halaman untuk memasukan saldo awal per pembelian saldo. Isikan nominal saldo dan tekan tombol input.

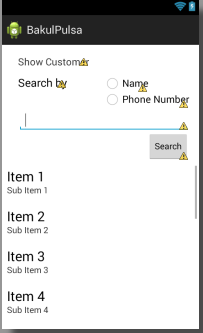
* + 1. Tampilan Halaman “Add Customer”



Gambar 8 Gambar tampilan Add Customer

Gambar 8 adalah halaman untuk menambahkan pelanggan. Kolom pertama untuk mengisikan nama dan kolom kedua untuk mengisikan nomor handphone. Setelah diisi maka tekan tombol “Add” untuk menambahkan.

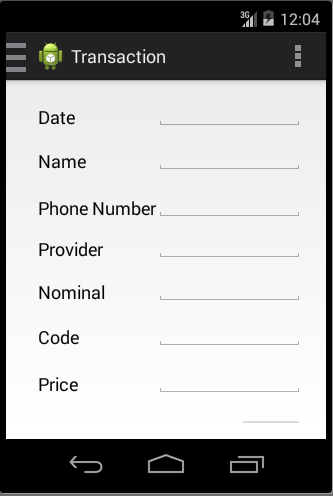
* + 1. Tampilan Halaman “Show Customer”



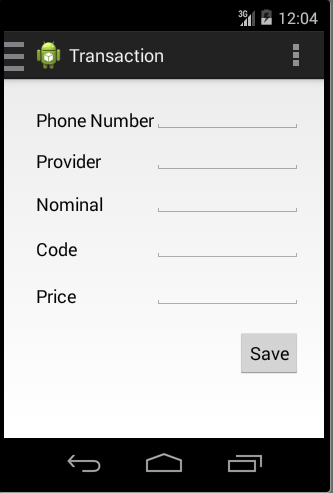
Gambar 9 Gambar tampilan Show Customer

Gambar 9 merupakan hasil yang akan ditampilkan ketika pencarian telah selesai dilakukan. Pada “Item” akan memunculkan nama dan pada “Sub Item” akan memunculkan nomor handphone.

* + 1. Tampilan Halaman “Transaction”



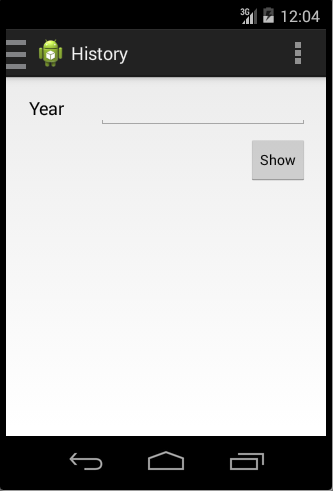
Gambar 10 Gambar tampilan halaman Transaction bagian 1



Gambar 11 Gambar tampilan halaman Transaction bagian 2

Gambar 10 dan 11 merupakan halaman untuk memasukan data transaksi yang sudah. Setelah mengisi semua entry maka selanjutnya tekan tombol “Save” untuk menyimpan data.

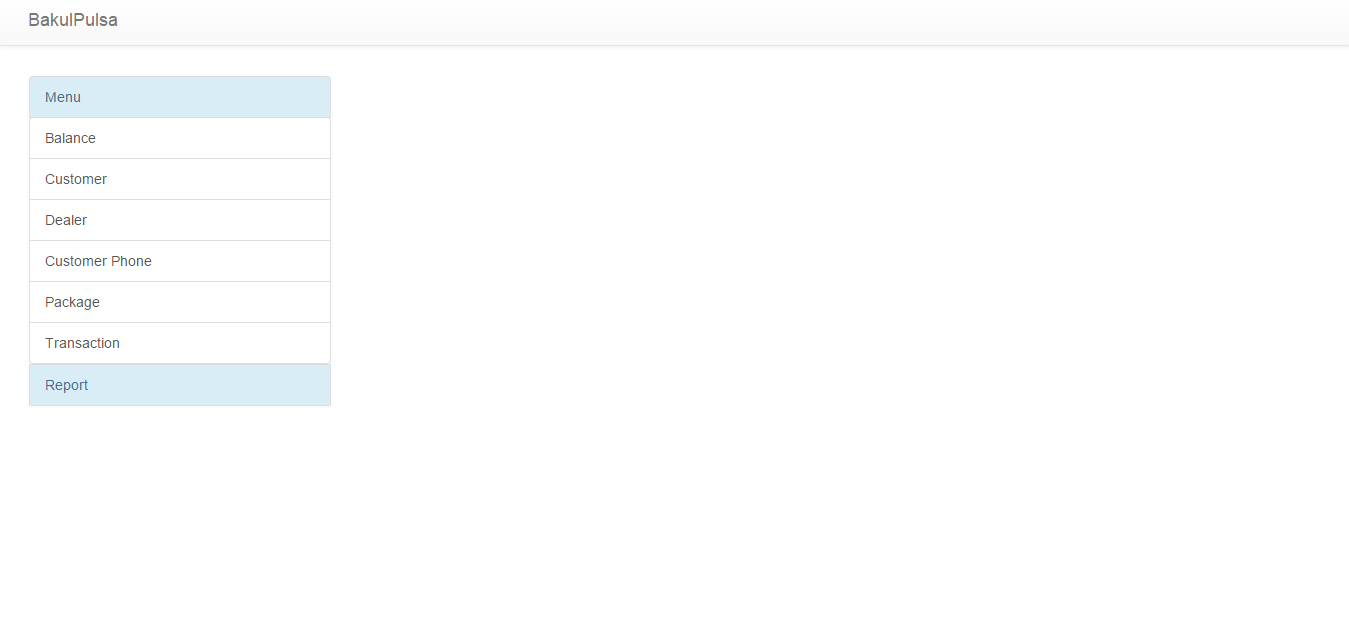
* + 1. Tampilan Halaman “History”



Gambar 12 Gambar tampilan halaman History

Gambar 13 merupakan halaman “history”. Kolom “Year” berfungsi untuk memasukan tahun yang diinginkan. Setelah itu tekan show untuk memunculkan rekapan.

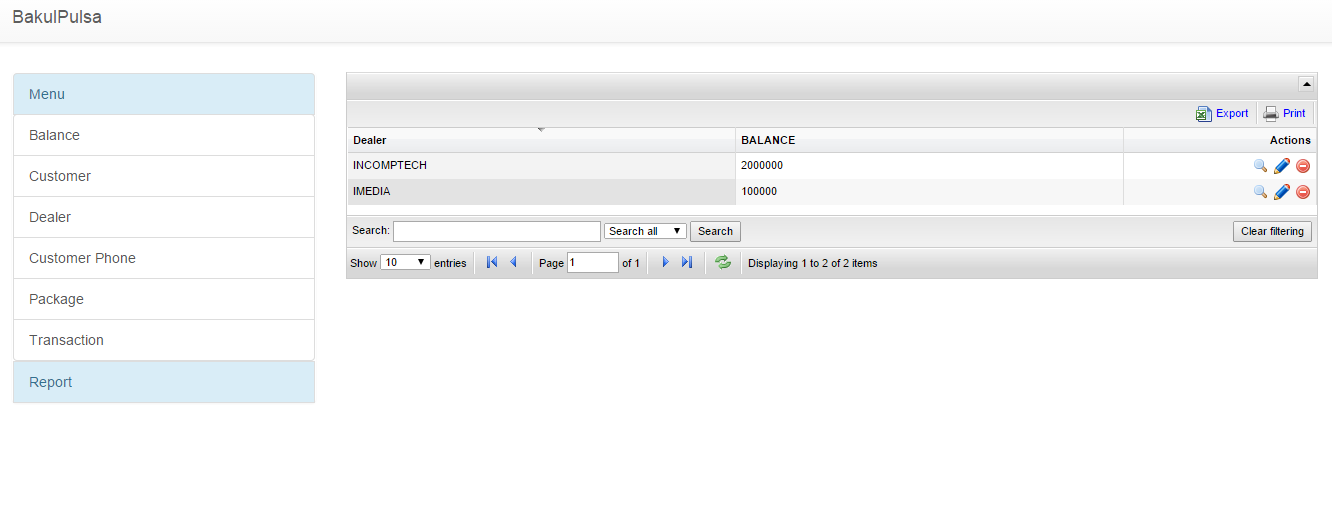
* 1. Perancangan Antarmuka Aplikasi Web
     1. Tampilan Halaman Utama



Gambar 13 Gambar tampilan Halaman Utama

Gambar 13 merupakan halaman utama dari aplikasi web. Pada halaman utama terdapat pilihan-pilihan modul yang tersedia dalam aplikasi web diantaranya, Modul Saldo, Pelanggan, Dealer Pulsa, Nomor Telepon, Paket Pengisian, Transaksi, dan Report.

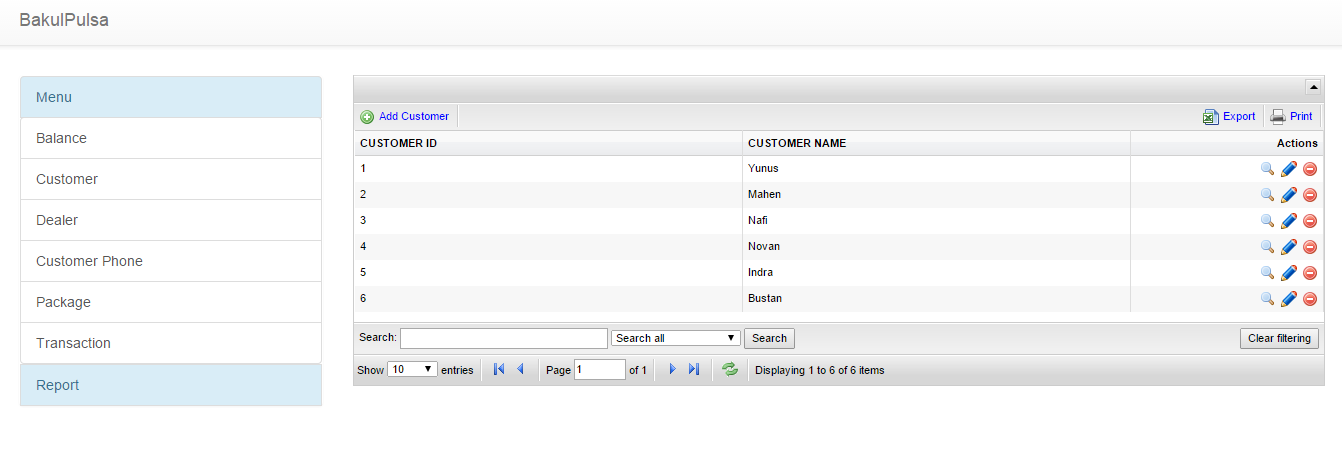
* + 1. Tampilan Halaman Manajemen Saldo



Gambar 14 Gambar tampilan halaman Manajemen Saldo

Gambar 14 merupakan halaman manajemen saldo yang berisikan sisa saldo penjual pulsa di dealer terkait, dalam halaman ini penjual dapat melakukan edit dan delete data-data yang diinginkan.

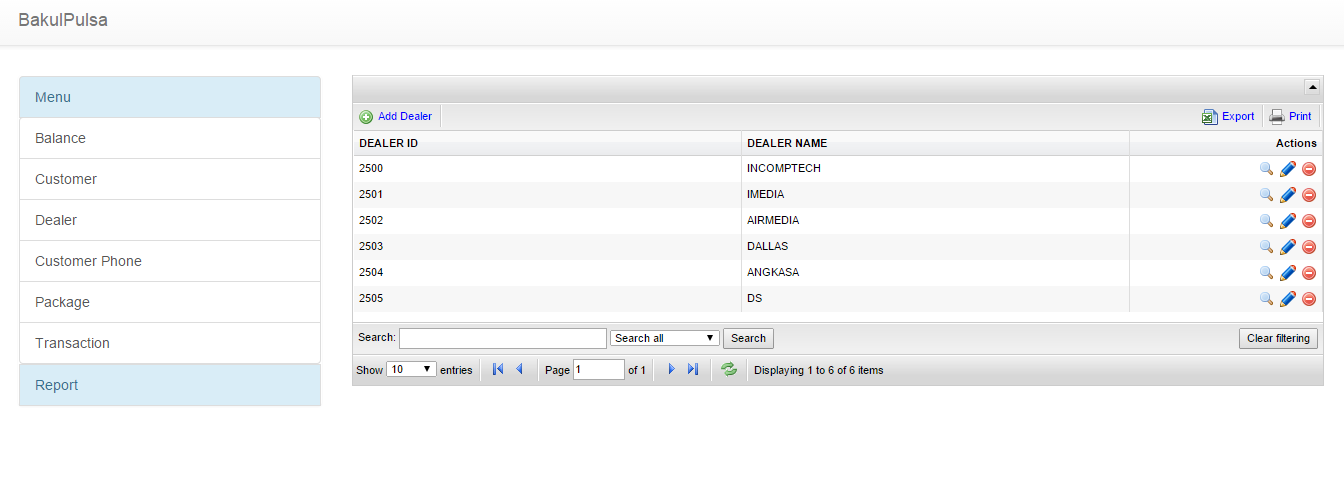
* + 1. Tampilan Halaman Manajemen Pelanggan



Gambar 15 Gambar tampilan halaman Manajemen Pelanggan

Gambar 15 merupakan halaman manajemen Pelanggan yang berisikan data pelanggan pada penjual pulsa terkait, dalam halaman ini penjual dapat melakukan tambah, edit dan delete data-data pelanggan yang diinginkan.

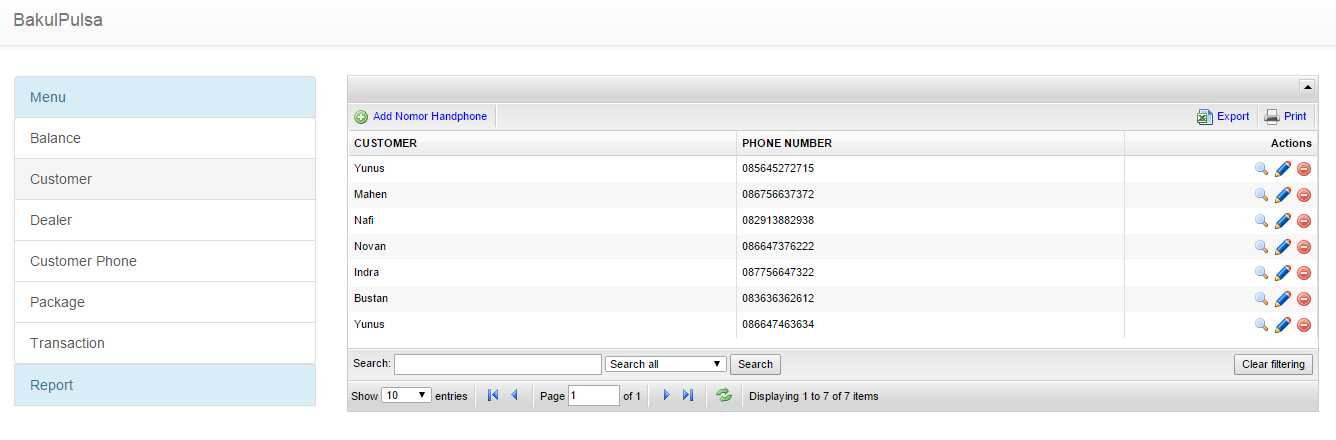
* + 1. Tampilan Halaman Manajemen Dealer Pulsa



Gambar 16 Gambar tampilan Majemen Delaer Pulsa

Gambar 16 merupakan halaman Dealer Pulsa yang berisikan data Dealer pulsa yang tersedia untuk membeli saldo, dalam halaman ini penjual dapat melakukan tambah, edit dan delete data-data dealer yang diinginkan.

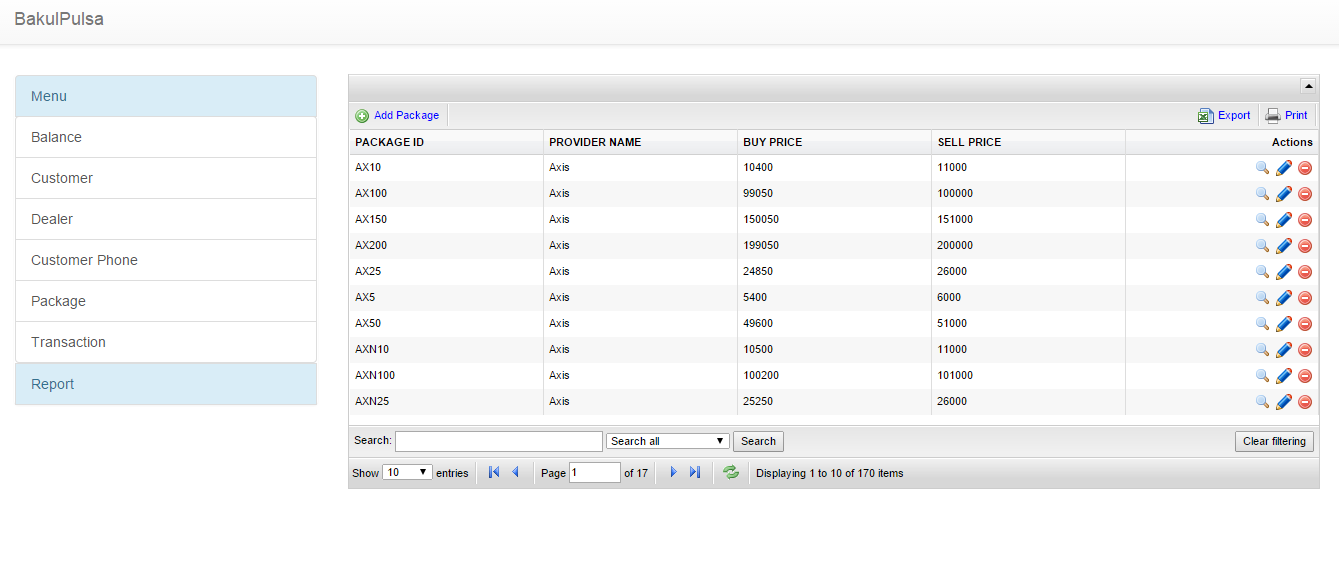
* + 1. Tampilan Halaman Manajemen Nomor Telepon



Gambar 17 Gambar tampilan Manajemen Nomor Telepon

Gambar 17 merupakan halaman manajemen Nomor telepon yang berisikan data nomor telepon pelanggan pada penjual pulsa terkait, dalam halaman ini penjual dapat melakukan tambah, edit dan delete data-data nomor telepon pelanggan yang diinginkan.

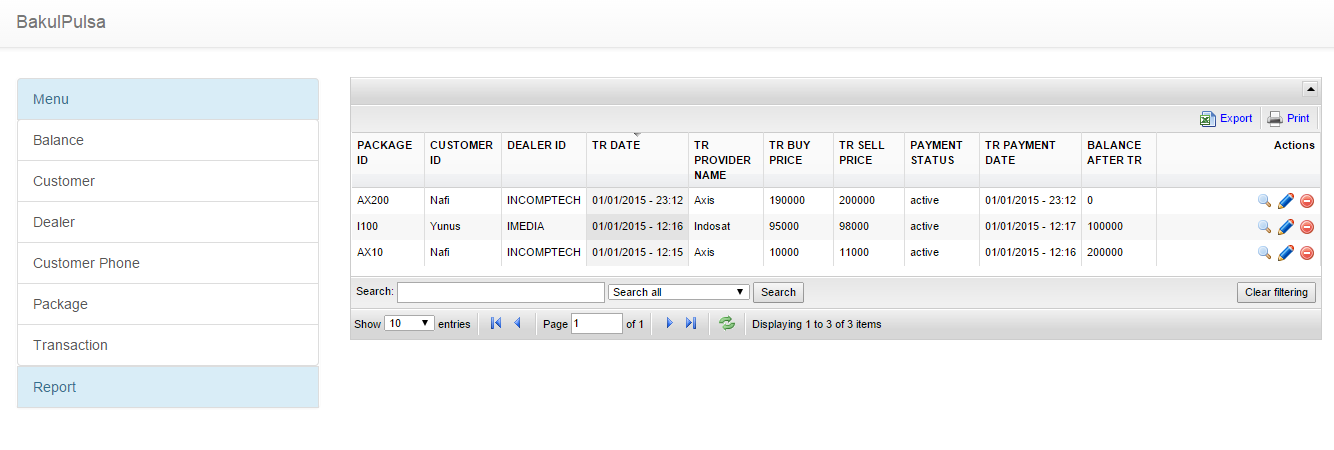
* + 1. Tampilan Halaman Manajemen Paket Pengisian



Gambar 18 Gambar tampilan halaman Manajemen Paket Pengisian

Gambar 18 merupakan halaman manajemen paket pengisian yang berisikan data paket pengisian pulsa berdasar operator penyedia layanan handphone, dalam halaman ini penjual dapat melakukan tambah, edit dan delete data-data paket yang diinginkan.

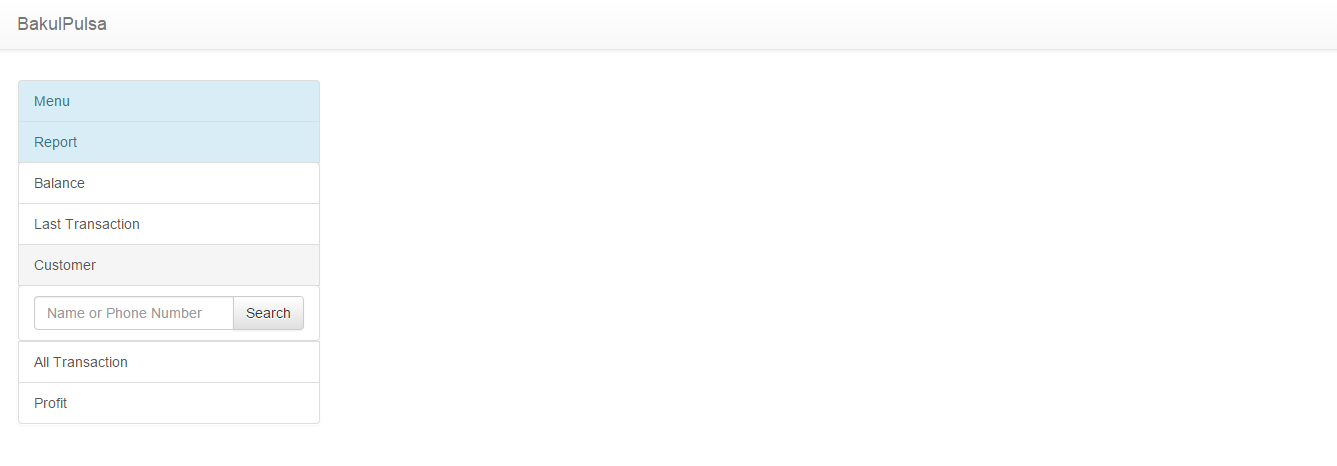
* + 1. Tampilan Halaman Manajemen Transaksi



Gambar 19 Gambar tampilan Halaman Manajemen Transaksi

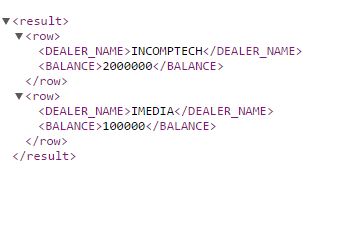
Gambar 19 merupakan halaman manajemen transaksi yang berisikan data penjualan pulsa pada penjual pulsa terkait, dalam halaman ini penjual dapat melakukan edit dan delete data-data transaksi yang diinginkan.

* + 1. Menu Reporting



Gambar 20 Gambar tampilan Menu Reporting

Gambar 20 merupakan menu report yang tersedia. Menu report untuk penggunaanya tidak tampil pada halaman utama karena menu-menu tersebut berfungsi untuk membangkitkan xml yang akan digunakan oleh android sebagai media penghubung database yang berbasis MySQL dari server ke data XML yang dapat dibaca oleh Android. Berikut contoh pembangkitan XML dari menu *Balance*



Gambar 21 Gambar contoh XML yang akan dikirim

# BAB V IMPLEMENTASI

Pada pengimplementasian, kelompok kami tidak diwajibkan menyelesaikan program kami hingga 100%. Aplikasi kami telah selesai 80%, terdiri dari semua Interface dan Fungsional pada XML. Kekurangan yang belum direalisasikan adalah pada bagian penghubung antara fungsionalitas XML dan Interface yang ada.

Tugas ini dikerjakan pada lingkungan kerja dengan rincian sebagai berikut:

* Sistem Operasi Windows 8.1 64 bit
* Processor Intel(R) Core(TM) i3
* RAM 2.00 GB

Untuk perangkat lunak / aplikasi pendukung yang kami gunakan

dalam pembuatan program serta laporan kegiatan adalah sebagai berikut:

* Android SDK
* Eclipse
* StarUML
* Microsoft Office Word 2013
  1. Pembuatan User Interface Aplikasi
     1. Kelas AboutFragment.java

Kelas ini berfungsi untuk memanggil layout yang berisi halaman “about”. Potongan kode kelas ini seperti pada gambar 22. Kelas ini menggunakan fungsi inflater.inflate untuk memanggil fragment halaman dari layout yang telah dibuat.

Gambar 22 Gambar potongan kode sumber Class AboutFragment.java

package com.kape.bakulpulsa;

import android.app.Fragment;

import android.os.Bundle;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

public class AboutFragment extends Fragment {

public AboutFragment(){}

@Override

public View onCreateView(LayoutInflater inflater, ViewGroup container,

Bundle savedInstanceState) {

View rootView = inflater.inflate(R.layout.fragment\_about, container, false);

return rootView;

}

}

* + 1. Kelas CustomerFragmentAdd.java

Kelas ini berfungsi untuk memanggil layout yang berisi halaman untuk menambahkan pelanggan. Potongan kode kelas ini dapat di lihat pada gambar 23. Kelas ini menggunakan fungsi inflater.inflate untuk memanggil fragment halaman dari layout.

Gambar 23 Gambar potongan kode sumber Class CustomerFragment.java

package com.kape.bakulpulsa;

import android.app.Fragment;

import android.os.Bundle;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

public class CustomerFragmentAdd extends Fragment {

public CustomerFragmentAdd(){}

@Override

public View onCreateView(LayoutInflater inflater, ViewGroup container,

Bundle savedInstanceState) {

View rootView = inflater.inflate(R.layout.fragment\_customer\_add, container, false);

return rootView;

}

}

* + 1. Kelas CustomerFragmentShow.java

Kelas ini berfungsi untuk memanggil layout yang berisi halaman untuk menampilkan pelanggan. Potongan kode kelas ini dapat di lihat pada gambar 24. Kelas ini menggunakan fungsi inflater.inflate untuk memanggil fragment halaman dari layout.

Gambar 24 Gambar potongan kode sumber Class CustomerFragmentShow.java

package com.kape.bakulpulsa;

import android.app.Fragment;

import android.os.Bundle;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

public class CustomerFragmentShow extends Fragment {

public CustomerFragmentShow(){}

@Override

public View onCreateView(LayoutInflater inflater, ViewGroup container,

Bundle savedInstanceState) {

View rootView = inflater.inflate(R.layout.fragment\_customer\_show, container, false);

return rootView;

}

}

* + 1. Kelas HistoryFragment.java

Kelas ini berfungsi untuk memanggil layout yang berisi halaman untuk menampilkan history dari transaksi penjualan. Potongan kode kelas ini dapat di lihat pada gambar 25. Kelas ini menggunakan fungsi inflater.inflate untuk memanggil fragment halaman dari layout.

package com.kape.bakulpulsa;

import android.app.Fragment;

import android.os.Bundle;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

public class HistoryFragment extends Fragment {

public HistoryFragment(){}

@Override

public View onCreateView(LayoutInflater inflater, ViewGroup container,

Bundle savedInstanceState) {

View rootView = inflater.inflate(R.layout.fragment\_history, container, false);

return rootView;

}

}

Gambar 25 Gambar potongan kode sumber Class HistoryFragment.java

* + 1. Kelas HomeFragment.java

Kelas ini berfungsi untuk memanggil layout yang berisi halaman untuk menampilkan halaman awal aplikasi. Potongan kode kelas ini dapat di lihat pada gambar 26. Kelas ini menggunakan fungsi inflater.inflate untuk memanggil fragment halaman dari layout.

Gambar 26 Gambar potongan kode sumber Class HomeFragment.java

package com.kape.bakulpulsa;

import android.app.Fragment;

import android.os.Bundle;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

public class HomeFragment extends Fragment {

public HomeFragment(){}

@Override

public View onCreateView(LayoutInflater inflater, ViewGroup container,

Bundle savedInstanceState) {

View rootView = inflater.inflate(R.layout.fragment\_home, container, false);

return rootView;

}

}

* + 1. Kelas SaldoFragment.java

Kelas ini berfungsi untuk memanggil layout yang berisi halaman untuk menampilkan halaman informasi saldo. Potongan kode kelas ini dapat di lihat pada gambar 27. Kelas ini menggunakan fungsi inflater.inflate untuk memanggil fragment halaman dari layout.

Gambar 27 Gambar potongan kode sumber Class SaldoFragment.java

package com.kape.bakulpulsa;

import android.app.Fragment;

import android.os.Bundle;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

public class SaldoFragment extends Fragment {

public SaldoFragment(){}

@Override

public View onCreateView(LayoutInflater inflater, ViewGroup container,

Bundle savedInstanceState) {

View rootView = inflater.inflate(R.layout.fragment\_saldo, container, false);

return rootView;

}

}

* + 1. Kelas TransactionFragment.java

Kelas ini berfungsi untuk memanggil layout yang berisi halaman pencatat transaksi pembelian. Potongan kode kelas ini dapat di lihat pada gambar 28. Kelas ini menggunakan fungsi inflater.inflate untuk memanggil fragment halaman dari layout.

Gambar 28 Gambar potongan kode sumber Class TransactionFragment.java

package com.kape.bakulpulsa;

import android.app.Fragment;

import android.os.Bundle;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

public class TransactionFragment extends Fragment {

public TransactionFragment(){}

@Override

public View onCreateView(LayoutInflater inflater, ViewGroup container,

Bundle savedInstanceState) {

View rootView = inflater.inflate(R.layout.fragment\_transaction, container, false);

return rootView;

}

}

* + 1. Kelas NavDrawerListAdapter.java

Kelas ini berfungsi mengatur item-item yang ada pada drawer list. Potongan kode kelas ini dapat di lihat pada gambar 29.

package info.androidhive.slidingmenu.adapter;

import com.kape.bakulpulsa.R;

public class NavDrawerListAdapter extends BaseAdapter {

private Context context;

private ArrayList<NavDrawerItem> navDrawerItems;

public NavDrawerListAdapter(Context context, ArrayList<NavDrawerItem> navDrawerItems){

this.context = context;

this.navDrawerItems = navDrawerItems;

}

@Override

public int getCount() {

return navDrawerItems.size();

}

@Override

public Object getItem(int position) {

return navDrawerItems.get(position);

}

@Override

public long getItemId(int position) {

return position;

}

Gambar 29 Gambar potongan kode sumber Class NavDrawerListAdapter.java

@Override

public View getView(int position, View convertView, ViewGroup parent) {

if (convertView == null) {

LayoutInflater mInflater = (LayoutInflater)

context.getSystemService(Activity.*LAYOUT\_INFLATER\_SERVICE*);

convertView = mInflater.inflate(R.layout.*drawer\_list\_item*, parent , false);

}

ImageView imgIcon = (ImageView) convertView.findViewById(R.id.*icon*);

TextView txtTitle = (TextView) convertView.findViewById(R.id.*title*);

TextView txtCount = (TextView) convertView.findViewById(R.id.*counter*);

imgIcon.setImageResource(navDrawerItems.get(position).getIcon());

txtTitle.setText(navDrawerItems.get(position).getTitle());

// displaying count

// check whether it set visible or not

if(navDrawerItems.get(position).getCounterVisibility()){

txtCount.setText(navDrawerItems.get(position).getCount());

}else{

// hide the counter view

txtCount.setVisibility(View.*GONE*);

}

return convertView;

}

}

* + 1. Kelas NavDrawerItem.java

Kelas ini berfungsi membuat item-item yang ada pada drawer list. Potongan kode kelas ini dapat di lihat pada gambar 30.

public NavDrawerItem(String title, int icon, boolean isCounterVisible, String count){

this.title = title;

this.icon = icon;

this.isCounterVisible = isCounterVisible;

this.count = count;

}

public String getTitle(){

return this.title;

}

public int getIcon(){

return this.icon;

}

public String getCount(){

return this.count;

}

public boolean getCounterVisibility(){

return this.isCounterVisible;

}

public void setTitle(String title){

this.title = title;

}

public void setIcon(int icon){

this.icon = icon;

}

public void setCount(String count){

this.count = count;

}

public void setCounterVisibility(boolean isCounterVisible){

this.isCounterVisible = isCounterVisible;

}

}

Gambar 30 Gambar potongan kode sumber Class NavDrawerItem.java

* + 1. Kelas MainActivity.java

Kelas ini berfungsi kelas utama dari semua aktifitas aplikasi. Potongan kode kelas ini dapat di lihat pada gambar 31.

package com.kape.bakulpulsa;

import info.androidhive.slidingmenu.adapter.NavDrawerListAdapter;

import info.androidhive.slidingmenu.model.NavDrawerItem;

…

public class MainActivity extends Activity {

private DrawerLayout mDrawerLayout;

private ListView mDrawerList;

private ActionBarDrawerToggle mDrawerToggle;

// nav drawer title

private CharSequence mDrawerTitle;

// used to store app title

private CharSequence mTitle;

// slide menu items

private String[] navMenuTitles;

private TypedArray navMenuIcons;

private ArrayList<NavDrawerItem> navDrawerItems;

private NavDrawerListAdapter adapter;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

navMenuTitles = getResources().getStringArray(R.array.nav\_drawer\_items);

// nav drawer icons from resources

navMenuIcons = getResources()

.obtainTypedArray(R.array.*nav\_drawer\_icons*);

mDrawerLayout = (DrawerLayout) findViewById(R.id.*drawer\_layout*);

mDrawerList = (ListView) findViewById(R.id.*list\_slidermenu*);

navDrawerItems = **new** ArrayList<NavDrawerItem>();

// adding nav drawer items to array

// Home

navDrawerItems.add(**new** NavDrawerItem(navMenuTitles[0], navMenuIcons.getResourceId(0, -1)));

// Find People

navDrawerItems.add(**new** NavDrawerItem(navMenuTitles[1], navMenuIcons.getResourceId(1, -1)));

// Photos

navDrawerItems.add(**new** NavDrawerItem(navMenuTitles[2], navMenuIcons.getResourceId(2, -1)));

// Communities, Will add a counter here

navDrawerItems.add(**new** NavDrawerItem(navMenuTitles[3], navMenuIcons.getResourceId(3, -1), **false**, "22"));

// Pages

navDrawerItems.add(**new** NavDrawerItem(navMenuTitles[4], navMenuIcons.getResourceId(4, -1)));

// What's hot, We will add a counter here

navDrawerItems.add(**new** NavDrawerItem(navMenuTitles[5], navMenuIcons.getResourceId(5, -1), **false**, "50+"));

navDrawerItems.add(**new** NavDrawerItem(navMenuTitles[6], navMenuIcons.getResourceId(6, -1)));

// Recycle the typed array

navMenuIcons.recycle();

mDrawerList.setOnItemClickListener(**new** SlideMenuClickListener());

// setting the nav drawer list adapter

adapter = **new** NavDrawerListAdapter(getApplicationContext(),

navDrawerItems);

mDrawerList.setAdapter(adapter);

// enabling action bar app icon and behaving it as toggle button

getActionBar().setDisplayHomeAsUpEnabled(**true**);

getActionBar().setHomeButtonEnabled(**true**);

mDrawerToggle = **new** ActionBarDrawerToggle(**this**, mDrawerLayout,

R.drawable.*ic\_drawer*, //nav menu toggle icon

R.string.*app\_name*, // nav drawer open - description for accessibility

R.string.*app\_name* // nav drawer close - description for accessibility

) {

**public** **void** onDrawerClosed(View view) {

getActionBar().setTitle(mTitle);

// calling onPrepareOptionsMenu() to show action bar icons

invalidateOptionsMenu();

}

**public** **void** onDrawerOpened(View drawerView) {

getActionBar().setTitle(mDrawerTitle);

// calling onPrepareOptionsMenu() to hide action bar icons

invalidateOptionsMenu();

}

};

mDrawerLayout.setDrawerListener(mDrawerToggle);

**if** (savedInstanceState == **null**) {

// on first time display view for first nav item

displayView(0);

}

}

/\*\*

\* Slide menu item click listener

\* \*/

**private** **class** SlideMenuClickListener **implements**

ListView.OnItemClickListener {

@Override

**public** **void** onItemClick(AdapterView<?> parent, View view, **int** position,

**long** id) {

// display view for selected nav drawer item

displayView(position);

// enabling action bar app icon and behaving it as toggle button

getActionBar().setDisplayHomeAsUpEnabled(**true**);

getActionBar().setHomeButtonEnabled(**true**);

mDrawerToggle = **new** ActionBarDrawerToggle(**this**, mDrawerLayout,

R.drawable.*ic\_drawer*, //nav menu toggle icon

R.string.*app\_name*, // nav drawer open - description for accessibility

R.string.*app\_name* // nav drawer close - description for accessibility

) {

**public** **void** onDrawerClosed(View view) {

getActionBar().setTitle(mTitle);

// calling onPrepareOptionsMenu() to show action bar icons

invalidateOptionsMenu();

}

**public** **void** onDrawerOpened(View drawerView) {

getActionBar().setTitle(mDrawerTitle);

// calling onPrepareOptionsMenu() to hide action bar icons

invalidateOptionsMenu();

}

};

mDrawerLayout.setDrawerListener(mDrawerToggle);

**if** (savedInstanceState == **null**) {

displayView(0);

}

}

**private** **class** SlideMenuClickListener **implements**

ListView.OnItemClickListener {

@Override

**public** **void** onItemClick(AdapterView<?> parent, View view, **int** position,

**long** id) {

displayView(position);}

}

@Override

public boolean onCreateOptionsMenu(Menu menu) {

getMenuInflater().inflate(R.menu.*main*, menu);

return true;

}

@Override

public boolean onOptionsItemSelected(MenuItem item) {

if (mDrawerToggle.onOptionsItemSelected(item)) {

return true;

}

switch (item.getItemId()) {

case R.id.*action\_settings*:

return true;

default:

return super.onOptionsItemSelected(item);

}

}

@Override

public boolean onPrepareOptionsMenu(Menu menu) {

boolean drawerOpen = mDrawerLayout.isDrawerOpen(mDrawerList);

menu.findItem(R.id.*action\_settings*).setVisible(!drawerOpen);

return super.onPrepareOptionsMenu(menu);

}

private void displayView(int position) {

Fragment fragment = null;

switch (position) {

case 0:

fragment = new HomeFragment();

break;

case 1:

fragment = new SaldoFragment();

break;

case 2:

fragment = new CustomerFragmentAdd();

break;

case 3:

fragment = new CustomerFragmentShow();

break;

case 4:

fragment = new TransactionFragment();

break;

case 5:

fragment = new HistoryFragment();

break;

case 6:

fragment = new AboutFragment();

break;

default:

break;

}

Gambar 31 Gambar potongan kode sumber Class MainActivity.java

default:

break;

}

if (fragment != null) {

FragmentManager fragmentManager = getFragmentManager();

fragmentManager.beginTransaction()

.replace(R.id.*frame\_container*, fragment).commit();

mDrawerList.setItemChecked(position, true);

mDrawerList.setSelection(position);

setTitle(navMenuTitles[position]);

mDrawerLayout.closeDrawer(mDrawerList);

} else {

Log.*e*("MainActivity", "Error in creating fragment");

}

}

@Override

public void setTitle(CharSequence title) {

mTitle = title;

getActionBar().setTitle(mTitle);

}

@Override

protected void onPostCreate(Bundle savedInstanceState) {

super.onPostCreate(savedInstanceState);

// Sync the toggle state after onRestoreInstanceState has occurred.

mDrawerToggle.syncState();

}

@Override

public void onConfigurationChanged(Configuration newConfig) {

super.onConfigurationChanged(newConfig);

// Pass any configuration change to the drawer toggls

mDrawerToggle.onConfigurationChanged(newConfig);

}

}

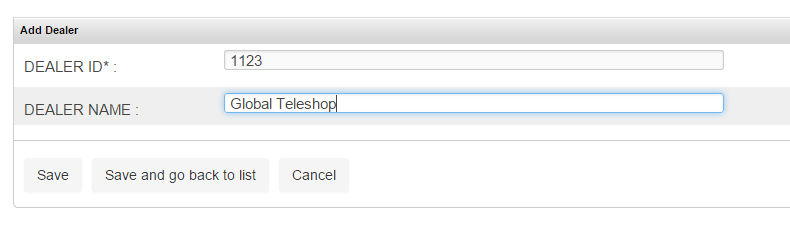
# BAB VI UJI COBA DAN EVALUASI

Pada bab ini akan dijelaskan tentang proses ujicoba fungsi–fungsi aplikasi BakulPulsa. Selain itu akan diuraikan juga hasil evaluasi fungsi terserbut.

* 1. Lingkungan Uji Coba

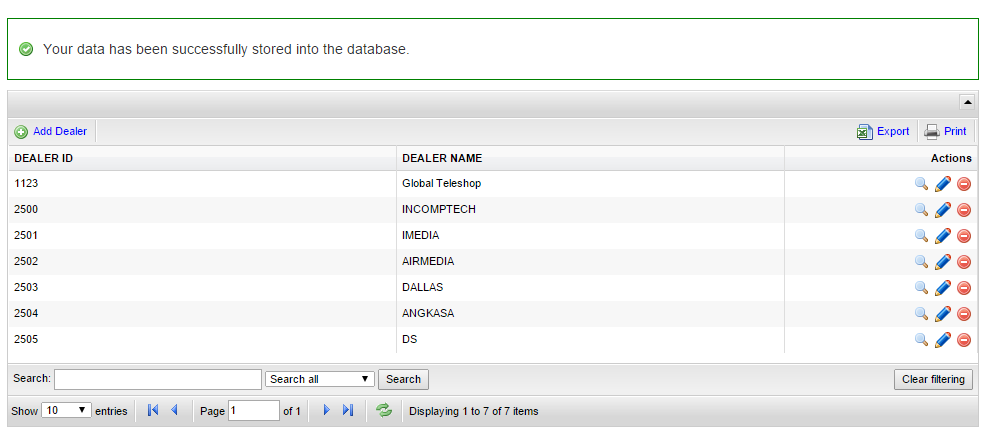
Lingkungan uji coba yang dilakukan untuk menguji fungsi–fungsi aplikasi adalah komputer dengan prosesor Intel(R) Core(TM) i3 2310M @2.10GHz(2 CPUs) ~2.1GHz, Memory 4096MB, menggunakan Operasi Windows 8 Pro 64-bit. Perangkat pengembang yang digunakan adalah Eclipse dan Netbeans. Selain itu, perangkat yang digunakan untuk menguji aplikasi perangkat bergerak adalah Android Emulator. Browser yang digunakan untuk menguji fungsi aplikasi web adalah Chrome.

* 1. Membuat Dealer Baru



Gambar 32 Gambar tampilan membuatan Dealer Baru

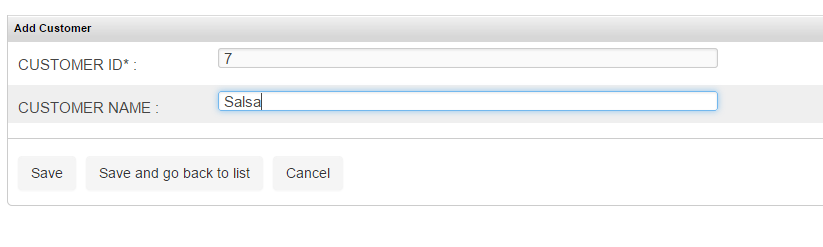
Tampilan halaman diatas adalah untuk membuat dealer baru. Setelah selesai mengisi data-data yang diperlukan pengguna menekan tombol *save* atau *save and go back to list* untuk menyimpan data tersebut ke database, atau pengguna dapat menekan tombol *cancel* untuk membatalkan.



Gambar 33 Gambar tampilan notifikasi perubahan database dealer baru

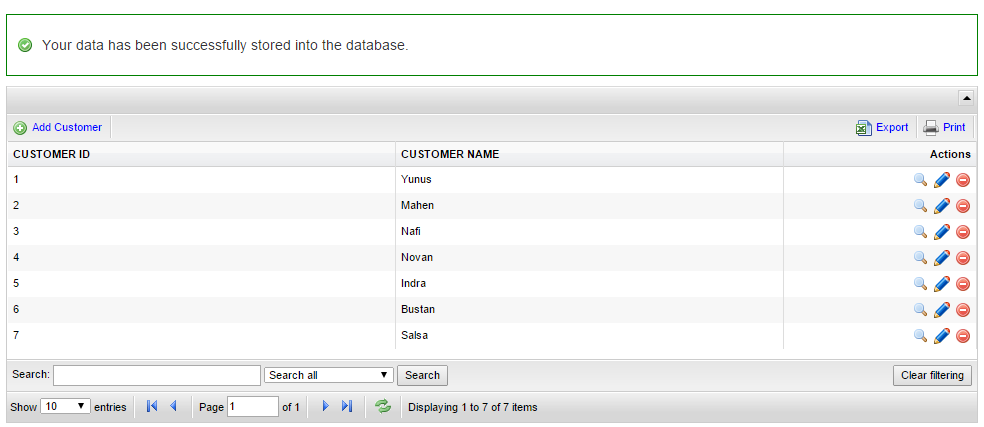
Setelah data sukses disimpan akan muncul tampilan halaman diatas yang berisikan notifikasi bahwa database telah diperbarui dengan data yang baru dimasukkan. dan data yang sudah dimasukkan akan muncul pada tabel tersebut.

* 1. Membuat Pelanggan Baru



Gambar 34 Gambar membuat pelanggan baru

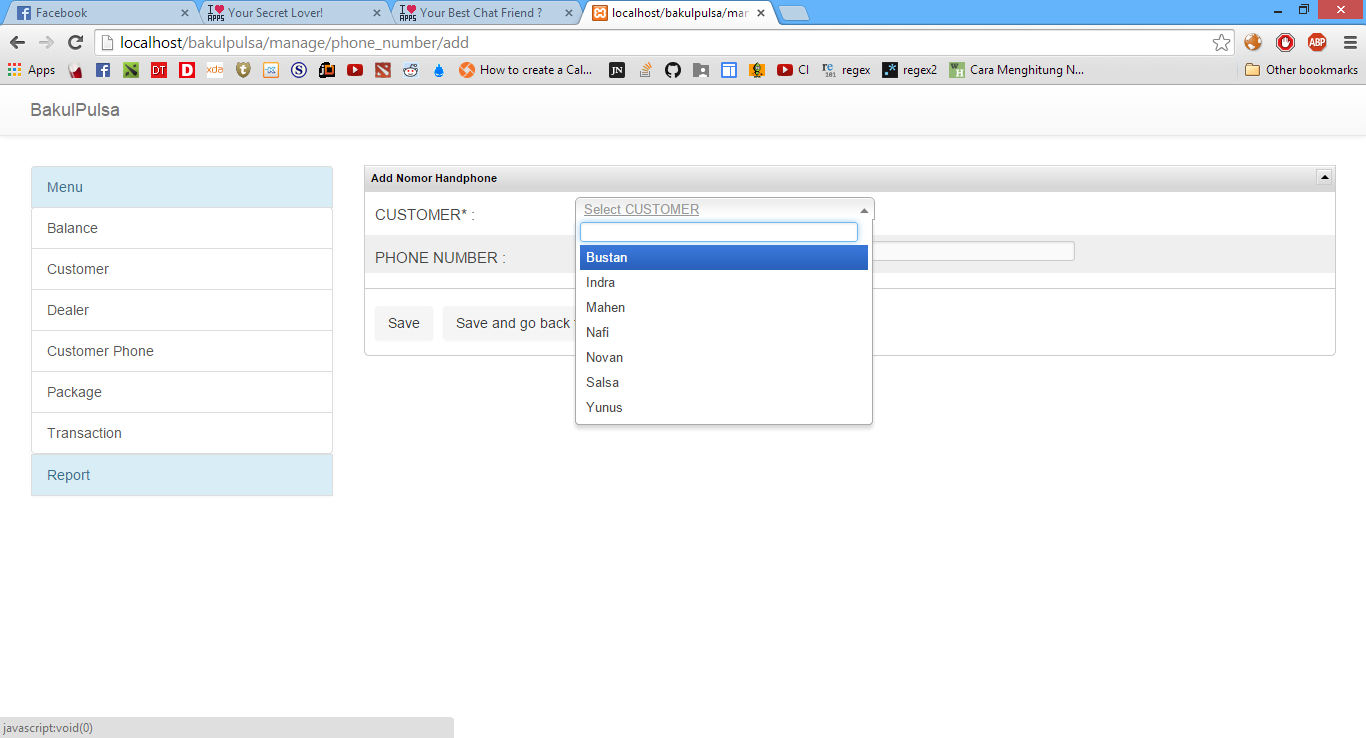
Tampilan halaman diatas adalah untuk membuat pelanggan baru. Setelah selesai mengisi data-data yang diperlukan pengguna menekan tombol *save* atau *save and go back to list* untuk menyimpan data tersebut ke database, atau pengguna dapat menekan tombol *cancel* untuk membatalkan.



Gambar 35 Gambar tampilan notifikasi perubahan database pelanggan

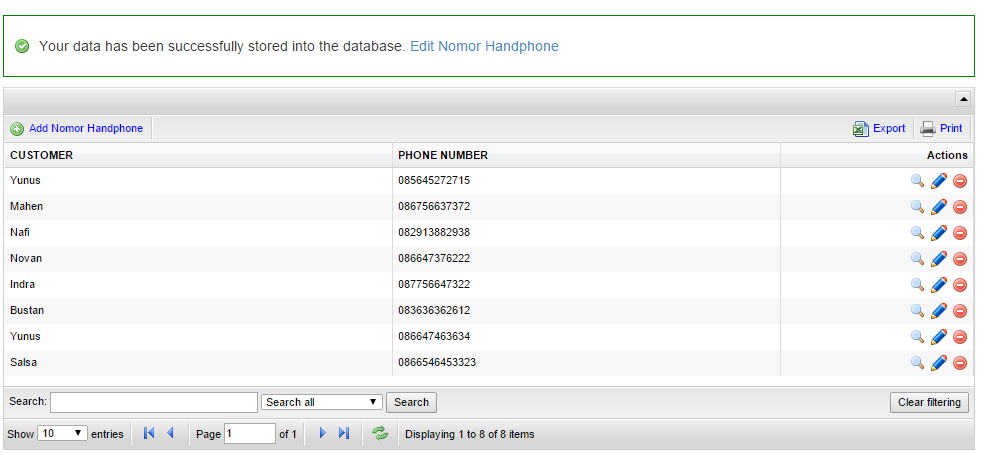
Setelah data sukses disimpan akan muncul tampilan halaman diatas yang berisikan notifikasi bahwa database telah diperbarui dengan data yang baru dimasukkan. dan data yang sudah dimasukkan akan muncul pada tabel tersebut.

* 1. Menambah Nomor pelanggan



Gambar 36 Gambar tampilan menambahkan nomor pelanggan

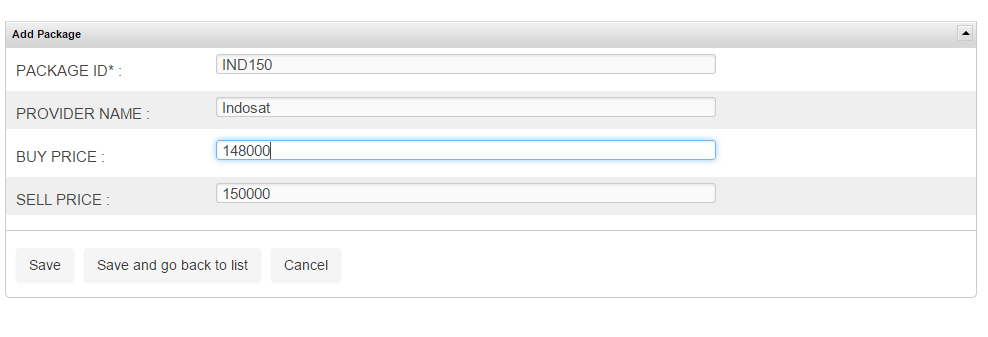
Tampilan halaman diatas adalah untuk membuat nomor telepon pelanggan baru. Setelah selesai mengisi data-data yang diperlukan pengguna menekan tombol *save* atau *save and go back to list* untuk menyimpan data tersebut ke database, atau pengguna dapat menekan tombol *cancel* untuk membatalkan.



Gambar 37 Gambar tampilan notifikasi perubahan database nomor pelanggan

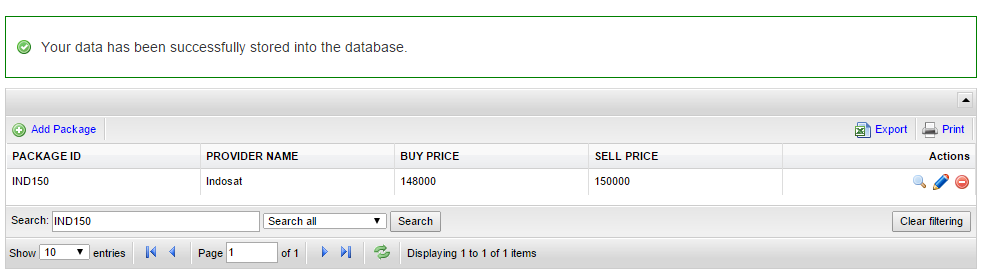
Setelah data sukses disimpan akan muncul tampilan halaman diatas yang berisikan notifikasi bahwa database telah diperbarui dengan data yang baru dimasukkan. dan data yang sudah dimasukkan akan muncul pada tabel tersebut.

* 1. Menambah Paket Pengisian



Gambar 38 Gambar tampilan penambahan paket pengisian

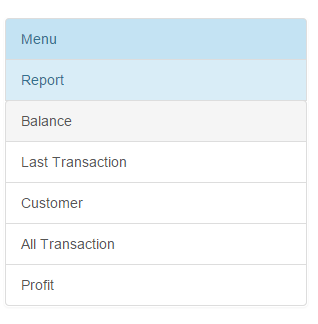
Tampilan halaman diatas adalah untuk membuat paket penjualan baru. Setelah selesai mengisi data-data yang diperlukan pengguna menekan tombol *save* atau *save and go back to list* untuk menyimpan data tersebut ke database, atau pengguna dapat menekan tombol *cancel* untuk membatalkan.



Gambar 39 Gambar tampilan notifikasi perubahan database paket pengisian

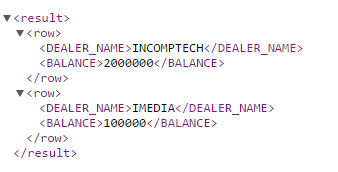
Setelah data sukses disimpan akan muncul tampilan halaman diatas yang berisikan notifikasi bahwa database telah diperbarui dengan data yang baru dimasukkan. dan data yang sudah dimasukkan akan muncul pada tabel tersebut.

* 1. Membangkitkan XML Saldo



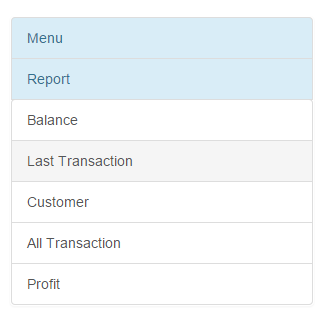
Gambar 40 Gambar tampilan menu pada web

Untuk menampilkan data XML saldo di web klik *report* kemudian *balance*. Untuk mendapatkan data saldo pada aplikasi perangkat bergerak maka aplikasi tersebut perlu melakukan permintaan halaman yang beralamat di *http://[domain]/bakulpulsa/request/saldo/*. Maka aplikasi web akan mengembalikan data saldo yang diminta berupa XML seperti berikut



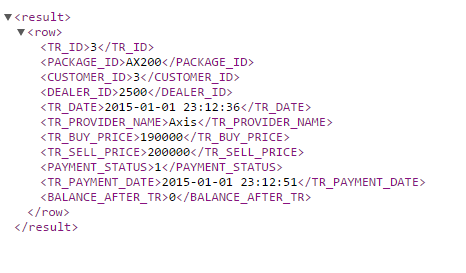
Gambar 41 Gambar XML dari menu balance yang dikirim

* 1. Membangkitkan XML Transaksi Terakhir



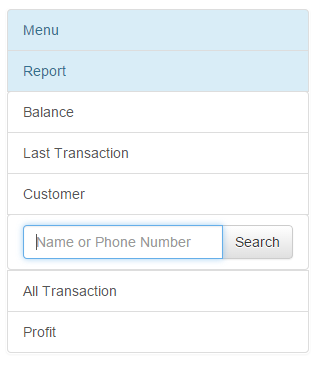
Gambar 42 Tampilan Menu Report di Web

Untuk menampilkan data XML transaksi terakhir di web klik *report* kemudian *last transaction*. Untuk mendapatkan data transaksi terakhir pada aplikasi perangkat bergerak maka aplikasi tersebut perlu melakukan permintaan halaman yang beralamat di *http://[domain]/bakulpulsa/request/last\_tr/*. Maka aplikasi web akan mengembalikan data saldo yang diminta berupa XML seperti berikut



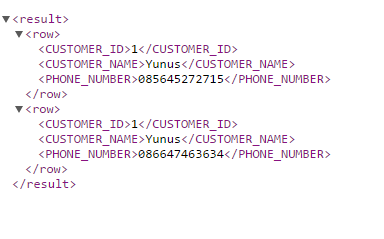
Gambar 43 Hasil Pembangkitan XML Transaksi Terakhir

* 1. Membangkitkan XML Pencarian Pelanggan



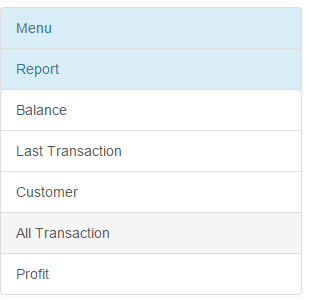
Gambar 44 Menu Report Pencarian Pelanggan

Untuk mencari data pelanggan di web klik *report* kemudian *customer*  kemudian isikan parameter. Untuk mendapatkan data saldo pada aplikasi perangkat bergerak maka aplikasi tersebut perlu melakukan permintaan halaman yang beralamat di *http://[domain]/bakulpulsa/request/pelanggan/[parameter]*. Maka aplikasi web akan mengembalikan data saldo yang diminta berupa XML seperti berikut.



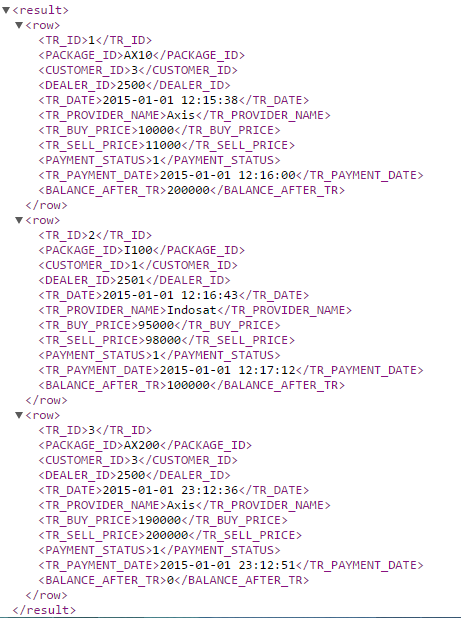
Gambar 45 Hasil Pembangkitan XML Pencarian Pelanggan

* 1. Membangkitkan XML Seluruh Transaksi



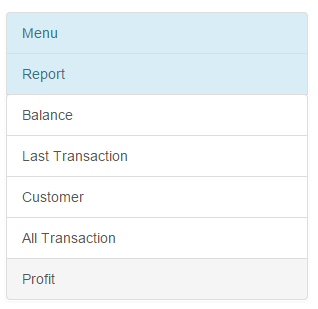
Gambar 46 Menu Report Seluruh Transaksi

Untuk mencari data pelanggan di web klik *report* kemudian *All Transaction*. Untuk mendapatkan data saldo pada aplikasi perangkat bergerak maka aplikasi tersebut perlu melakukan permintaan halaman yang beralamat di *http://[domain]/bakulpulsa/request/all\_tr/*. Maka aplikasi web akan mengembalikan data saldo yang diminta berupa XML seperti berikut.



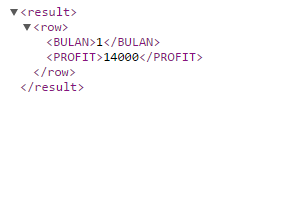
Gambar 47 Hasil Pembangkitan XML Seluruh Transaksi

* 1. Membangkitkan XML Keuntungan Per Bulan



Gambar 48 Menu Report Keuntungan Perbulan

Untuk mencari data pelanggan di web klik *report* kemudian *Profit*. Untuk mendapatkan data saldo pada aplikasi perangkat bergerak maka aplikasi tersebut perlu melakukan permintaan halaman yang beralamat di *http://[domain]/bakulpulsa/request/profit/*. Maka aplikasi web akan mengembalikan data saldo yang diminta berupa XML seperti berikut.



Gambar 49 Hasil Pembangkitan XML Keuntungan Perbulan

# BAB VII PENUTUP

## 7.1 Kesimpulan

Aplikasi “BakulPulsa” ini berguna bagi penjual pulsa pada tingkat pengecer. Mudah digunakan untuk melakukan perekapan transaksi dan pelanggan yang membeli. Dengan laporan yang diletakakn pada web, memudahkan pemakai aplikasi “BakulPulsa” untuk mengecek laporan penjualan pulsa.

Pengerjaan aplikasi “BakulPulsa” menggunakan android yang sesuai dengan smartphone yang semakin marak akhir-akhir ini. Diharapkan dengan adanya aplikasi “BakulPulsa” ini dapat mempermudah dan semakin melancarkan bisnis para penjual pulsa tingkat *eceran*.

## 7.2 Saran

Diharapkan pengembangan aplikasi “BakulPulsa”ini tidak berhenti sampai disini saja. Semoga pada waktu mendatang dikembangkan menjadi aplikasi yang dapat dikomersilkan dan dapat digunakan oleh banyak orang.

# DAFTAR PUSTAKA

|  |  |
| --- | --- |
| [1] | PT. AirMedia Persada, "About Us," [Online]. Available: http://www.airmediapersada.com/profil-perusahaan.html. [Accessed 21 Maret 2015]. |
| [2] | Learn Data Modeling, "Conceptual Data Model," [Online]. Available: http://www.learndatamodeling.com/cdm.php. [Accessed 22 Maret 2015]. |
| [3] | Learn Data Modeling, "Physical Data Model," [Online]. Available: http://www.learndatamodeling.com/pdm.php. [Accessed 22 Maret 2015]. |
| [4] | CodeIgniter, "Model View Controller," [Online]. Available: https://ellislab.com/codeigniter/user-guide/overview/mvc.html. [Accessed 23 Maret 2015]. |
| [5] | Android, "About Android," [Online]. Available: http://developer.android.com/about/index.html. [Accessed 15 Maret 2015]. |
| [6] | Wikipedia, "Android(Operating System)," [Online]. Available: http://en.wikipedia.org/wiki/Android\_%28operating\_system%29. [Accessed 21 Maret 2015]. |
| [7] | Apache, "Apache," [Online]. Available: http://httpd.apache.org/. [Accessed 25 Maret 2015]. |
| [8] | PHP, "PHP," [Online]. Available: http://php.net/. [Accessed 22 Maret 2015]. |
| [9] | Mozilla Developer Network, "HTML," [Online]. Available: https://developer.mozilla.org/en-US/docs/Web/HTML. [Accessed 22 Maret 2015]. |
| [10] | Mozilla Developer Network, "CSS," [Online]. Available: https://developer.mozilla.org/en-US/docs/Web/CSS. [Accessed 22 Maret 2015]. |

# LAMPIRAN

Kode sumber lengkap untuk implementasi aplikasi web.

* Controller manajemen data

|  |
| --- |
| <?php if (!defined('BASEPATH'))  exit('No direct script access allowed');  class Manage extends CI\_Controller {  public function \_\_construct() {  parent::\_\_construct();  $this->load->database();  $this->load->helper('url');  $this->load->library('grocery\_CRUD');  }  public function output($output = null) {  $this->load->view('v\_manage.php', $output);  }  public function index() {  $this->output((object) array('output' => '',  'js\_files' => array(base\_url() . 'assets/js/jquery.min.js'),  'css\_files' => array()));  }  public function balance() {  $out = new Grocery\_CRUD();  $out->set\_table('balance')  ->unset\_add()  ->set\_subject('Balance')  ->required\_fields('DEALER\_ID')  ->set\_relation('DEALER\_ID', 'dealer', 'DEALER\_NAME')  ->display\_as('DEALER\_ID', 'Dealer');  $res = $out->render();  $this->output($res);  }  public function customer() {  $out = new Grocery\_CRUD();  $out->set\_table('customer')  ->set\_subject('Customer')  ->required\_fields('CUSTOMER\_ID');  $res = $out->render();  $this->output($res);  }  public function dealer() {  $out = new Grocery\_CRUD();  $out->set\_table('dealer')  ->set\_subject('Dealer')  ->required\_fields('DEALER\_ID');  $res = $out->render();  $this->output($res);  }  public function phone\_number() {  $out = new Grocery\_CRUD();  $out->set\_table('phone\_number')  ->set\_subject('Nomor Handphone')  ->required\_fields('CUSTOMER\_ID')  ->set\_relation('CUSTOMER\_ID', 'customer', 'CUSTOMER\_NAME')  ->display\_as('CUSTOMER\_ID', 'CUSTOMER');  $res = $out->render();  $this->output($res);  }  public function refill\_package() {  $out = new Grocery\_CRUD();  $out->set\_table('refill\_package')  ->set\_subject('Package')  ->required\_fields('PACKAGE\_ID');  $res = $out->render();  $this->output($res);  }  public function transaction\_details() {  $out = new Grocery\_CRUD();  $out->set\_table('transaction\_details')  ->unset\_add()  ->set\_subject('Transaksi')  ->set\_relation('CUSTOMER\_ID', 'customer', 'CUSTOMER\_NAME')  ->set\_relation('DEALER\_ID', 'dealer', 'DEALER\_NAME');  $res = $out->render();  $this->output($res);  }  } |

* Controler Report

|  |
| --- |
| <?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');  class Request extends CI\_Controller {    public function \_\_construct()  {  parent::\_\_construct();  $this->load->model('m\_request');  }    public function saldo()  {  $data = $this->m\_request->get\_saldo();  $this->render\_xml($data);  }    public function last\_tr()  {  $data = $this->m\_request->get\_last\_tr();  $this->render\_xml($data);  }    public function pelanggan($nama = "")  {  $by = array(  'CUSTOMER\_NAME' => $nama,  'PHONE\_NUMBER' =>$nama  );  $data = $this->m\_request->get\_pelanggan($by);  // echo $this->db->last\_query();  $this->render\_xml($data);  }    public function all\_tr()  {  $data = $this->m\_request->get\_all\_tr();  $this->render\_xml($data);  }    public function profit()  {  $data = $this->m\_request->get\_profit();  $this->render\_xml($data);  }  public function test()  {  $data = $this->m\_request->callproc();  var\_dump($data);  }    public function render\_xml($data)  {  header("Content-type: text/xml");  $XML = "<?xml version=\"1.0\"?>\n";  $col\_name = $data->list\_fields();  // root node  $XML .= "<result>\n";  // rows  foreach ($data->result() as $value) {  $XML .= "\t<row>\n";  $i = 0;  // cells  foreach ($value as $cell) {  // Escaping illegal characters - not tested actually ;)  $cell = str\_replace("&", "&amp;", $cell);  $cell = str\_replace("<", "&lt;", $cell);  $cell = str\_replace(">", "&gt;", $cell);  $cell = str\_replace("\"", "&quot;", $cell);  // creates the "<tag>contents</tag>" representing the column  $XML .= "\t\t<".$col\_name[$i].">".$cell."</" .$col\_name[$i].">\n";  $i++;  }  $XML .= "\t</row>\n";  }  $XML .= "</result>\n";  // output the whole XML string  echo $XML;  }  } |

* Model Report

|  |
| --- |
| <?php if (!defined('BASEPATH')) exit('No direct script access allowed');    class M\_Request extends CI\_Model {    function \_\_construct() {  parent::\_\_construct();  $this->load->database();  }    function get\_saldo()  {  $data = $this->db->query('  select DEALER\_NAME, BALANCE  from balance join dealer on balance.DEALER\_ID = dealer.DEALER\_ID  ');  return $data;  }    function get\_last\_tr()  {  $data = $this->db->query('  select \*  from  transaction\_details  order by transaction\_details.TR\_ID desc  limit 0,1  ');  return $data;  }    function get\_pelanggan($data)  {  $this->db->or\_like($data);  $this->db->select('customer.CUSTOMER\_ID, CUSTOMER\_NAME, PHONE\_NUMBER');  $this->db->join('phone\_number', 'phone\_number.CUSTOMER\_ID = customer.CUSTOMER\_ID', 'left');  $data = $this->db->get('customer');  return $data;  }    function get\_all\_tr()  {  return $this->db->get('transaction\_details');  }    function get\_profit()  {  return $this->db->query('  select EXTRACT(month FROM tr.TR\_DATE) as BULAN, sum(tr.TR\_SELL\_PRICE)-sum(tr.TR\_BUY\_PRICE) as PROFIT  from transaction\_details tr  group by EXTRACT(month FROM tr.TR\_DATE)');  }  } |

* Antarmuka Aplikasi Web

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <meta charset="utf-8" />  <meta name="viewport" content="width=device-width, initial-scale=1.0">  <?php foreach ($css\_files as $file): ?>  <link type="text/css" rel="stylesheet" href="<?php echo $file; ?>" />  <?php endforeach; ?>  <link type="text/css" rel="stylesheet" href="<?php echo base\_url('assets/css/bootstrap.min.css') ?>"></link>  <link type="text/css" rel="stylesheet" href="<?php echo base\_url('assets/css/bootstrap-theme.css') ?>"></link>  <?php foreach ($js\_files as $file): ?>  <script src="<?php echo $file; ?>"></script>  <?php endforeach; ?>  <script src="<?php echo base\_url('assets/js/bootstrap.min.js'); ?>"></script>  <!--<script src="<?php echo base\_url('assets/js/jquery.min.js'); ?>"></script>-->  </head>  <body>  <div class="navbar navbar-default navbar-fixed-top">  <div>  <div class="col-md-12 column">  <ul class="nav">  <a href="<?php echo site\_url('manage') ?>" class="navbar-brand pull-left">BakulPulsa</a>  </ul>  </div>  </div>  </div>  <div style="height:80px"></div>  <div class="col-md-12">  <div class="col-md-3">  <div id="Menu">  <div class="list-group panel">  <a href="#MenuList" class="list-group-item list-group-item-info" data-toggle="collapse" data-parent="#Menu">Menu</a>  <div class="collapse in" id="MenuList">  <a class="list-group-item" href="<?php echo site\_url('manage/balance'); ?>">Balance</a>  <a class="list-group-item" href="<?php echo site\_url('manage/customer') ?>">Customer</a>  <a class="list-group-item" href="<?php echo site\_url('manage/dealer') ?>">Dealer</a>  <a class="list-group-item" href="<?php echo site\_url('manage/phone\_number') ?>">Customer Phone</a>  <a class="list-group-item" href="<?php echo site\_url('manage/refill\_package') ?>">Package</a>  <a class="list-group-item" href="<?php echo site\_url('manage/transaction\_details') ?>">Transaction</a>  </div>  <a href="#ReportList" class="list-group-item list-group-item-info" data-toggle="collapse" data-parent="#Menu">Report</a>  <div class="collapse" id="ReportList">  <a class="list-group-item" target="\_blank" href="<?php echo site\_url('request/saldo'); ?>">Balance</a>  <a class="list-group-item" target="\_blank" href="<?php echo site\_url('request/last\_tr'); ?>">Last Transaction</a>  <a href="#customer" class="list-group-item" data-toggle="collapse">Customer</a>  <div class="collapse" id="customer" class="list-group-item list-group-item-success">  <div class="list-group-item">  <div class="input-group">  <input class="form-control" placeholder="Name or Phone Number" type="text" id="custName" name="custName" onchange="update()">  <span class="input-group-btn">  <a target="\_blank" id="btnCust" href="<?php echo site\_url('request/pelanggan/'); ?>" class="btn btn-default">Search</a>  </span>  </div>  </div>  </div>  <a class="list-group-item" target="\_blank" href="<?php echo site\_url('request/all\_tr'); ?>">All Transaction</a>  <a class="list-group-item" target="\_blank" href="<?php echo site\_url('request/profit'); ?>">Profit</a>  </div>  </div>  </div>  </div>  <div class="col-md-9">  <?php echo $output; ?>  </div>  </div>  <script>  function update() {  var inp = document.getElementById("btnCust");  var ind = document.getElementById("custName");  inp.href = '<?php echo site\_url('request/pelanggan') ?>' + '/' + ind.value;  }  </script>  </body>  </html> |