

APLIKASI SMART LAUNDRY BERBASIS ANDROID

TUGAS AKHIR

Oleh :

Arif Ravindra

3311011038

Muhammad Fiqih Tandomato

3311011040

Disusun untuk memenuhi syarat kelulusan Program Diploma III



PROGRAM STUDI TEKNIK INFORMATIKA

POLITEKNIK NEGERI BATAM

BATAM

2013

LEMBAR PENGESAHAN

Batam, 17 Juli 2013

Pembimbing,

Mir'atul Khusna Mufida.S,ST

NIK. 109057

LEMBAR PERNYATAAN

Dengan ini, saya:

NIM : Arif Ravindra

Nama : 3311011038

adalah mahasiswa Teknik Informatika Politeknik Batam yang menyatakan bahwa tugas akhir dengan judul:

APLIKASI SMART LAUNDRY BERBASIS ANDROID

disusun dengan:

1. tidak melakukan plagiat terhadap naskah karya orang lain
2. tidak melakukan pemalsuan data
3. tidak menggunakan karya orang lain tanpa menyebut sumber asli atau tanpa ijin pemilik

Jika kemudian terbukti terjadi pelanggaran terhadap pernyataan di atas, maka saya bersedia menerima sanksi apapun termasuk pencabutan gelar akademik.

Lembar pernyataan ini juga memberikan hak kepada Politeknik Batam untuk mempergunakan, mendistribusikan ataupun memproduksi ulang seluruh hasil Tugas Akhir ini.

Batam, 17 Juli 2013

Arif Ravindra

3311011038

LEMBAR PERNYATAAN

Dengan ini, saya:

NIM : Muhammad Fiqih Tandomato

Nama : 3311011040

adalah mahasiswa Teknik Informatika Politeknik Batam yang menyatakan bahwa tugas akhir dengan judul:

APLIKASI SMART LAUNDRY BERBASIS ANDROID

disusun dengan:

1. tidak melakukan plagiat terhadap naskah karya orang lain
2. tidak melakukan pemalsuan data
3. tidak menggunakan karya orang lain tanpa menyebut sumber asli atau tanpa ijin pemilik

Jika kemudian terbukti terjadi pelanggaran terhadap pernyataan di atas, maka saya bersedia menerima sanksi apapun termasuk pencabutan gelar akademik.

Lembar pernyataan ini juga memberikan hak kepada Politeknik Batam untuk mempergunakan, mendistribusikan ataupun memproduksi ulang seluruh hasil Tugas Akhir ini.

Batam, 17 Juli 2013

Muhammad Fiqih Tandomato

3311011040

HALAMAN PERSEMBAHAN

Dengan rasa yang penuh bangga Tugas Akhir ini kupersembahkan kepada:

My beloved family, Ayah saya Aminuddin dan ibu saya Femil Um Kharyani Tandomato yang selama ini telah mendidik, menyemangati dan memotivasi sehingga saya bisa seperti saat sekarang ini.

Kemudian saya persembahkan juga untuk Alm.Amanuddin Tandomato, sosok atok(kakek) yang sangat saya rindukan saat ini, yang mana dari saya kecil dialah yang senantiasa memacu semua cucu-cucunya untuk dapat berprestasi di bidang akademik.

Semoga seterusnya saya dapat menjadi kebanggan kalian. Salam sayang yang terdalam buat kalian, Ayah, Ibu dan Atok ku tersayang.

Yang Kalian banggakan

Muhammad Fiqih Tandomato

HALAMAN PERSEMBAHAN

Dengan rasa yang penuh bangga Tugas Akhir ini kupersembahkan kepada:

My beloved family, Ayah dan ibu saya yang selama ini telah mendidik, menyemangati dan memotivasi sehingga saya bisa seperti saat sekarang ini.

Kemudian saya persembahkan juga tugas akhir ini untuk Hanim Yulia Suryanti istriku tercinta yang senantiasa memotivasi dan menyemangati saya, dan yang sangat spesial saya persembahkan untuk putri kecil ku Icha yang selalu menjadi pelepas lelah ku.

Semoga seterusnya saya dapat menjadi kebanggan kalian. Salam sayang yang terdalam buat kalian, Ayah, Ibu serta Istri dan anakku.

Yang Kalian banggakan

Arif Ravindra

KATA PENGANTAR

Dengan memanjatkan segala puji dan syukur kepada Tuhan Yang Maha Esa yang telah memberikan rahmat dan karunia-Nya yang sangat besar sehingga penulis dapat menyelesaikan Tugas akhir ini dengan baik. Adapun maksud penulisan tugas akhir yang berjudul **"APLIKASI SMART LAUNDRY BERBASIS ANDROID"** adalah sebagai syarat dalam menyelesaikan pendidikan jenjang Diploma III guna mencapai gelar Ahli Madya Komputer dari Politeknik Negeri Batam.

Dengan selesainya penulisan tugas akhir ini, penulis menyadari bahwa semua ini tidak terlepas dari semua pihak yang turut membantu guna terselesaikannya tugas akhir ini baik secara langsung ataupun tidak langsung. Untuk itu dengan segenap hati penulis mengucapkan terima kasih kepada :

1. Ibu Hilda Widyastuti, M.T selaku kepala prodi Teknik Informatika Politeknik Negeri Batam
2. Bapak Ari Wibowo, MT selaku dosen pengampu matakuliah tugas akhir.
3. Bapak Andi Triwinarko, MT selaku dosen wali.
4. Ibu Mir'atul Khusna Mufida, S.ST selaku dosen pembimbing yang sangat banyak membantu dan membimbing dalam menyelesaikan tugas akhir ini.
5. Bapak Riwinoto, ST, M.Kom selaku dosen penguji saat seminar tugas akhir yang banyak memberi pengarahan perbaikan.
6. Bapak Dwi Ely Kurniawan, M.Kom selaku dosen penguji saat seminar tugas akhir yang banyak memberikan pengarahan perbaikan.
7. Seluruh staff Dosen di Politeknik Negeri Batam yang tidak dapat penulis sebutkan satu persatu yang selama ini telah banyak memberikan ilmu kepada penulis.

8. Kedua orang tua yang senantiasa menjadi memberi semangat dan dukungan moril selama pengerjaan tugas akhir ini
9. Tidak lupa penulis juga mengucapkan terimakasih kepada seluruh teman-teman seperjuangan yang sering kumpul di ruangan Ormawa untuk mengerjakan tugas akhir, yang senantiasa saling membantu, saling mendukung, dan saling mengingatkan selama penyusunan tugas akhir ini

Penulis menyadari dari penyusunan tugas akhir masih terdapat banyak kekurangan, oleh karena itu dengan rendah hati penulis menyampaikan permohonan maaf dan akan menerima kritik dan saran membangun dari pembaca. Akhir kata penulis mengucapkan terimakasih, semoga tugas akhir ini bermanfaat bagi semuanya.

Batam, Juli 2013

Penulis

ABSTRAK

APLIKASI SMART LAUNDRY BERBASIS ANDROID

Pada saat ini sistem laundry menggunakan sistem pencatatan manual. Dimana tidak adanya informasi yang tersedia di usaha laundry tersebut tentang proses yang dijalani pakaian milik pelanggan. Dan juga pelanggan harus mendatangi ataupun menelfon petugas laundry untuk mencari informasi harga terkini dan meminta penjemputan pakaian.

Saat ini juga telah diketahui sebuah sistem dimana webserver dapat terhubung dengan *smartphone android* dalam pertukaran datanya. Tugas akhir ini melakukan penelitian dan pembuatan aplikasi tentang manfaat keterhubungan webserver dengan *smartphone* dalam proses administrasi dan panggilan petugas di proses laundry. Metode yang digunakan untuk *parsing* data antara *smartphone* dengan *server* menggunakan JSON.

Dengan metode yang digunakan, maka dihasilkan sebuah aplikasi dimana pelanggan dengan mudah meminta permohonan penjemputan, pelacakan status tentang tahap pencucian dan pencarian tentang informasi harga terkini.

Kata kunci: Smart Laundry, JSON, Android, Client-Server

ABSTRACT

SMART LAUNDRY APPLICATION BASED ANDROID

In these days recording of laundry system still manual way. Where there is no information that provide about process of clothes from customer. And also customer has to visit or call staff laundry for find information price and pick-up clothes.

Right now have been know a new system where web server could connect with android *smartphone* for data exchange. This final project has been doing research and made an application about advantage from connectedness web server with *smartphone* in process administration and staff calling in laundry process method used for parsing data between *smartphone* with server was using JSON.

With the method used, than the result an application where customer will be more easy to do request the pick-up , tracking status about washing step and about information the newest price.

Key words: Smart Laundry, JSON, Android, Client-Server

DAFTAR ISI

LEMBAR PENGESAHAN.....	ii
LEMBAR PERNYATAAN	iii
LEMBAR PERNYATAAN	iv
HALAMAN PERSEMBAHAN.....	v
KATA PENGANTAR	vii
ABSTRAK	ix
ABSTRACT	ix
DAFTAR ISI	xi
DAFTAR GAMBAR	xiii
DAFTAR TABEL	xiv
Bab I Pendahuluan.....	1
I.1 Latar Belakang.....	1
I.2 Rumusan Masalah.....	2
I.3 Batasan Masalah	2
I.4 Tujuan	3
I.5 Sistematika Penulisan	3
Bab II Landasan Teori	4
II.1 Laundry	4
II.1.1 Tahapan Laundry	4
II.2 Android.....	8
II.2.1 Arsitektur Android.....	8
II.2.2 Versi Android	9
II.2.3 ADT(Android Development Tools)	12
II.2.4 Android SDK (Software Development Kit)	13
II.3 Java	13
II.3.1 Fase Pemrograman Java	13
II.3.2 JRE (Java Runtime Environment)	15
II.3.3 JSON (Java Script Object Notation).....	15

II.4	Eclipse	15
II.4.1	Sejarah Eclipse	16
II.4.2	Arsitektur Eclipse	16
II.4.3	Versi Eclipse.....	17
II.5	HTML.....	17
II.6	PHP(<i>Hypertext Preprocessor</i>).....	18
II.6.1	Sejarah PHP.....	18
II.7	MySQL	19
II.8	UML(United Modeling Language).....	19
II.8.1	Diagram Dalam UML.....	19
II.8.2	Tujuan Penggunaan UML.....	22
II.8.3	Perangkat lunak yang mendukung pembuatan diagram UML... 22	
Bab III	Analisis dan Perancangan	24
III.1	Analisis	24
III.1.1	Analisis Kinerja Sistem Secara Umum	24
III.1.2	Analisis Kinerja Sistem Aplikasi Smart Laundry.....	25
III.1.3	Analisis Kebutuhan Sistem.....	25
III.1.4	Analisis <i>Use Case Diagram</i>	29
III.1.5	Analisis Class	31
III.1.6	Analisis Sequence Diagram	33
III.1.7	Class Diagram.....	41
III.2	Perancangan.....	42
III.2.1	Perancangan Algoritma	43
III.2.2	Interface Aplikasi Di Sisi Customer (Pelanggan).....	47
Bab IV	Implementasi dan Pengujian.....	57
IV.1	Implementasi <i>interface</i> (Antarmuka).....	57
IV.1.1	<i>Interface customer</i> (pelanggan)	57
IV.1.2	<i>Interface</i> petugas.....	59
IV.1.3	<i>Interface Admin</i>	62
IV.2	Hasil Uji.....	68

Bab V	Kesimpulan dan Saran	74
V.1	Kesimpulan	74
V.2	Saran	74

DAFTAR GAMBAR

Gambar 1 Tahapan proses laundry	7
Gambar 2 Fase pemrograman java.....	14
Gambar 3 Notasi actor	20
Gambar 4 Notasi use case	20
Gambar 5 Notasi interaction	20
Gambar 6 Deskripsi umum sistem	24
Gambar 7 Use case diagram customer	29
Gambar 8 Use case petugas.....	30
Gambar 9 Use case admin.....	30
Gambar 10 Use case keterhubungan server	31
Gambar 11 Robustness diagram customer	31
Gambar 12 Robustness diagram admin.....	32
Gambar 13 Robustness diagram petugas	33
Gambar 14 Sequence diagram login pelanggan	33
Gambar 15 Sequence diagram request	34
Gambar 16 Sequence diagram ubah sandi	34
Gambar 17 Sequence diagram tampil status	35
Gambar 18 Sequence diagram masukkan pakaian	35
Gambar 19 Sequence diagram tampil history	36
Gambar 20 Sequence diagram login admin	36
Gambar 21 Sequence diagram tampil user account	37
Gambar 22 Sequence diagram kelola user account.....	37
Gambar 23 Sequence diagram tampil data pelanggan	38
Gambar 24 Sequence diagram kelola data pelanggan.....	38
Gambar 25 Sequence diagram tampil status cuci.....	39
Gambar 26 Sequence diagram cetak kwitansi.....	39
Gambar 27 Sequence diagram kelola data keranjang	40
Gambar 28 Sequence diagram tampil list harga.....	40

Gambar 29 Sequence diagram kella list harga	41
Gambar 30 Sequence diagram tampil list harga customer	41
Gambar 31 Rancangan class diagram	42
Gambar 32 Rancangan interface Login pelanggan.....	47
Gambar 33 Rancangan interface Menu utama pelanggan.....	48
Gambar 34 Rancangan interface tampil status pencucian.....	48
Gambar 35 Rancangan interface ubah sandi pelanggan.....	49
Gambar 36 Rancangan interface tampil list harga pelanggan	49
Gambar 37 Rancangan interface Menu utama petugas	50
Gambar 38 Rancangan interface pilihan jenis pencucian	50
Gambar 39 Rancangan interface masukkan pakaian per Kg.....	51
Gambar 40 Rancangan interface masukkan pakaian per Pcs	51
Gambar 41 Rancangan interface history cucian.....	52
Gambar 42 Rancangan interface login admin	52
Gambar 43 Rancangan interface home page admin.....	53
Gambar 44 Rancangan interface status cucian.....	53
Gambar 45 Rancangan interface tampil user account	54
Gambar 46 Rancangan interface tampil data pelanggan	54
Gambar 47 Rancangan interface tampil daftar harga.....	55
Gambar 48 Rancangan interface cetak kwitansi	55
Gambar 49 Rancangan interface tampil history laundry	56
Gambar 50 Rancangan interface kelola data pelanggan.....	56
Gambar 51 Interface login pelanggan	57
Gambar 52 Interface menu utama pelanggan.....	58
Gambar 53 Interface status cuci pelanggan.....	58
Gambar 54 Interface tampil list harga.....	59
Gambar 55 Interface ubah sandi pelanggan	59
Gambar 56 Interface menu utama petugas	60
Gambar 57 Interface pilihan pencucian.....	60
Gambar 58 Interface masukkan cucian per Kg	61

Gambar 59 Interface masukkan cucian per Pcs.....	61
Gambar 60 Interface tampil history laundry.....	62
Gambar 61 Interface isi keranjang	62
Gambar 62 Interface login admin.....	63
Gambar 63 Interface home page admin	63
Gambar 64 Interface tampil status cuci.....	64
Gambar 65 Interface tampil data pelanggan.....	64
Gambar 66 Interface tampil list harga.....	65
Gambar 67 Interface tampil user account.....	65
Gambar 68 Interface tampil history.....	66
Gambar 69 Interface cetak kwitansi.....	66
Gambar 70 Interface kelola data pelanggan	67

DAFTAR TABEL

Tabel 1 Sifat eclipse	16
Tabel 2 Vesri eclipse	17
Tabel 3 Hardware untuk pembuatan	26
Tabel 4 Hardware untuk penerapan client dan petugas.....	26
Tabel 5 Hardware penerapan administrator	26
Tabel 6 Hasil uji aplikasi.....	68

Bab I Pendahuluan

I.1 Latar Belakang

Laundry pada saat ini sangat dibutuhkan oleh masyarakat Batam yang sebagian besar penduduk adalah pekerja, pebisnis. *Laundry* itu sendiri telah mengalami banyak perubahan, baik dari segi teknis maupun dari segi teknologi.

Dapat diketahui pada sekarang ini, sistem pendataan pelanggan *Laundry* dilakukan secara manual. Dengan menyertakan kartu anggota kepada pelanggan. Sistem penjemputan *Laundry* tersebut juga dilakukan dengan cara menghubungi *call center Laundry* tersebut. Karena pengerjaan yang secara manual, maka pelanggan tidak dapat memperoleh mengenai informasi proses yang dijalani pakaian yang mereka titipkan.

Selain susahnya mengetahui proses yang dijalani pakaian yang dititipkan, sulitnya menghubungi *call centre* juga menjadi hal yang menghambat permintaan penjemputan dan pencarian informasi tentang harga dikarenakan no *call centre* yang sibuk akibat sedang melayani telfon dari pelanggan yang lain dalam waktu yang bersamaan.

Terkait perkembangan internet sebagai media baru, konsumsi masyarakat terhadap *smartphone* juga semakin meningkat. Jumlah pengguna internet didunia meningkat secara signifikan dalam beberapa tahun terakhir dikarenakan kehadiran teknologi internet seluler tak terkecuali pengguna di kota Batam.

Jika membahas tentang *smartphone* maka akan terlintas sebuah kata yaitu *Android* , yang merupakan sistem operasi yang digunakan di *smartphone* dan juga *tablet PC*. Fungsinya sama seperti sistem operasi *Symbian* di *Nokia*, *iOS* di *Apple* dan *BlackBerry OS*. *Android* tidak terikat ke satu merek *handphone* saja, beberapa vendor terkenal yang sudah memakai *Android* antara lain *Samsung* , *Sony Ericsson*, *HTC*, *Nexus*, dan *Motorolla*.

Android yang telah memasyarakat karna harganya yang relatif murah mulai menyentuh kehidupan masyarakat di banyak bidang. Sebagai contoh *Android* digunakan di restoran untuk mencatat pesanan pelanggan. Ada juga orang – orang yang menggunakan *Android* sebagai pengganti buku catatan dan *reminder*. Dan tidak menutup kemungkinan juga merambat ke dunia *laundry*, yang sekarang ini sedang merentangkan sayapnya di kota Batam.

Maka dari itu dibuatlah suatu aplikasi berbasis *Android* untuk meningkatkan pelayanan dari usaha *laundry* tersebut. Yang mana aplikasi tersebut akan membantu setiap pelanggan untuk mengetahui sudah sampai tahap mana penyelesaian pakaian pelanggan tersebut, mempermudah pelanggan dalam permintaan penjemputan dan mempermudah pelanggan untuk mengetahui harga terkini dari *laundry* tersebut.

I.2 Rumusan Masalah

1. Pelanggan sulit mendapatkan informasi tentang tahap pengerjaan pakaian.
2. Pelanggan sulit mendapatkan informasi harga terkini karena harus mendatangi *Laundry* atau menghubungi *call center*.
3. Sulitnya meminta layanan penjemputan karena harus menghubungi *call centre*.

I.3 Batasan Masalah

1. Aplikasi hanya mendukung pelayanan laundry, tidak menangani cara kerja laundry.
2. Aplikasi pada pengguna di uji coba pada *Android* versi 4.0

I.4 Tujuan

Adapun tujuan dari tugas akhir ini adalah

1. Memberikan kemudahan kepada pelanggan dalam pencarian informasi tentang tahap yang sedang dijalani oleh pakaian atau *garment* milik pelanggan.
2. Mempermudah pelanggan dalam permintaan penjemputan pakaian atau *garment*.
3. Mempermudah pelanggan dalam permintaan penjemputan pakaian

I.5 Sistematika Penulisan

Sistematika penulisan menggambarkan secara singkat organisasi penulisan laporan beserta ringkasan isi dari setiap bagiannya. Pada penulisan laporan ini sistematika penulisan yang dipakai adalah sebagai berikut:

1. Bab I Pendahuluan

Pada bagian pendahuluan ini diulas tentang latar belakang, rumusan masalah, batasan masalah dan tujuan dari tugas akhir yang dilakukan.

2. Bab II Landasan Teori

Pada bagian landasan teori dijelaskan materi-materi atau tinjauan pustaka yang digunakan sebagai pendukung selama pengerjaan tugas akhir.

Pada landasan teori, sub bab materi yang di bahas adalah tentang laundry, android, java, HTML 5, PHP,MySQL dan UML.

3. Bab III Analisis dan Perancangan

Pada bab ini yang ditulis ialah hasil analisis sebelum melakukan perancangan. Analisis yang dilakukan adalah analisis kinerja sistem secara umum, analisis kebutuhan fungsional, analisis kebutuhan non fungsional, analisa kebutuhan hardware dan software untuk sistem, analisis use case,

analisis robustness, analisis sequence diagram dan analisis class diagram. Pada tahap perancangan yang dilakukan ialah perancangan *interface* dan perancangan algoritma sistem.

4. Bab IV Implementasi dan pengujian

Pada bab ini akan dibahas tentang implementasi rancangan sistem yang dibuat, uji hasil implementasi sistem, serta hasil analisis hasil pengujian sistem.

5. Bab V Kesimpulan dan saran

Pada bab ini dijelaskan kesimpulan yang didapat dari pengerjaan tugas akhir ini serta saran untuk pihak lain yang ingin mencoba mengembangkan aplikasi yang dibuat saat ini.

Bab II Landasan Teori

II.1 Laundry

Istilah laundry yang umumnya di Indonesia dari dulu dikenal dengan istilah binatu adalah fasilitas di mana pakaian dicuci dan dikeringkan^[11]. Pelanggan cukup dengan menitipkan pakaiannya, kemudian dua atau tiga hari berikutnya bisa diambil dalam keadaan sudah disetrika rapi. Lebih praktis, tidak repot, murah dan pakaian pun siap pakai.

II.1.1 Tahapan Laundry

Dalam usaha bisnis laundry kiloan kita mengenal beberapa tahapan atau proses antara lain :

1. Administrasi Laundry

Terbagi dari administrasi awal dan administrasi akhir. Administrasi awal terdapat proses pencatatan pakaian, serta penimbangan maupun pencucian pakaian. Lalu administrasi akhir terdiri dari pengecekan pakaian dengan struk kwitansi dan pembayaran.

Pada administrasi awal, pelanggan akan diberikan struk seperti gambar dibawah ini.

2. Penerimaan Cucian Kotor

- a. Timbang pakaian pada mesin timbangan.
- b. Penghitungan jumlah helai cucian, baik kemeja, baju, celana,dll untuk menghindari perselisihan(*dispute*) jumlah pakaian dengan konsumen laundry kiloan.

- c. Penandaan cucian, Pisahkan cucian konsumen laundry kiloan satu dengan lainnya, bisa menggunakan tag pin (*tagging*) atau laundry *bag* tersendiri jika proses pencuciannya satu mesin satu konsumen.
- d. Pisahkan pakaian yang dapat dicuci menggunakan air (Laundry) atau harus menggunakan teknik pencucian *dry cleaning*.
- e. Pisahkan pakaian luntur dan tidak luntur
- f. Pisahkan pakaian putih dan bukan
- g. Pisahkan pakaian yang bernoda berat yang memerlukan proses penghilangan noda khusus (*spotting*).

3. *Spotting / Pre-Wash Process* (proses sebelum pencucian)

Proses pre-wash / Spotting ini dilakukan di mesin cuci atau diluar dengan proses perendaman, sebelum kepada proses pencucian utama, antara lain:

- a. Proses pemutihan pakaian menggunakan pemutihan laundry (*Bleaching*).
- b. Proses Emulsifikasi atau pengekstrakan noda lemak, seperti pakaian-pakaian bengkel yang penuh noda oli.
- c. Proses mematikan kuman atau bakteri (*Disinfeksi*) pada pakaian, dengan menggunakan disinfektant.

Dalam Proses pre-washing biasanya *chemical* (kimia) laundry yang dibutuhkan adalah: General Spotter / Emulsifier / Oxy booster.

4. Proses pencucian Utama

proses ini adalah proses pencucian dengan menggunakan deterjen dan mesin cuci. Ini seperti kita mencuci biasa dengan mesin cuci, dan kemudian pakaian akan dibilas dan kemudian akan di peras (spin)

Dalam proses washing biasanya *chemical* (kimia) laundry yang dibutuhkan adalah: deterjen, pelembut (softener) dan *chemical* tambahan lain yang bertujuan untuk membantu proses pencucian agar lebih bersih.

5. Proses Pengeringan(*Oven*)

proses ini adalah proses pengeringan pakaian. Untuk laundry yang memiliki mesin dryer maka pengeringan dilakukan di mesin dryer. Buat laundry yang tidak memiliki dryer ini adalah proses penjemuran pakaian sampai kering. Proses penyetrikaan

6. Proses Penyetrikaan/Pressing

Proses penyetrikaan adalah proses dimana setelah pakaian dikeringkan kemudian disetrika dan dirapikan sebelum dilakukan tahap akhir yaitu *packing*.

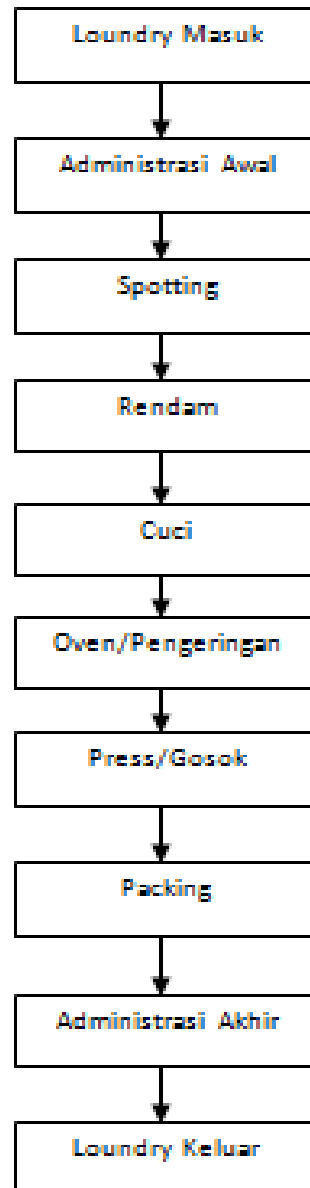
Dalam proses pressing biasanya chemical (kimia) laundry yang dibutuhkan adalah: Pelicin.

7. Proses Finishing / Packing

pada saat packing biasanya pekerja laundry akan memasukan dan memilih pakaian yang telah di setrika didalam satu bungkus sesuai dengan nota bon/pelanggan. Disini juga akan dilakukan scenting atau pemberian parfum agar pakaian menjadi wangi.

Dalam Proses Finishing biasanya chemical (kimia) laundry yang dibutuhkan adalah: Parfum Laundry.

PROSES LOUNDRY



Gambar 1 Tahapan proses laundry

II.2 Android

Android adalah sistem operasi yang berbasis linux sebagai kernelnya yang digunakan pada telepon seluler seperti telepon pintar(*smart phone*) dan komputer tablet.^[6] Saat ini Android bisa disebut raja dari *smartphone*. Android begitu pesat perkembangan di era saat ini, salah satu penyebabnya yaitu karena Android menyediakan platform terbuka (*Open Source*) bagi para pengembang untuk menciptakan aplikasi mereka sendiri. Awalnya, perusahaan search engine terbesar saat ini, yaitu Google Inc. membeli Android Inc. , pendatang baru yang membuat peranti lunak untuk ponsel. Android, Inc. didirikan oleh *Andy Rubin, Rich Milner, Nick Sears* dan *Chris White* pada tahun 2003. Pada Agustus 2005 Google membeli Android Inc. Kemudian untuk mengembangkan Android dibentuklah *Open Handset Alliance* konsorsium dari 34 perusahaan hardware, software dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile dan Nvidia. Pada saat perilis perdana Android, 5 November 2007, Android bersama Open Handset Alliance menyatakan mendukung pengembangan standar terbuka pada perangkat seluler. Di lain pihak, Google merilis kode-kode Android di bawah lisensi Apache, sebuah lisensi perangkat lunak dan standar terbuka perangkat seluler.

Android memiliki dua distributor, yaitu *Google Mail Service (GMS)* dan *Open Handset Distributor (OHD)*. *GMS* adalah distributor Android yang mendapatkan dukungan penuh dari Google, sedangkan *OHD* adalah distibtor Android tanpa dukungan langsung dari Google.

Saat ini sudah banyak bermunculan vendor-vendor untuk *smartphone*, yaitu diantaranya : HTC, Motorola, Samsung, LG, HKC, Huawei, Archos, Webstation Camangi, Dell, Nexus, SciPhone, WayteQ, Sony Ericsson, Acer, Philips, T-Mobile, Nexian, IMO, Asus dan lainnya vendor yang memproduksi *smartphone* Android. Android menjadi pesaing utama dari produk

smartphone lainnya seperti Apple dan BlackBerry. Pesatnya pertumbuhan Android karena Android adalah platform yang sangatlah lengkap baik dari segi sistem operasinya, aplikasi dan tools pengembangannya, market yang menyimpan berbagai aplikasi serta ditambah dengan berbagai dukungan oleh komunitas open source di dunia, sehingga Android berkembang pesat hingga saat ini, baik dari segi teknologi maupun dai segi jumlah device di dunia.

II.2.1 Arsitektur Android

Berikut adalah garis besar pada arsitektur yang Android miliki^[6]:

1. Applications dan widget
2. Applications Frameworks
3. Libraries
4. Android Run Time
5. Linux Kernel

II.2.2 Versi Android

1. Android versi awal (2007 – 2008)

Pada September 2007 Google mengajukan hak paten aplikasi telepon seluler. Google mengenalkan Nexus One, salah satu jenis telepon pintar GSM yang menggunakan Android pada sistem operasinya. Telepon seluler ini diproduksi oleh HTC Corporation dan tersedia di pasaran pada 5 Januari 2010.

Pada 9 Desember 2008, diumumkan anggota baru yang bergabung dalam program kerja Android ARM Holdings, Atheros Communications, diproduksi oleh Asustek Computer Inc, Garmin Ltd, Softbank, Sony Ericsson, Toshiba Corp, dan Vodafone Group Plc. Seiring pembentukan Open Handset Alliance, OHA mengumumkan produk perdana mereka, Android, perangkat bergerak (mobile) yang merupakan modifikasi kernel Linux 2.6. Sejak Android dirilis telah dilakukan berbagai pembaruan berupa perbaikan bug dan penambahan fitur baru.

Smartphone yang memakai sistem operasi Android adalah HTC Dream, yang dirilis pada 22 Oktober 2008. Pada penghujung tahun 2009 diperkirakan di dunia ini paling sedikit terdapat 18 jenis telepon seluler yang menggunakan Android.

1. Android versi 1.1

Pada 9 Maret 2009, Google merilis Android versi 1.1. Android versi ini dilengkapi dengan pembaruan estetis pada aplikasi, jam alarm, voice search (pencarian suara), pengiriman pesan dengan Gmail, dan pemberitahuan email.

2. Android versi 1.5 (Cupcake)

Pada pertengahan Mei 2009, Google kembali merilis telepon seluler dengan menggunakan Android dan SDK (Software Development Kit) dengan versi 1.5 (Cupcake). Terdapat beberapa pembaruan termasuk juga penambahan beberapa fitur dalam seluler versi ini yakni kemampuan merekam dan menonton video dengan modus kamera, mengunggah video ke Youtube dan gambar ke Picasa langsung dari telepon, dukungan Bluetooth A2DP, kemampuan terhubung secara otomatis ke headset Bluetooth, animasi layar, dan keyboard pada layar yang dapat disesuaikan dengan sistem.

3. Android versi 1.6 (Donut)

Donut (versi 1.6) dirilis pada September dengan menampilkan proses pencarian yang lebih baik dibanding sebelumnya, penggunaan baterai indikator dan kontrol applet VPN. Fitur lainnya adalah galeri yang memungkinkan pengguna untuk memilih foto yang akan dihapus; kamera, camcorder dan galeri yang dintegrasikan; CDMA / EVDO, 802.1x, VPN, Gestures, dan Text-to-speech engine; kemampuan dial kontak; teknologi

text to change speech (tidak tersedia pada semua ponsel; pengadaan resolusi VWGA.

4. Android versi 2.0 / 2.1 (Éclair)

Pada 3 Desember 2009 kembali diluncurkan ponsel Android dengan versi 2.0/2.1 (Eclair), perubahan yang dilakukan adalah pengoptimalan hardware, peningkatan Google Maps 3.1.2, perubahan UI dengan browser baru dan dukungan HTML5, daftar kontak yang baru, dukungan flash untuk kamera 3,2 MP, digital Zoom, dan Bluetooth 2.1.

Untuk bergerak cepat dalam persaingan perangkat generasi berikut, Google melakukan investasi dengan mengadakan kompetisi aplikasi mobile terbaik (killer apps - aplikasi unggulan). Kompetisi ini berhadiah \$25,000 bagi setiap pengembang aplikasi terpilih. Kompetisi diadakan selama dua tahap yang tiap tahapnya dipilih 50 aplikasi terbaik.

Dengan semakin berkembangnya dan semakin bertambahnya jumlah handset Android, semakin banyak pihak ketiga yang berminat untuk menyalurkan aplikasi mereka kepada sistem operasi Android. Aplikasi terkenal yang diubah ke dalam sistem operasi Android adalah Shazam, Backgrounds, dan WeatherBug. Sistem operasi Android dalam situs Internet juga dianggap penting untuk menciptakan aplikasi Android asli, contohnya oleh MySpace dan Facebook.

5. Android versi 2.2 (Froyo : Frozen Yoghurt)

Pada 20 Mei 2010, Android versi 2.2 (Froyo) diluncurkan. Perubahan-perubahan umumnya terhadap versi-versi sebelumnya antara lain dukungan Adobe Flash 10.1, kecepatan kinerja dan aplikasi 2 sampai 5 kali lebih cepat, integrasi V8 JavaScript engine yang dipakai Google Chrome yang mempercepat kemampuan rendering pada browser,

pemasangan aplikasi dalam SD Card, kemampuan WiFi Hotspot portabel, dan kemampuan auto update dalam aplikasi Android Market.

6. Android versi 2.3 (Gingerbread)

Pada 6 Desember 2010, Android versi 2.3 (Gingerbread) diluncurkan. Perubahan-perubahan umum yang didapat dari Android versi ini antara lain peningkatan kemampuan permainan (gaming), peningkatan fungsi copy paste, layar antar muka (User Interface) didesain ulang, dukungan format video VP8 dan WebM, efek audio baru (reverb, equalization, headphone virtualization, dan bass boost), dukungan kemampuan Near Field Communication (NFC), dan dukungan jumlah kamera yang lebih dari satu.

7. Android versi 3.0 / 3.1 (Honeycomb)

Android Honeycomb dirancang khusus untuk tablet. Android versi ini mendukung ukuran layar yang lebih besar. User Interface pada Honeycomb juga berbeda karena sudah didesain untuk tablet. Honeycomb juga mendukung multi prosesor dan juga akselerasi perangkat keras (hardware) untuk grafis. Tablet pertama yang dibuat dengan menjalankan Honeycomb adalah Motorola Xoom. Perangkat tablet dengan platform Android 3.0 akan segera hadir di Indonesia. Perangkat tersebut bernama Eee Pad Transformer produksi dari Asus. Rencana masuk pasar Indonesia pada Mei 2011.

8. Android versi 4.0 (Ice Cream)

Android versi 4.0 akan dirilis akhir tahun 2011. Setelah kita ketahui versi Android ini perlu diketahui bahwa nama lain dari versi-versi tersebut diambil oleh Google dari nama makanan penutup.

9. Android Versi 4.1(Jelly Bean)

Jelly Bean berfokus pada platform yang dapat menyuguhkan tiga aspek penting; Simple, Beautiful, and Really Smart. Dibangun dalam sebuah proyek besar Google sudah dipersiapkan sejak lama dan diberi nama Project Butter, yang adalah sebuah perbaikan seluruh sistem untuk meningkatkan penampilan lebih smooth dan responsif terhadap seluruh input disetiap sentuhan pengguna, penambahan kontrol suara pada fitur pencarian, dan penyesuaian ukuran widget dan kustomisasi homescreen lebih mudah dan bebas.

II.2.3 ADT(Android Development Tools)

Android Development Tools (ADT) adalah plugin untuk *Eclipse* IDE yang dirancang untuk memberikan lingkungan yang terintegrasi yang kuat di mana untuk membangun aplikasi *Android*.^[2]

ADT memperluas kemampuan *Eclipse* untuk membantu Anda cepat membuat proyek baru *Android*, membuat aplikasi *UI*, menambahkan paket didasarkan pada *Android* Kerangka API, debug aplikasi Anda menggunakan alat SDK *Android*, dan bahkan ekspor ditandatangani (atau *unsigned*). File apk dalam rangka untuk mendistribusikan aplikasi Anda.

Mengembangkan di *Eclipse* dengan ADT sangat dianjurkan dan merupakan cara tercepat untuk memulai. Dengan dipandu setup proyek menyediakan, serta integrasi alat, *XML* kustom *editor*, dan *panel output debug*, ADT memberikan Anda sebuah peningkatan yang luar biasa dalam mengembangkan aplikasi *Android*.

II.2.4 Android SDK (Software Development Kit)

Android SDK merupakan sebuah tools yang menyediakan *API Libraries* diperlukan untuk membangun suatu aplikasi berbasis android.^[3] Pada saat ini Android SDK telah menjadi alat bantu dan API (Application Programming Interface) untuk mengembangkan aplikasi berbasis Android.

II.3 Java

Java adalah bahasa pemrograman yang berorientasi objek (OOP) dan dapat dijalankan pada berbagai *platform* sistem operasi. Perkembangan Java tidak hanya terfokus pada satu sistem operasi, tetapi dikembangkan untuk berbagai sistem operasi dan bersifat *open source*. Java berasal dari pengembangan bahasa C++ sehingga strukturnya seperti bahasa C++.^[7]

James Gosling dan Patrick Naughton adalah orang yang berhasil mengembangkan Java dalam suatu proyek dari SUN Microsystem. Awalnya James ingin memberi bahasa pemrograman tersebut dengan nama OAK yang berasal dari pohon yang terdapat pada kantor James Gosling, tetapi akhirnya diberi nama Java setelah mendapat ide sewaktu minum kopi.

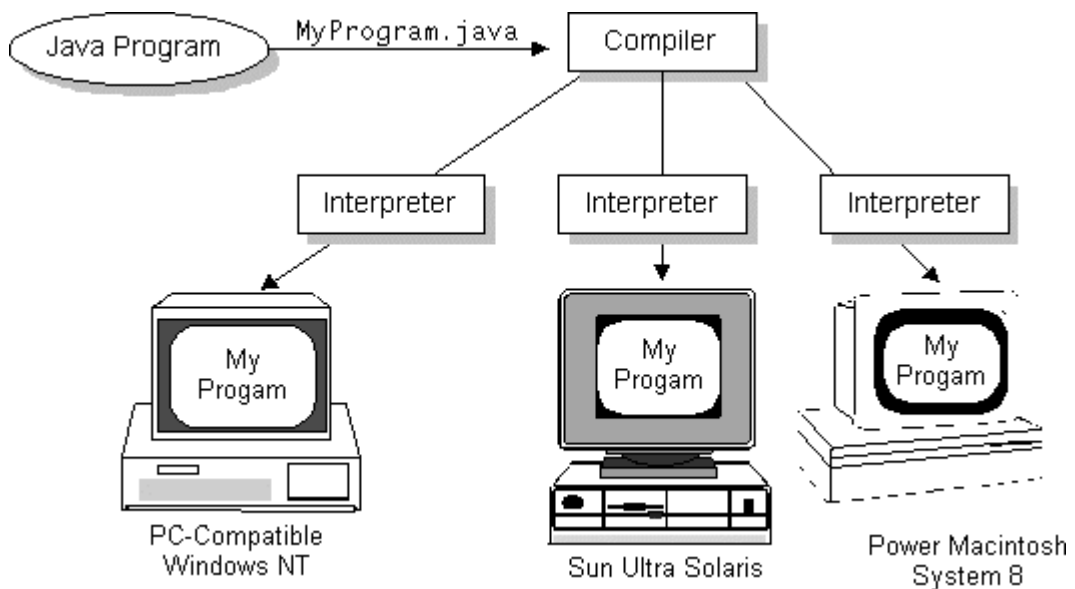
Java sering digunakan oleh semua segmen, karena mampu merambah seluruh hal, dari pemrograman dekstop hingga pemrograman mobile (mobile programming) dan Java juga terkenal mudah digunakan, karena seluruh sistem operasi dapat menjalankan Java. Sistem operasi itu antara lain Window, Linux, DOS, Unix, dan lain-lainnya sehingga banyak pengguna memakainya.

Java telah berkembang dari semula diciptakan untuk pemrograman *applet* (program komputer yang telah dikompilasi menjadi bytecode, yaitu suatu keadaan program yang dapat dieksekusi langsung; tambahan di dalam suatu homepage selain gambar) yang berjalan di web browser menjadi bahasa pemrograman yang terkenal untuk pengembangan berbagai aplikasi komputer yang berjalan di beragam perangkat mulai dari handled devices

seperti handphone, PDA (Personal Digital Assistant) sampai aplikasi khusus yang tersebar di berbagai macam komputer server.

II.3.1 Fase Pemrograman Java

Gambar dibawah ini menjelaskan aliran proses kompilasi dan eksekusi sebuah program Java:^[8]



Gambar 2 Fase pemrograman java

Langkah pertama dalam pembuatan sebuah program berbasis Java adalah menuliskan kode program pada *text editor*. Contoh *text editor* yang dapat digunakan antara lain : notepad, vi, emacs dan lain sebagainya.

Kode program yang dibuat kemudian tersimpan dalam sebuah berkas berekstensi *.java*.

Setelah membuat dan menyimpan kode program, kompilasi file yang berisi kode program tersebut dengan menggunakan Java Compiler. Hasil dari adalah berupa berkas *bytecode* dengan ekstensi *.class*.

Berkas yang mengandung *bytecode* tersebut kemudian akan dikonversikan oleh Java Interpreter menjadi bahasa mesin sesuai dengan jenis dan *platform* yang digunakan.

Macam-macam Text Editor / Tool yang umum digunakan untuk menuliskan source code Java, antara lain :

1. NotePad
2. JCreator (<http://www.jcreator.com>)
3. Edit Plus (<http://www.editplus.com>)
4. Gel (<http://www.gexperts.com>)
5. JFrameBuilder (<http://www.mars3000.com>)
6. NetBeans (www.netbeans.org)
7. Eclipse (<http://www.eclipse.org>)
8. Borland JBuilder (www.borland.com)
9. JDeveloper (<http://www.oracle.com/technology/jdev>)

Dari jenis text editor yang tersedia untuk bahasa java diatas, yang digunakan dalam pengembangan aplikasi *Smart Laundry* Berbasis Android adalah text editor Eclips.

II.3.2 JRE (Java Runtime Environment)

Java Runtime Environment (JRE) adalah salah satu teknologi yang dibuat dan dikembangkan oleh Oracle. Java Runtime Environment merupakan perangkat lunak yang digunakan untuk menjalankan aplikasi yang dibangun menggunakan java. Versi JRE harus sama atau lebih tinggi dari JDK yang digunakan untuk membangun aplikasi agar aplikasi dapat berjalan sesuai dengan yang diharapkan.^[8] Dengan Java Runtime Environment , anda dimungkinkan untuk menjalankan aplikasi yang disebut “applet” yang ditulis dalam bahasa pemrograman berbasis Java. Dengan Applet ini, memungkinkan anda dan siapa saja untuk merasakan pengalaman yang jauh lebih mengesankan daripada sekedar berinteraksi dengan halaman HTML yang statis dan kaku.

Java Plug-in technology, termasuk dalam bagian Java Runtime Environment Standard Edition (JRE), yang menghubungkan antara browser dengan platform berbasis Java. Banyak aplikasi lintas platform juga memerlukan Java untuk dapat berfungsi dengan baik dan dapat digunakan.

II.3.3 JSON (Java Script Object Notation)

JSON adalah format pertukaran data yang ringan, mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan di buat(generate) oleh komputer. Format ini dibuat berdasarkan bagian dari bahasa pemrograman JavaScript, Standar ECMA-262 Edisi ke-3 Desember 1999. JSON sendiri merupakan format teks yang tidak tergantung pada bahasa pemrograman apapun karena menggunakan gaya bahasa yang umum digunakan oleh programmer keluarga C, termasuk C, C++, C3, Java, JavaScript, Perl, Python, dan lain-lain. Oleh karena sifat-sifat tersebut, menjadikan JSON ideal sebagai pertukaran data.^[5]

II.4 Eclipse

Eclipse adalah sebuah IDE (*Integrated Development Environment*) untuk mengembangkan perangkat lunak dan dapat dijalankan di semua platform (*platform-independent*). Berikut ini adalah sifat dari Eclipse : ^[10]

Tabel 1 Sifat eclipse

Multi-platform	Target sistem operasi Eclipse adalah Microsoft Windows, Linux, Solaris, AIX, HP-UX dan Mac OS X.
Multi-language	Eclipse dikembangkan dengan bahasa pemrograman Java, akan tetapi Eclipse mendukung pengembangan aplikasi berbasis bahasa pemrograman lainnya, seperti C/C++, Cobol, Python, Perl, PHP, dan lain sebagainya.
Multi-role	Selain sebagai IDE untuk pengembangan aplikasi, Eclipse pun bisa digunakan untuk aktivitas dalam siklus pengembangan perangkat lunak, seperti dokumentasi, test perangkat lunak, pengembangan web, dan lain sebagainya.

Eclipse pada saat ini merupakan salah satu IDE favorit dikarenakan gratis dan *open source*, yang berarti setiap orang boleh melihat kode pemrograman perangkat lunak ini. Selain itu, kelebihan dari Eclipse yang membuatnya populer adalah kemampuannya untuk dapat dikembangkan oleh pengguna dengan komponen yang dinamakan *plug-in*.

II.4.1 Sejarah Eclipse

Eclipse awalnya dikembangkan oleh IBM untuk menggantikan perangkat lunak pengembangan IBM Visual Age for Java 4.0. Produk Eclipse ini diluncurkan oleh IBM pada tanggal 5 November 2001. IBM menginvestasikan US\$ 40 juta untuk pengembangannya. Sejak 5 November 2001, konsorsium Eclipse Foundation mengambil alih pengembangan Eclipse lebih lanjut.^[10]

II.4.2 Arsitektur Eclipse

Sejak versi 3.0, Eclipse pada dasarnya merupakan sebuah *kernel*. Apa yang dapat digunakan di dalam Eclipse sebenarnya adalah fungsi dari *plug-in* yang sudah dipasang (diinstal). Ini merupakan basis dari Eclipse yang dinamakan *Rich Client Platform* (RCP).^[10]

Berikut ini adalah komponen yang membentuk RCP:

1. *Core platform*
2. OSGi
3. SWT (*Standard Widget Toolkit*)
4. JFace
5. *Eclipse Workbench*

Secara standar Eclipse selalu dilengkapi dengan JDT (*Java Development Tools*), *plug-in* yang membuat Eclipse kompatibel untuk mengembangkan program Java, dan PDE (*Plug-in Development Environment*) untuk mengembangkan *plug-in* baru. Eclipse beserta *plug-in*-nya diimplementasikan dalam bahasa pemrograman Java.

II.4.3 Versi Eclipse

Sejak tahun 2006, Eclipse Foundation mengkoordinasikan peluncuran Eclipse secara rutin dan simultan yang dikenal dengan nama *Simultaneous Release*. Setiap versi peluncuran terdiri dari Eclipse Platform dan juga sejumlah proyek yang terlibat dalam proyek Eclipse.

Tujuan sistem ini adalah untuk menyediakan distribusi Eclipse dengan fitur-fitur dan versi yang terstandarisasi. Hal ini juga dimaksudkan untuk mempermudah *deployment* dan *maintenance* untuk sistem enterprise, serta untuk kenyamanan. Peluncuran simultan dijadwalkan pada bulan Juni setiap tahunnya.

Tabel 2 Vesri eclipse

Kode Peluncuran	Tanggal Peluncuran	Platform	Nama Proyek
Eclipse 3.0	28 Juni 2004	3.0	
Eclipse 3.0	28 Juni 2005	3.1	
Calisto	30 Juni 2006	3.2	Calisto Project
Europa	29 Juni 2007	3.3	Europa Project
Ganymede	25 Juni 2008	3.4	Ganymede Project
Galileo	24 Juni 2009	3.5	Galileo Project
Helios	23 Juni 2010	3.6	

II.5 HTML

HTML merupakan pengembangan dari standar pemformatan dokumen teks yaitu *Standard Generalized Markup Language* (SGML). HTML sebenarnya adalah dokumen ASCII atau teks biasa, yang dirancang untuk tidak tergantung pada suatu sistem operasi tertentu. HTML dibuat oleh *Tim Berners-Lee* ketika masih bekerja untuk *CERN* dan dipopulerkan pertama

kali oleh *browser* Mosaic. Selama awal tahun 1990, HTML mengalami perkembangan yang sangat pesat.

Setiap pengembangan HTML pasti akan menambahkan kemampuan dan fasilitas yang lebih baik dari versi sebelumnya. Namun perkembangan resmi dikeluarkan pada bulan November 1995 oleh IETF (*Internet Engineering Task Force*). HTML 2.0 ini merupakan penyempurnaan dari HTML+ (1993). HTML 3.0 (1995) memberikan kemampuan lebih daripada versi sebelumnya. Sebuah Usaha dari *World Wide Web Consortium's* (W3C) HTML Working Group pada tahun 1996 menghasilkan HTML 3.2. HTML versi ini secara resmi diterbitkan^[9]

II.6 PHP(*Hypertext Preprocessor*)

PHP adalah bahasa skrip yang dapat ditanamkan atau disisipkan ke dalam HTML. PHP banyak dipakai untuk memrogram situs web dinamis. PHP dapat digunakan untuk membangun sebuah CMS.^[4]

Web Server yang mendukung php dapat ditemukan dimana - mana dari mulai IIS sampai dengan apache, dengan konfigurasi yang relatif mudah. Dalam sisi pengembangan lebih mudah, karena banyaknya milis - milis dan developer yang siap membantu dalam pengembangan. Dalam sisi pemahaman, php adalah bahasa scripting yang paling mudah karena referensi yang banyak.

PHP adalah bahasa open source yang dapat digunakan di berbagai mesin (linux, unix, windows) dan dapat dijalankan secara runtime melalui *console* serta juga dapat menjalankan perintah-perintah *system*.

II.6.1 Sejarah PHP

PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu PHP masih bernama FI (Form Interpreted), yang wujudnya berupa sekumpulan script yang digunakan untuk mengolah data form dari web.

Selanjutnya Rasmus merilis kode sumber tersebut untuk umum dan menamakannya PHP/FI, kependekan dari *Hypertext Preprocessing'/Form Interpreter*. Dengan perilsan kode sumber ini menjadi *open source*, maka banyak programmer yang tertarik untuk ikut mengembangkan PHP.

Pada November 1997, dirilis PHP/FI 2.0. Pada rilis ini interpreter PHP sudah diimplementasikan dalam program C. Dalam rilis ini disertakan juga modul-modul ekstensi yang meningkatkan kemampuan PHP/FI secara signifikan.

Pada tahun 1997, sebuah perusahaan bernama Zend menulis ulang interpreter PHP menjadi lebih bersih, lebih baik, dan lebih cepat. Kemudian pada Juni 1998, perusahaan tersebut merilis interpreter baru untuk PHP dan meresmikan rilis tersebut sebagai PHP 3.0.

Pada pertengahan tahun 1999, Zend merilis interpreter PHP baru dan rilis tersebut dikenal dengan PHP 4.0. PHP 4.0 adalah versi PHP yang paling banyak dipakai pada awal abad ke-21. Versi ini banyak dipakai disebabkan kemampuannya untuk membangun aplikasi web kompleks tetapi tetap memiliki kecepatan dan stabilitas yang tinggi.

Pada Juni 2004, Zend merilis PHP 5.0. Dalam versi ini, inti dari interpreter PHP mengalami perubahan besar. Versi ini juga memasukkan model pemrograman berorientasi objek ke dalam PHP untuk menjawab perkembangan bahasa pemrograman ke arah paradigma berorientasi objek.

II.7 MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (bahasa Inggris: *database management system*) atau DBMS

yang *multithread, multi-user*, dengan sekitar 6 juta instalasi di seluruh dunia. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis dibawah lisensi GNU General Public License (GPL), tetapi mereka juga menjual dibawah lisensi komersial untuk kasus-kasus dimana penggunaannya tidak cocok dengan penggunaan GPL.

Dengan menggunakan MySQL Server maka data dapat diakses oleh banyak pemakai secara bersamaan sekaligus dapat membatasi akses para pemakai berdasarkan *privilege* (hak akses) yang diberikan.^[4]

II.8 UML(United Modeling Language)

UML adalah suatu bahasa yang digunakan untuk menentukan, memvisualisasikan, membangun, dan mendokumentasikan suatu sistem informasi. UML dikembangkan sebagai suatu alat untuk analisis dan desain berorientasi objek oleh Grady Booch, Jim Rumbaugh, dan Ivar Jacobson. Namun demikian UML dapat digunakan untuk memahami dan mendokumentasikan setiap sistem informasi. Penggunaan UML dalam industri terus meningkat. Ini merupakan standar terbuka yang menjadikannya sebagai bahasa pemodelan yang umum dalam industri peranti lunak dan pengembangan sistem.^[12]

UML merupakan suatu kumpulan teknik terbaik yang telah terbukti sukses dalam memodelkan system yang besar dan kompleks. UML tidak hanya digunakan dalam proses pemodelan perangkat lunak, namun hampir dalam semua bidang yang membutuhkan pemodelan.

II.8.1 Diagram Dalam UML

Diagram berbentuk grafik yang menunjukkan simbol elemen model yang disusun untuk mengilustrasikan bagian atau aspek tertentu dari sistem. Sebuah diagram merupakan bagian dari suatu view tertentu dan ketika digambarkan biasanya dialokasikan untuk view tertentu

II.8.1.1 Use Case Diagram

Use case adalah abstraksi dari interaksi antara system dan actor.^[1] Use case bekerja dengan cara mendeskripsikan tipe interaksi antara user sebuah system dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah system dipakai. Use case merupakan konstruksi untuk mendeskripsikan bagaimana system akan terlihat di mata user. Sedangkan use case diagram memfasilitasi komunikasi diantara analis dan pengguna serta antara analis dan client.

II.8.1.1.1 Komponen Pembentuk Use Case Diagram

1. Actor



Gambar 3 Notasi actor

Actor menggambarkan segala pengguna software aplikasi (user). Actor memberikan suatu gambaran jelas tentang apa yang harus dikerjakan software aplikasi. Sebagai contoh sebuah actor dapat memberikan input kedalam dan menerima informasi dari software aplikasi, perlu dicatat bahwa sebuah actor berinteraksi dengan use case, tetapi tidak memiliki kontrol atas use case. Sebuah actor mungkin seorang manusia, satu device, hardware atau sistem informasi lainnya.

2. Use case



Gambar 4 Notasi use case

Use Case adalah kegiatan yang dilakukan actor dan system untuk mencapai suatu tujuan tertentu.

3. Interaction



Gambar 5 Notasi interaction

Interaction digunakan untuk menunjukkan baik aliran pesan atau informasi antar obyek maupun hubungan antar obyek. Biasanya interaction ini dilengkapi juga dengan teks bernama operation signature yang tersusun dari nama operasi, parameter yang dikirim dan tipe parameter yang dikembalikan.

II.8.1.1.2 Relasi Dalam Use Case

1. *Association*, menghubungkan link antar element.
2. *Generalization*, disebut juga *inheritance* (pewarisan), sebuah elemen dapat merupakan spesialisasi dari elemen lainnya.
3. *Dependency*, sebuah element bergantung dalam beberapa cara ke element lainnya.
4. *Aggregation*, bentuk association dimana sebuah elemen berisi elemen lainnya.

II.8.1.1.3 Tipe relasi / *stereotype* pada use case

1. <<include>>, yaitu kelakuan yang harus terpenuhi agar sebuah *event* dapat terjadi, dimana pada kondisi ini sebuah *use case* adalah bagian dari *use case* lainnya.
2. <<extends>>, kelakuan yang hanya berjalan di bawah kondisi tertentu seperti menggerakkan alarm.

3. <<communicates>>, mungkin ditambahkan untuk asosiasi yang menunjukkan asosiasinya adalah *communicates association*. Ini merupakan pilihan selama asosiasi hanya tipe *relationship* yang dibolehkan antara *actor* dan *use case*.

II.8.1.2 Class Diagram

Class adalah dekripsi kelompok obyek-obyek dengan property, perilaku (operasi) dan relasi yang sama. Sehingga dengan adanya class diagram dapat memberikan pandangan global atas sebuah system. Hal tersebut tercermin dari class- class yang ada dan relasinya satu dengan yang lainnya. Sebuah sistem biasanya mempunyai beberapa class diagram. Class diagram sangat membantu dalam visualisasi struktur kelas dari suatu system.

II.8.1.3 Sequence Diagram

Sequence Diagram digunakan untuk menggambarkan perilaku pada sebuah scenario. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara object juga interaksi antaraobject, sesuatu yang terjadi pada titik tertentu dalam eksekusi system.

II.8.2 Tujuan Penggunaan UML

1. Memberikan bahasa pemodelan yang bebas dari berbagai bahas pemrograman dan proses rekayasa.
2. Menyatukan praktek-praktek terbaik yang terdapat dalam pemodelan.
3. Memberikan model yang siap pakai, bahsa pemodelan visual yang ekspresif untuk mengembangkan dan saling menukar model dengan mudah dan dimengerti secara umum.
4. UML bisa juga berfungsi sebagai sebuah (blue print) cetak biru karena sangat lengkap dan detail. Dengan cetak biru ini maka akan bias diketahui informasi secara detail tentang coding program atau bahkan membaca program dan menginterpretasikan kembali ke dalam bentuk diagram (reserve engineering).

II.8.3 Perangkat lunak yang mendukung pembuatan diagram UML

1. StarUML

StarUML adalah sebuah proyek open source untuk mengembangkan cepat, fleksibel, extensible, featureful, dan bebas-tersedia UML / platform MDA berjalan pada platform Win32. Tujuan dari proyek StarUML adalah untuk membangun sebuah alat pemodelan perangkat lunak dan juga platform yang menarik adalah pengganti alat UML komersial seperti Rational Rose, dan sebagainya

2. Acceleo

Acceleo adalah generator kode yang mengubah model menjadi kode. Acceleo mudah digunakan dan menyediakan “dari rak” generator (Jee, Bersih, Php ...) dan template editor untuk Eclipse.

3. ArgoUML

ArgoUML adalah open source UML modeling tool terkemuka dan termasuk dukungan untuk semua diagram UML standar 1.4. Ini berjalan pada setiap platform Java dan tersedia dalam bahasa sepuluh. ArgoUML ditulis seluruhnya di Jawa dan menggunakan Java Kelas Foundation. Hal ini memungkinkan ArgoUML untuk berjalan di hampir semua platform

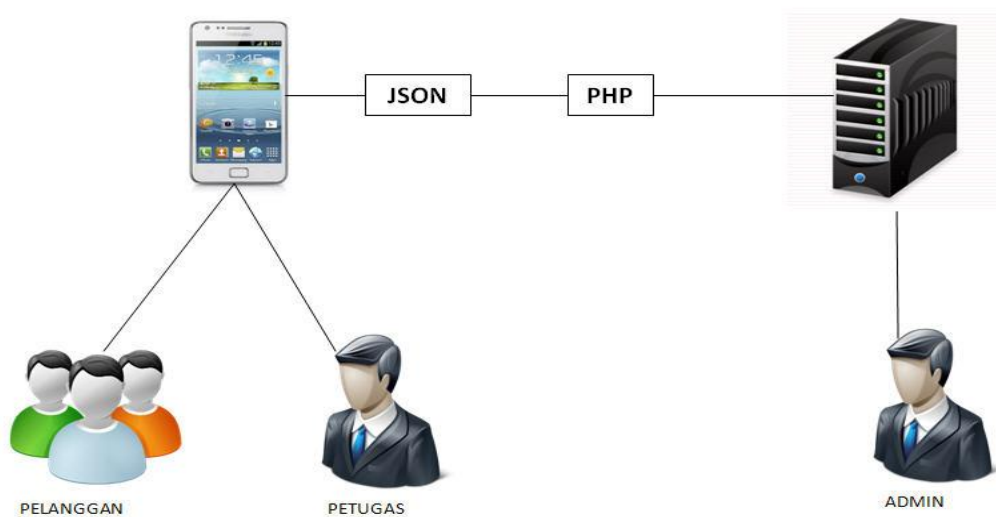
Bab III Analisis dan Perancangan

III.1 Analisis

Analisis sistem adalah penguraian dari suatu sistem informasi secara utuh ke dalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan, kesempatan, hambatan yang terjadi dan kebutuhan yang diharapkan sehingga dapat perancangan dan implementasi yang sempurna dari suatu aplikasi.

III.1.1 Analisis Kinerja Sistem Secara Umum

Secara garis besar, aplikasi Smart Laundry nantinya akan berhubungan dengan server yang dikelola administrator. Pada sisi server, akan dibangun dalam bentuk HTML, PHP, dan MySQL. Untuk proses pertukaran data antara aplikasi dan server, digunakan plugin JSON dan script PHP sehingga aplikasi bisa memasukkan data ke database server dan mengambil data dari database server dengan bantuan plugin JSON.



Gambar 6 Deskripsi umum sistem

III.1.2 Analisis Kinerja Sistem Aplikasi Smart Laundry

Gambaran umum *Aplikasi* ini menjelaskan tentang proses kerja aplikasi *SmartLaundry*. Deskripsi dari proses kerja ini adalah:

1. Pelanggan yang telah terdaftar dapat memanggil petugas dengan memanfaatkan tombol “panggil” pada aplikasi yang terdapat di *smartphone* miliknya.
2. Server menerima notifikasi panggilan dari pelanggan, lalu menugaskan salah satu petugas menjemput pakaian.
3. Petugas mendatangi tempat tinggal pelanggan, lalu memasukkan data pakaian yang akan dicuci melalui *smartphone* milik usaha laundry.
4. Data tersebut masuk ke data keranjang yang terdapat di server.
5. Admin *mengupdate* status pencucian berdasarkan data keranjang.
6. Pelanggan dapat melihat status pencucian, harga, dan data keranjang yang telah *diupdate*.

III.1.3 Analisis Kebutuhan Sistem

Analisis kebutuhan Sistem adalah dimana beberapa kebutuhan bahan dalam sistem yang akan dipergunakan untuk menambah dan membantu jalan proses pembuatan suatu obyek. Dibagian ini akan dibagi menjadi dua bagian yaitu Analisis kebutuhan fungsional dan analisis kebutuhan non-fungsional.

III.1.3.1 Analisis Kebutuhan Fungsional

a. Analisis Kebutuhan Perangkat Keras (*Hardware*)

Perangkat keras adalah alat yang digunakan untuk mengolah data dan penyajian laporan. Perangkat keras yang digunakan terdiri dari :

1. *Hardware* untuk pembuatan

Hardware untuk pembuatan, menggunakan spesifikasi :

Tabel 3 Hardware untuk pembuatan

<i>Processor</i>	AMD C-60 APU
<i>Memory</i>	2048 MB DDR3
<i>Hard Drive</i>	320 GB ATA
<i>Video Card</i>	AMD Radeon 945 MB
<i>Display</i>	14.0" HD LED LCD
<i>Audio</i>	Realtek High Defenition Audio

2. Hardware untuk penerapan client dan petugas

Tabel 4 Hardware untuk penerapan client dan petugas

<i>General</i>	850 / 900 / 1.800 / 1.900MHz GSM & EDGE Band
<i>Dimension</i>	103,9 x 57,9 x 11,98mm Dimension
<i>Display</i>	2,8" External Display Size
<i>Memory</i>	Memory standard 3GB (1GB User Memory + 2GB Shared Storage) dan 32GB External Memory
<i>OS</i>	Android 2.3 (Gingerbread)
<i>Camera</i>	2 Megapixels dengan , 2x Digital Zoom, Single / Smile / Panorama, Photo Effects: Black & White / Negative / No Effect / Sepia dan White Balance: Auto / Daylight / Incandescent / Fluorescent / Cloudy
<i>Connectivity</i>	WIFI 802.11b/g/n, T 3.0 HS Bluetooth, 2.0 High Speed USB, WAP 2.0 /

	HTTP, USB Mass Storage available, Android Internet HTML Browser, SyncML (DM) 1.2, FOTA
<i>Battery</i>	1.200mAh Battery Capacity

3. *Hardware* untuk penggunaan administrator system

Tabel 5 Hardware penerapan administrator

<i>Processor</i>	Intel Core 2 duo 2.00 GHz
<i>Memory</i>	2048 MB DDR3
<i>Hard Drive</i>	320 GB ATA
<i>Video Card</i>	ATI Radeon HD 4330 1024 MB
<i>Display</i>	14.0" HD LED LCD
<i>Audio</i>	IDT Audio

2 Analisis Perangkat Lunak (Software)

a. Software Untuk Pembuatan

Software yang digunakan dalam pembuatan aplikasi adalah :

1. Sistem Operasi Untuk penggunaan sistem operasi menggunakan Windows 7 (32 bit).
2. Bahasa Pemrograman Java Dalam hal ini digunakan *Java Development Kit* (JDK) 2.2 dan *Java Runtime Environment* (JRE).
3. *Integrated Development Environment* (IDE) Eclipse Adapun memakai perangkat lunak ini karena terdapat *Android Development Tools*.
4. *Android Software Development Kit* (*Android SDK*) Dalam hali ini Android SDK menyediakan development environmen dengan semua komponen yang diperlukan.

5. *Android Development Tools* (ADT) Android membuat kostum *plugin* untuk IDE Eclipse yaitu ADT yang dimana memberikan kemudahan untuk pengembangan dalam membangun sebuah sistem aplikasi berbasis android.

b. Software Untuk Penerapan

Software yang digunakan dalam penerapan aplikasi adalah :

1. Dilingkungan *client* membutuhkan system operasi android minimal versi 2.2 (*Froyo*).
2. Dilingkungan server membutuhkan *software* Google Chrome atau Mozilla Firefox.

3 Analisis Kebutuhan Sistem SDM (Brainware)

a. Customer

User adalah pengguna dari aplikasi itu sendiri, user dapat mengakses fitur-fitur dalam aplikasi (Login terlebih dahulu).

b. Admin

Admin mengontrol penuh atau mengolah data.

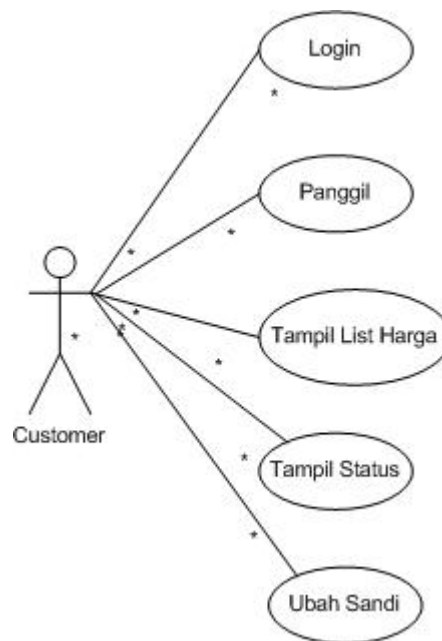
c. Petugas

Mengambil data pakaian yang akan dicuci di tempat tinggal pelanggan.

III.1.4 Analisis Use Case Diagram

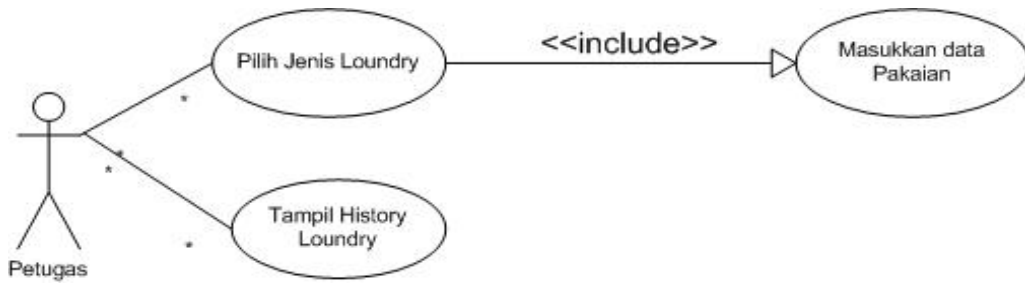
Untuk *use case*, ada 3 use case yang akan di analisis, yaitu *use case* aplikasi pada *smartphone* yang digunakan oleh pelanggan maupun petugas dan aplikasi berbasis *web based* yang digunakan oleh *admin*.

1. Use case Aplikasi *Smartphone*(Customer)



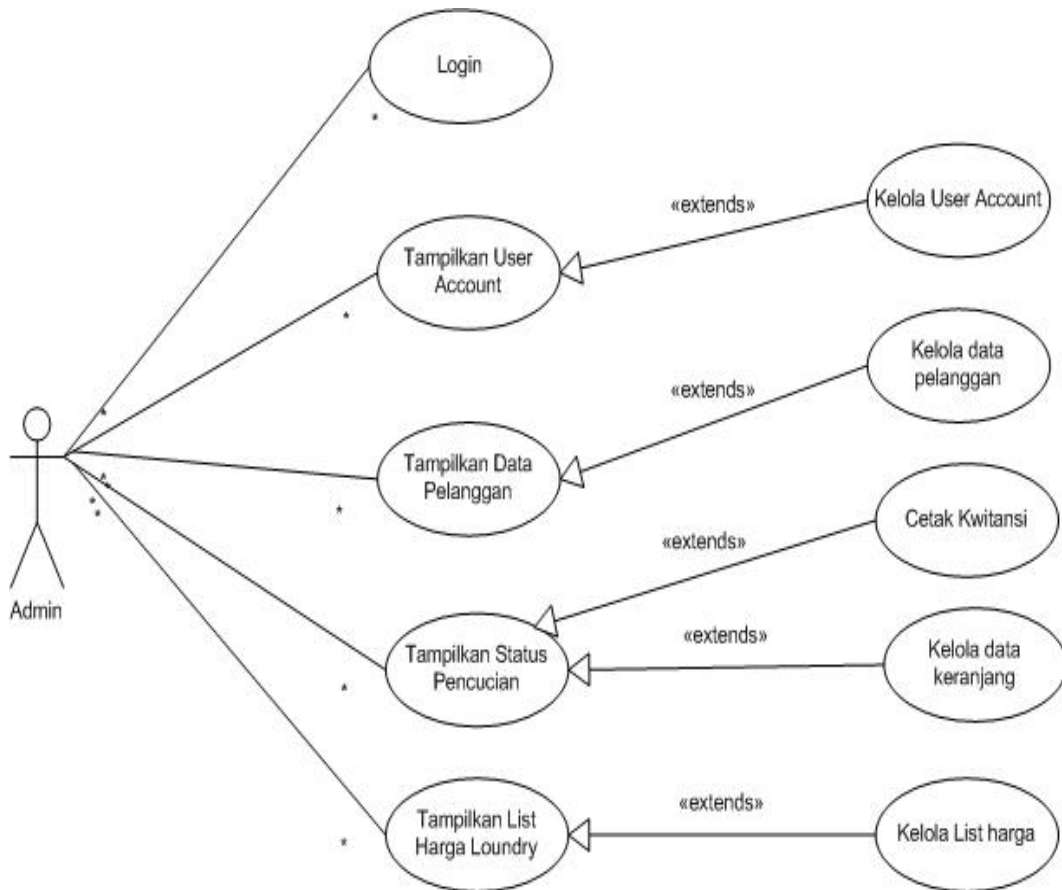
Gambar 7 Use case diagram customer

2. Use Case Aplikasi *Smartphone*(Petugas)



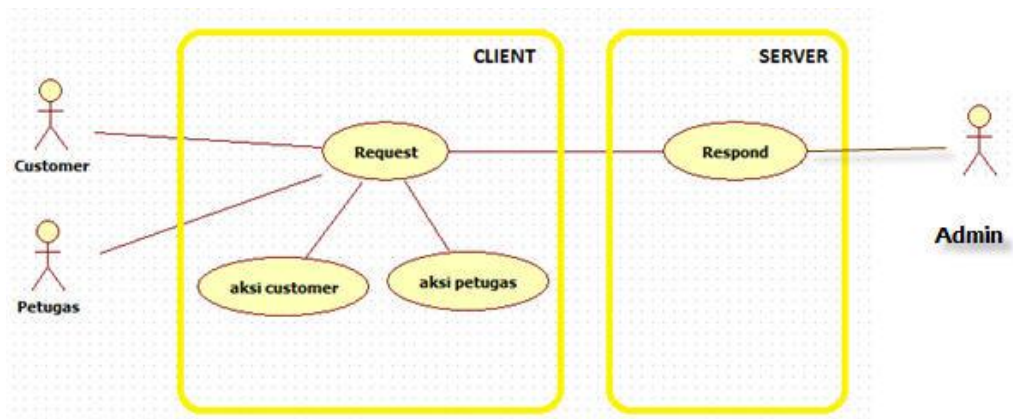
Gambar 8 Use case petugas

3. Use Case Aplikasi Web Based(Admin).



Gambar 9 Use case admin

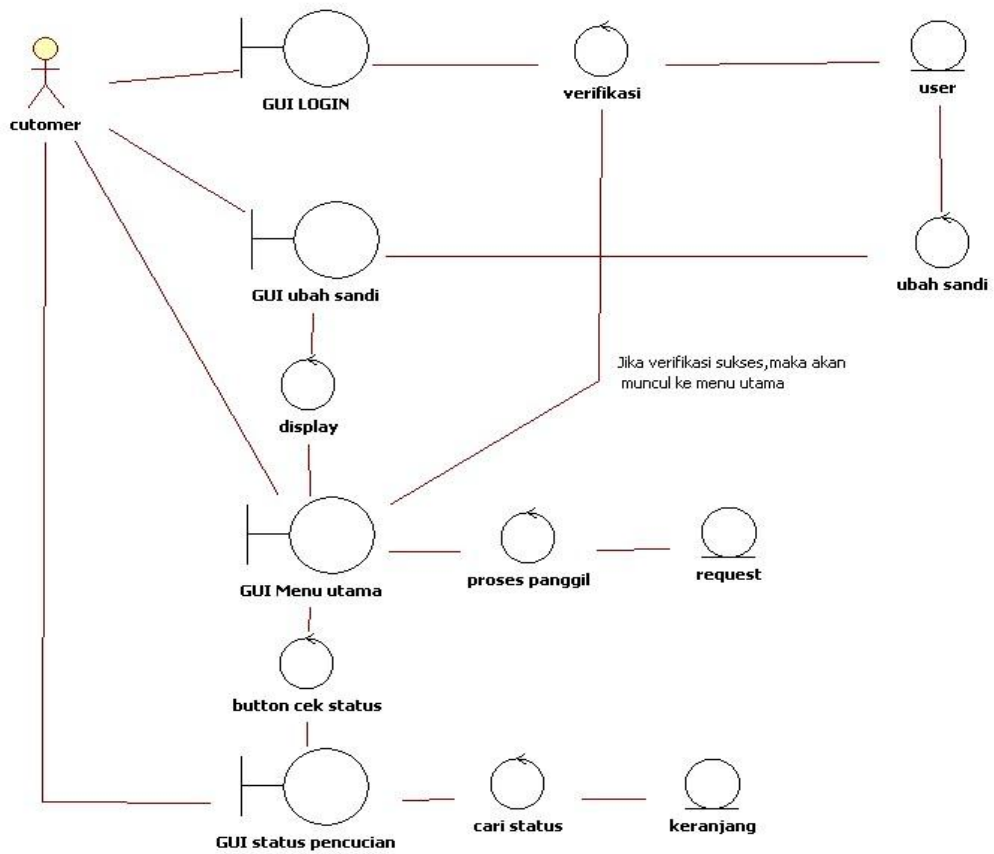
4. User Case Keterhubungan *Client Server*.



Gambar 10 Use case keterhubungan server

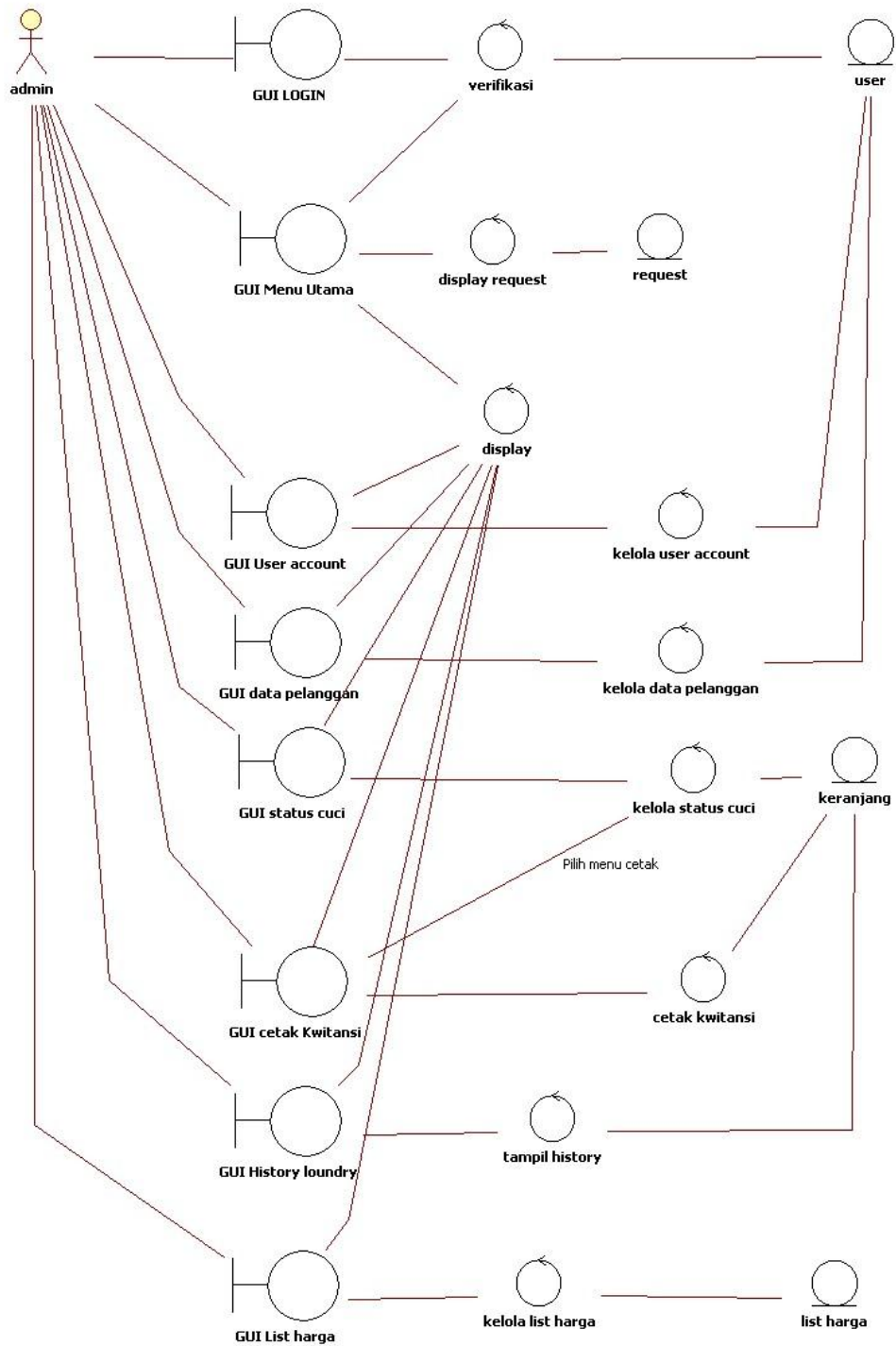
III.1.5 Analisis Class

III.1.5.1 Robustness diagram customer



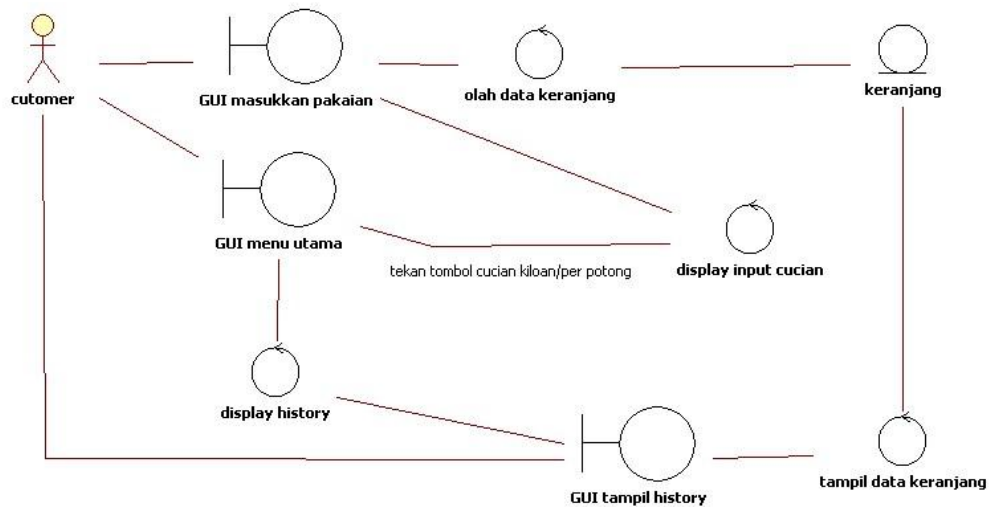
Gambar 11 Robustness diagram customer

III.1.5.2 Robustness diagram admin



Gambar 12 Robustness diagram admin

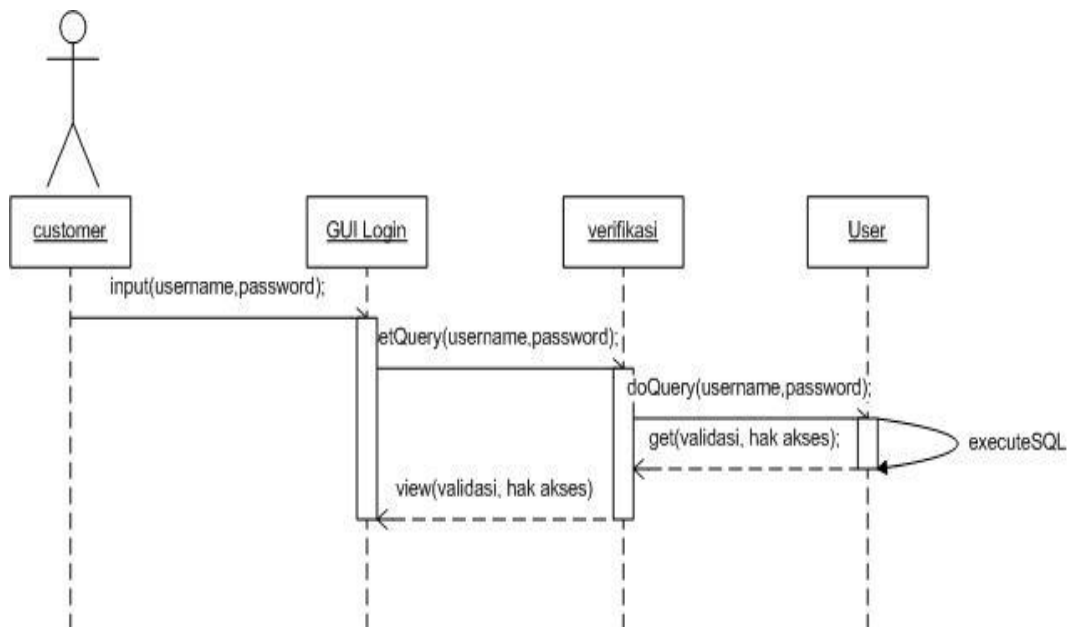
III.1.5.3 Robustness diagram petugas



Gambar 13 Robustness diagram petugas

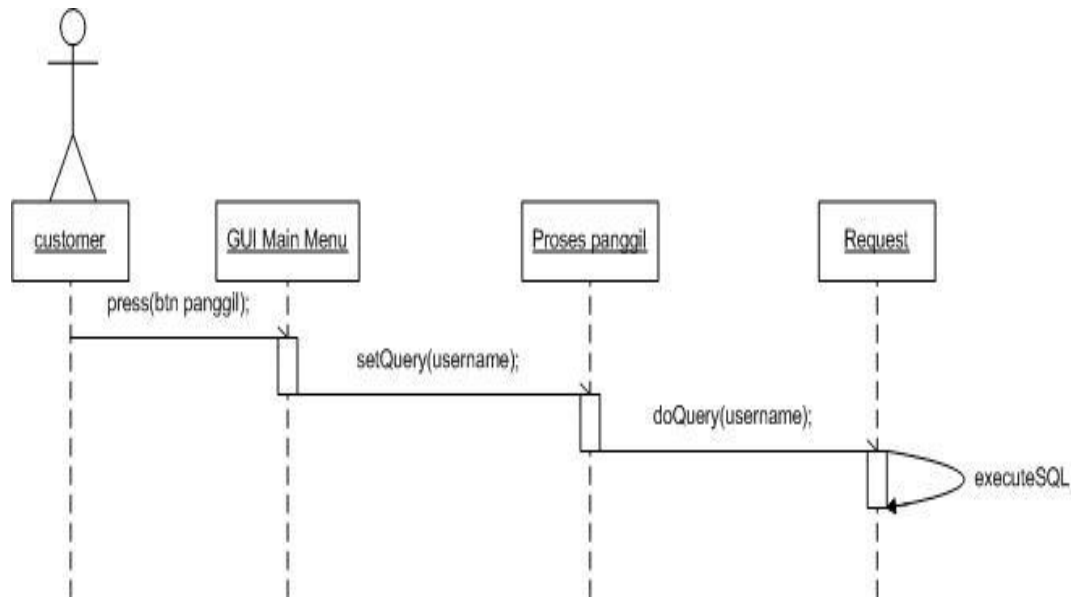
III.1.6 Analisis Sequence Diagram

1. Sequence diagram Login Pelanggan



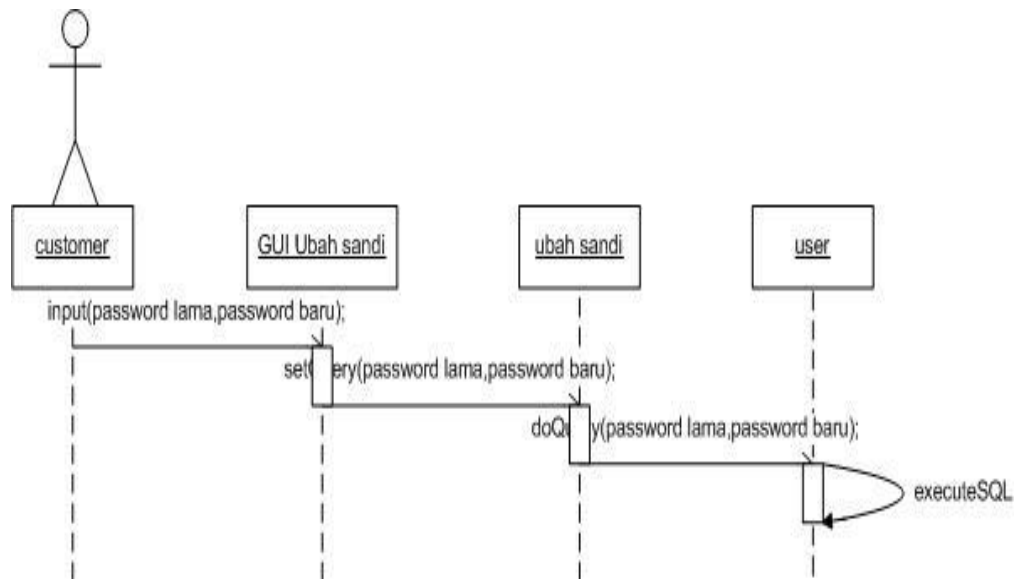
Gambar 14 Sequence diagram login pelanggan

2. Sequence Diagram Request (Pelanggan)



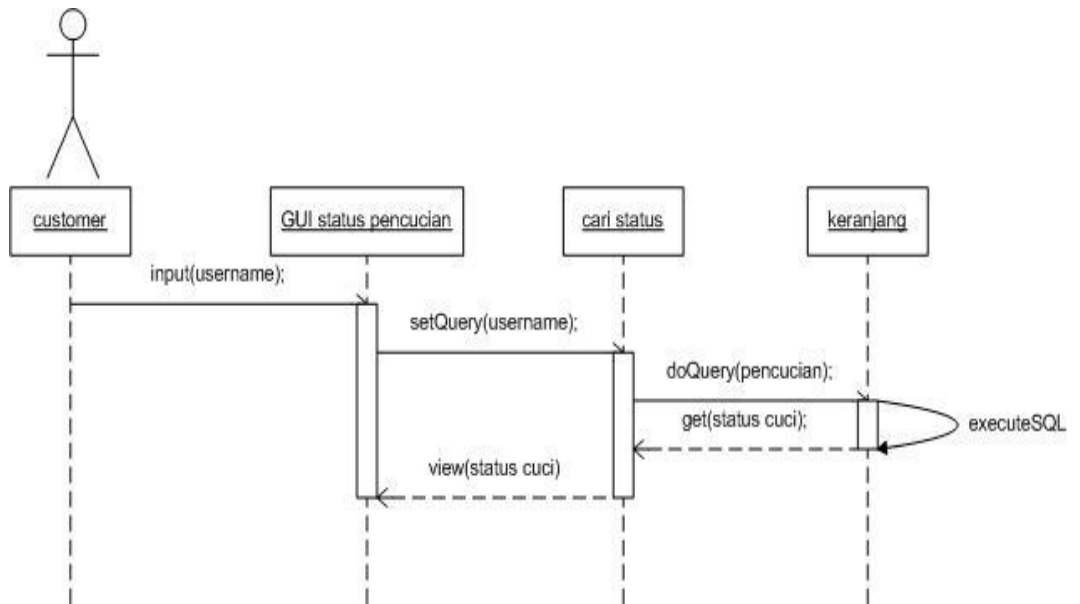
Gambar 15 Sequence diagram request

3. Sequence Diagram Ubah Sandi (Pelanggan)



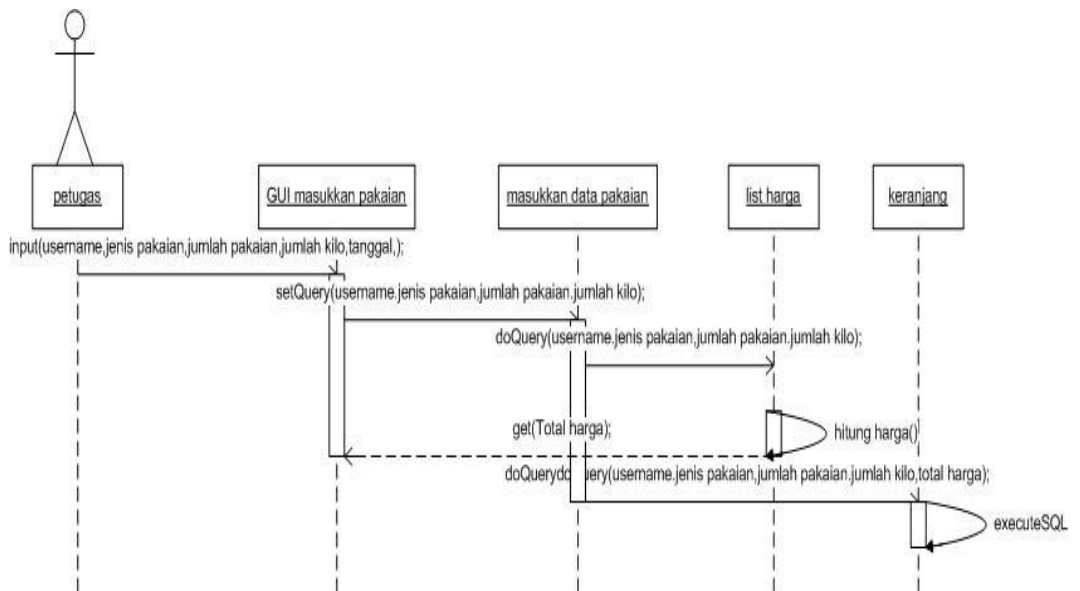
Gambar 16 Sequence diagram ubah sandi

4. Sequence Diagram Status Pencucian (Pelanggan)



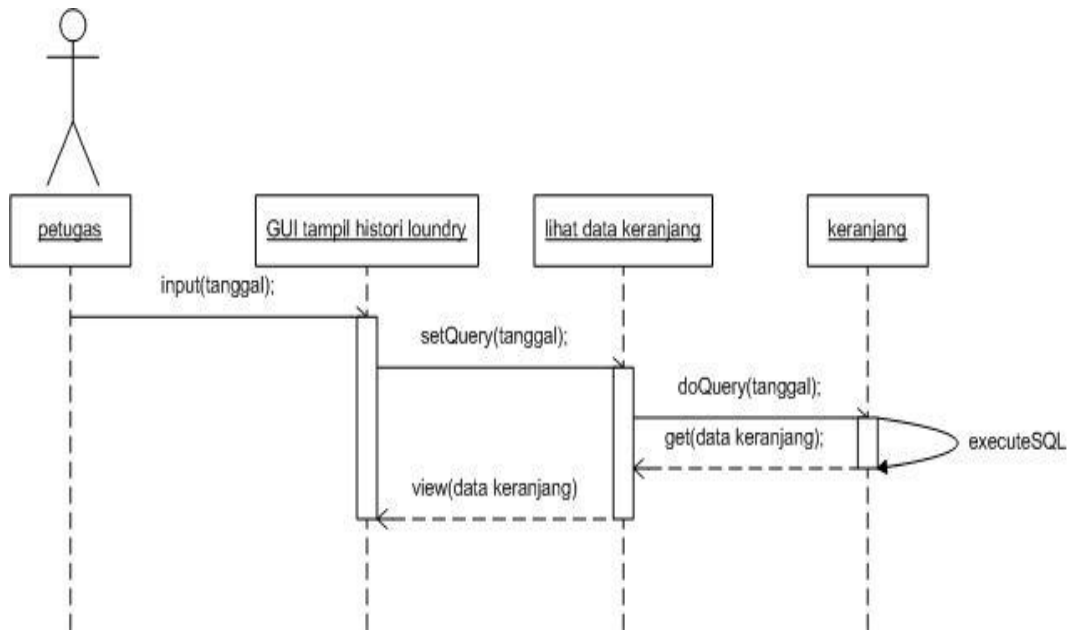
Gambar 17 Sequence diagram tampil status

5. Sequence Diagram Masukkan Pakaian (Petugas Laundry)



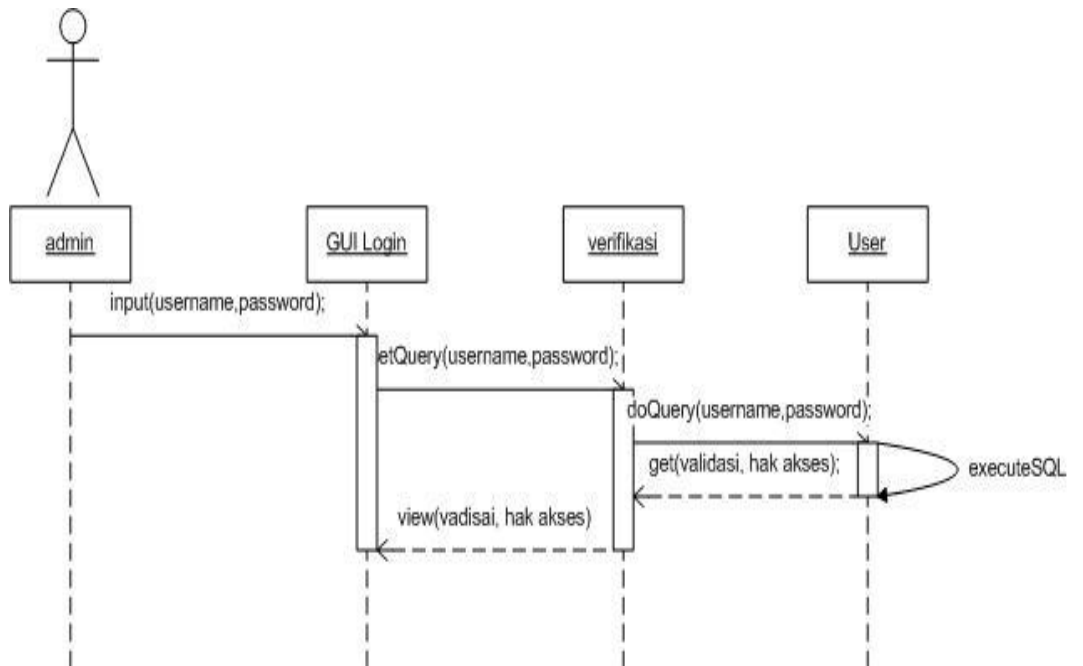
Gambar 18 Sequence diagram masukkan pakaian

6. Sequence Diagram Tampil History Laundry (Petugas Laundry)



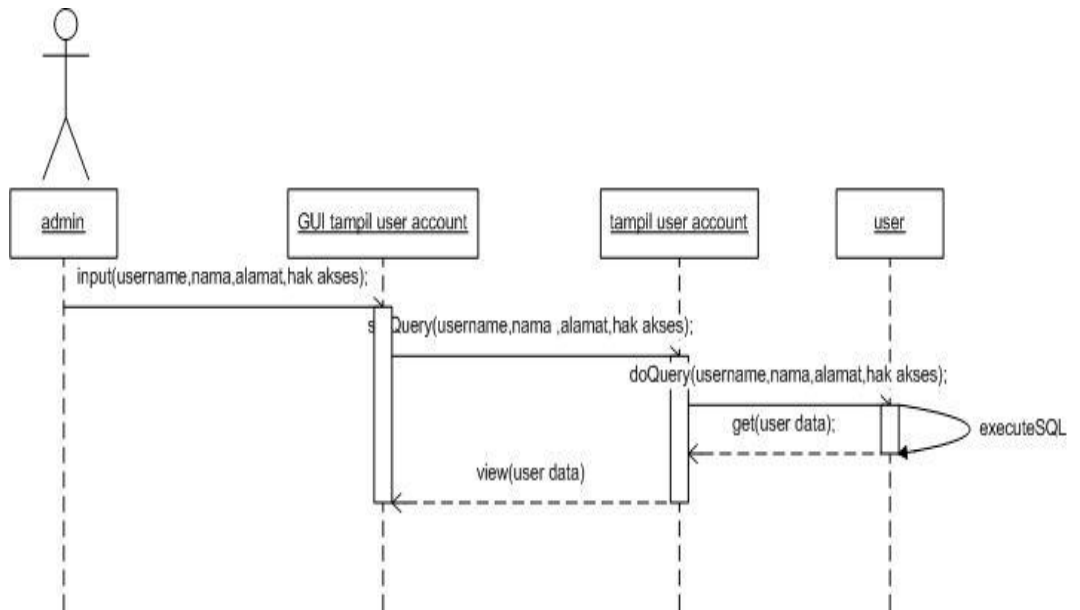
Gambar 19 Sequence diagram tampil history

7. Sequence Diagram Login (Admin)



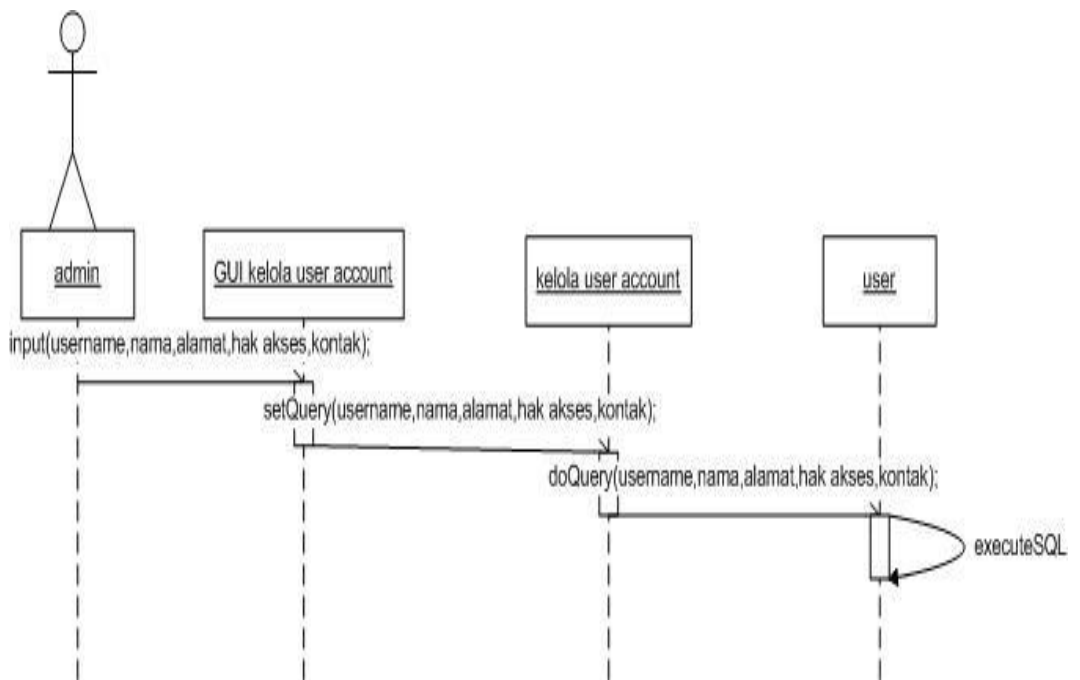
Gambar 20 Sequence diagram login admin

8. Sequence Diagram Menampilkan User Account (Admin)



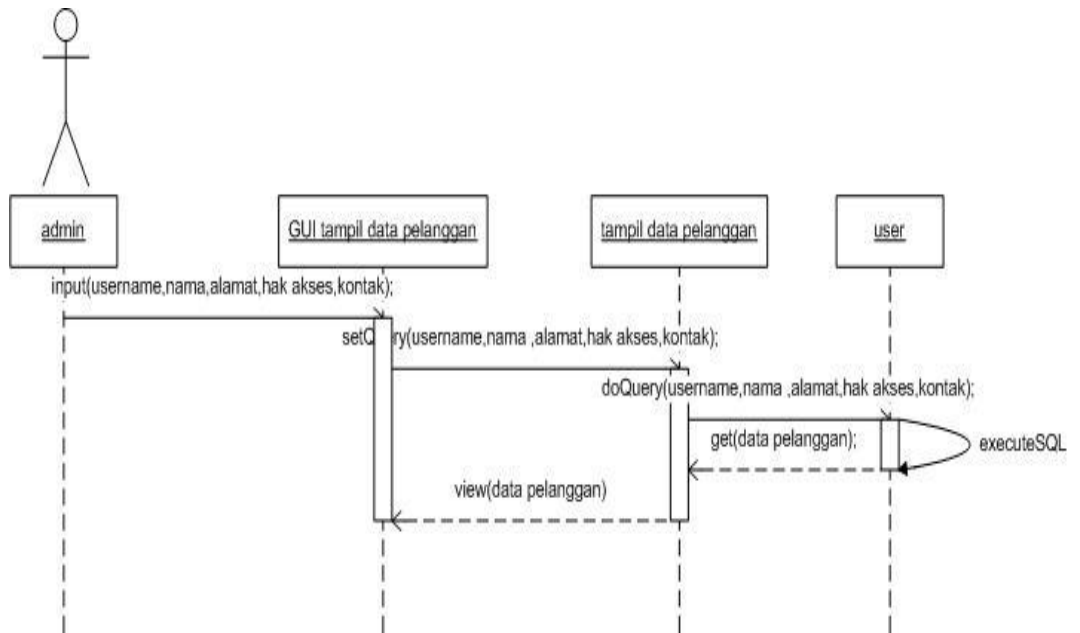
Gambar 21 Sequence diagram tampil user account

9. Sequence Diagram Mengelola User Account (Admin)



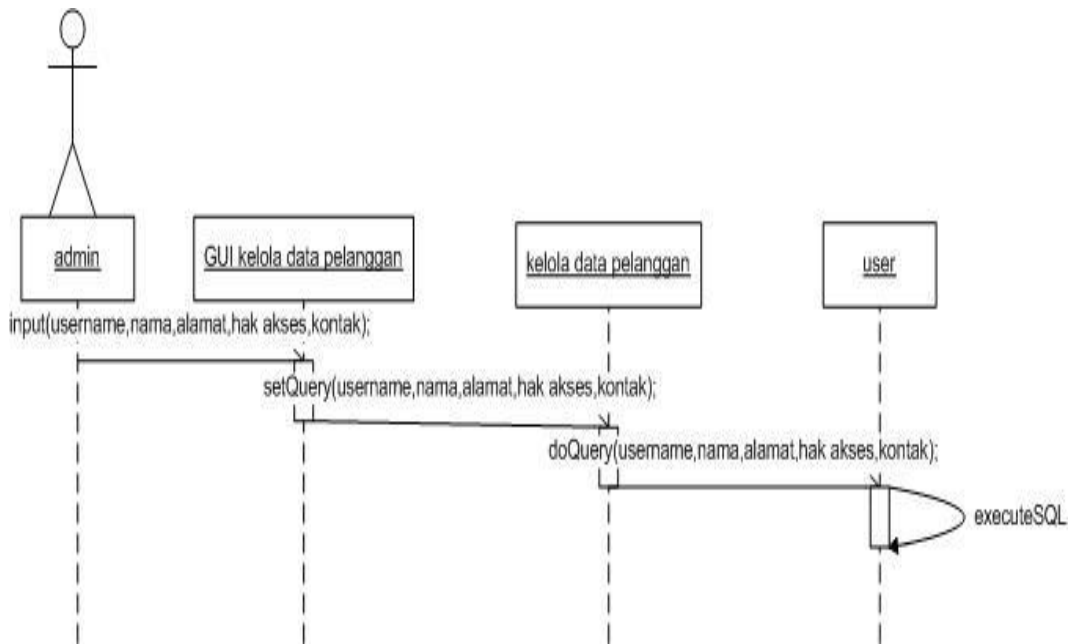
Gambar 22 Sequence diagram kelola user account

10. Sequence Diagram Menampilkan Data Pelanggan (Admin)



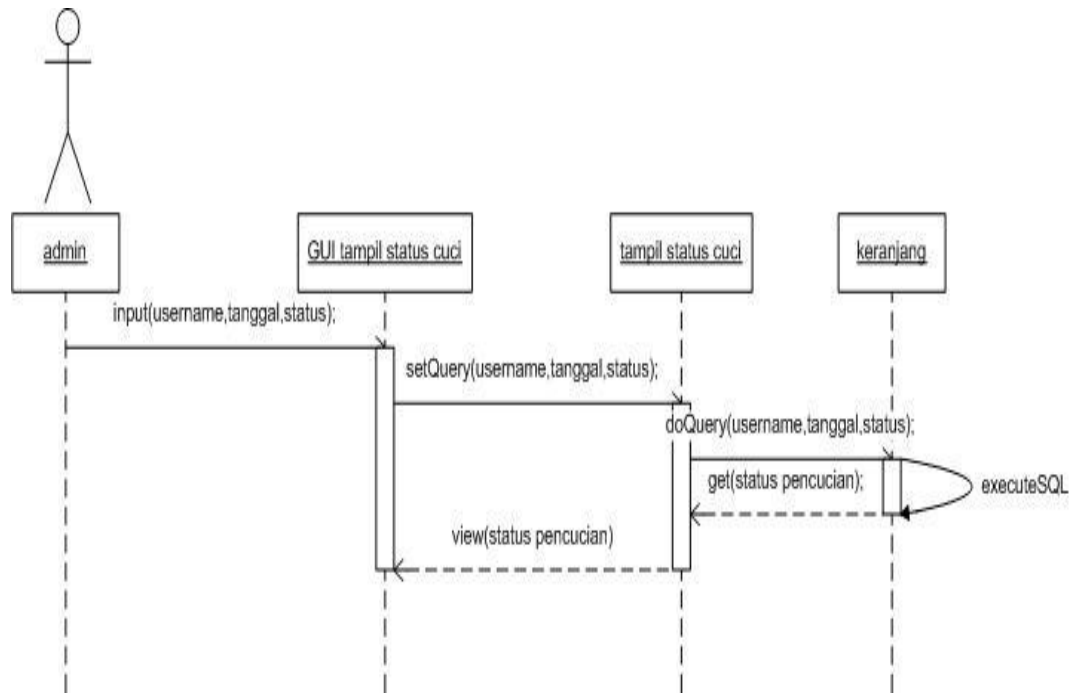
Gambar 23 Sequence diagram tampil data pelanggan

11. Sequence Diagram Mengelola Data Pelanggan (Admin)



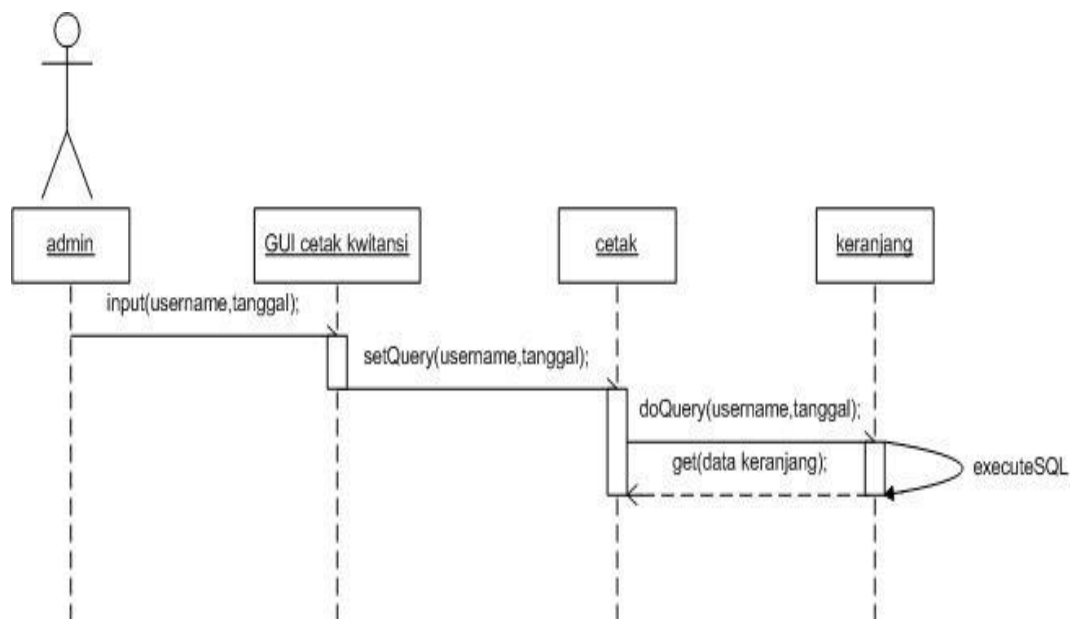
Gambar 24 Sequence diagram kelola data pelanggan

12. Sequence Diagram Menampilkan Status Pencucian (Admin)



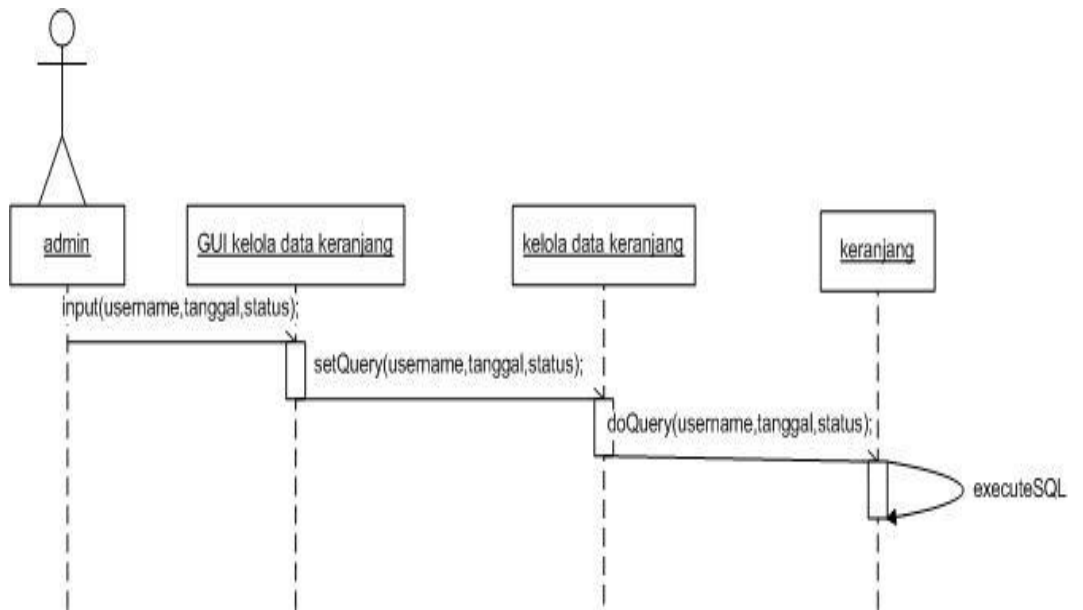
Gambar 25 Sequence diagram tampil status cuci

13. Sequence Diagram Cetak Kwitansi (Admin)



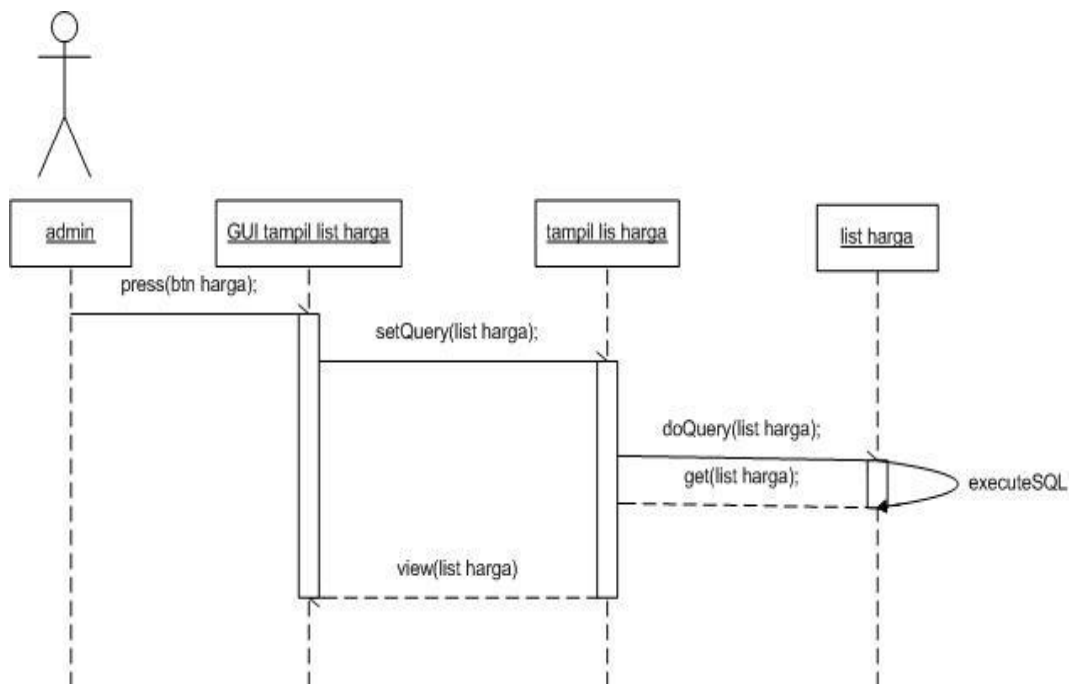
Gambar 26 Sequence diagram cetak kwitansi

14. Sequence Diagram Mengelola Data Keranjang (Admin)



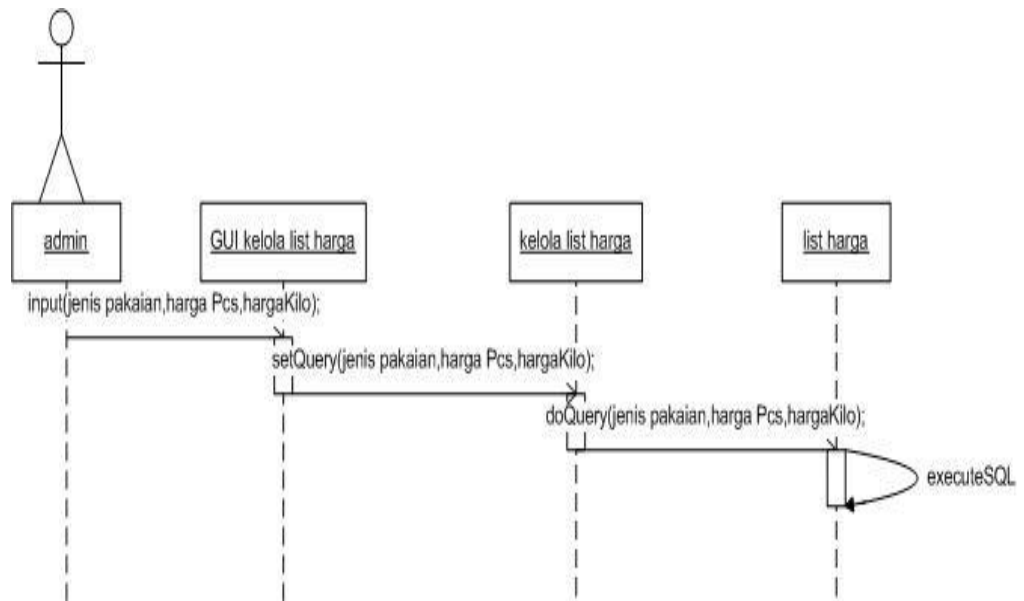
Gambar 27 Sequence diagram kelola data keranjang

15. Sequence diagram menampilkan List Harga Laundry (Admin)



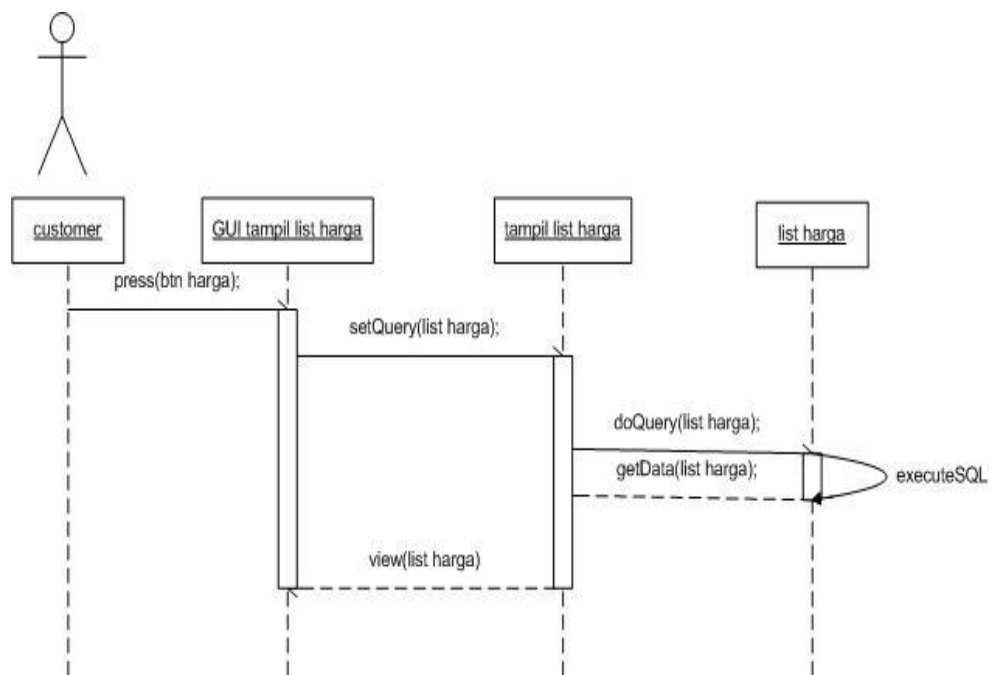
Gambar 28 Sequence diagram tampil list harga

16. Sequence Diagram Mengelola List Harga Laundry (Admin)



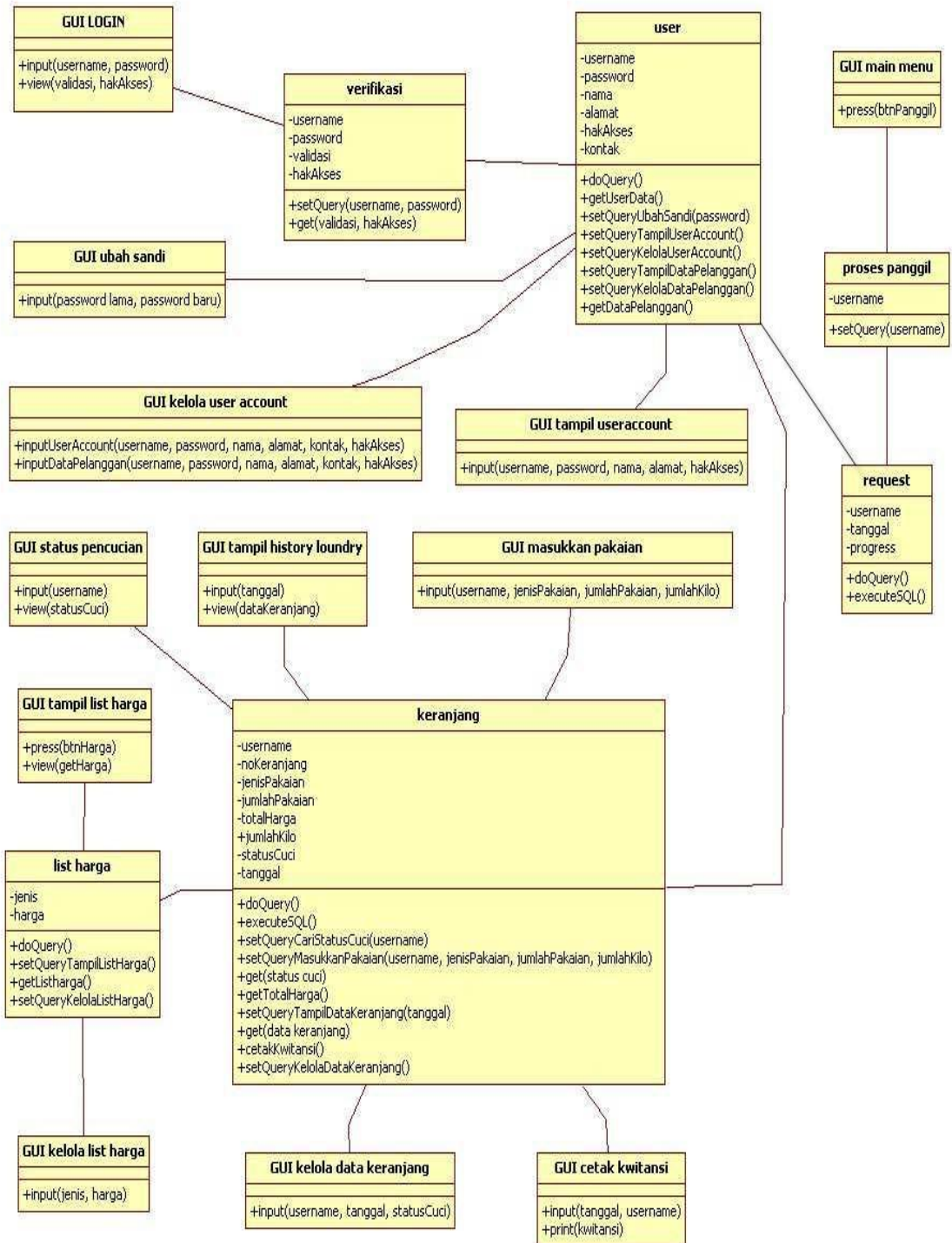
Gambar 29 Sequence diagram kella list harga

17. Sequence diagram menampilkan list harga(Pelanggan)



Gambar 30 Sequence diagram tampil list harga customer

III.1.7 Class Diagram



Gambar 31 Rancangan class diagram

III.2 Perancangan

Pada tahap perancangan, yang dilakukan adalah perancangan algoritma setiap method dari program dan perancangan *interface* program. Interface yang akan dirancang terdiri dari 3 bagian, yaitu *interface* disisi pelanggan (*customer*), *interface* disisi petugas dan *interface* disisi admin.

III.2.1 Perancangan Algoritma

1. Algoritma Verifikasi

```
Initial State : user belum login
Final State  : user telah login

Read username,password
IF Select username = username , password = password
Then
    Read hakAkses
Write {"Login berhasil"}
Else
Write {" Login gagal "}
```

2. Algoritma Ubah Sandi

```
Initial State : sandi belum diganti
Final State  : sandi telah diganti

Read username,password,confirmpass
If confirmpass = password
Then
    If t_user.username = username anf t_user.pasoword = password
    Then
        Update data where t_user.username = username
        Write("Password telah diganti")
```

3. Algoritma tampilkan user account

Initial State : Data user account belum tampil

Final State : Data user account telah tampil

Read username, nama, hakAkses

If select * from t_user

Where username like username or nama like nama or hakAkses =
hakAkses

Then

Write("Tabel User Account")

4. Algoritma kelola user account

Initial State : user account belum dikelola

Final State : admin telah mengubah ataupun menghapus data user account

Read username, password, nama, alamat, kontak, hakAkses

If select * from t_user

Where username like username or nama like nama or hakAkses = hakAkses

Then

If username <> t_user.username

Then

Insert data(username, password, nama, alamat, kontak, hakAkses)

Else

Update data where t_user.username = username

If buttonDelete click

Then

Delete data where t_user.username = username

5. Algoritma lihat data pelanggan

```
Initial State : data pelanggan belum tampil
Final State : Admin mendapat data pelanggan

Read username, nama
If select * from t_user
Where (username like username or nama like nama) and hakAkses =
customer
Then
    Write("Tabel Pelanggan")
```

6. Algoritma kelola data pelanggan

```
Initial State : Data pelanggan belum dikelola
Final State : Data pelanggan telah ubah ataupun di hapus

Read username, password, nama, alamat, kontak
If select * from t_user
Where (username like username or nama like nama) and hakAkses = customer
Then
    If username <> t_user.username
    Then
        Insert data(username, password, nama, alamat, kontak, hakAkses)
    Else
        Update data where t_user.username = username
If buttonDelete click
Then
Delete data where t_user.username = username
```

7. Algoritma Request

```
Initial State : customer belum me-request
Final State : data Request terkirim ke server

Read username
If (select count(*) from t_request where t_request.username = username) = 0
Then
Insert data(username, date.today)
Message("Request Berhasil")
```

8. Algoritma Mencetak Kwitansi

```
Initial State : Admin belum mencetak kwitansi
Final State : Admin mendapat kwitansi

Read username, noKeranjang,tanggal
If select * from t_keranjang
Where username = username or noKeranjang = noKeranjang or tanggal =
tanggal
Then
    hitungTotal()
    Write("Kwitansi")
```

9. Algoritma lihat list harga

```
Initial State : admin belum melihat list harga
Final State : admin telah melihat list harga

If select * from t_harga
Then
    Write("Tabel list harga")
```

10. Algoritma kelola list harga

```
Initial State : data harga belum dikelola
Final State  : data harga telah diubah ataupun dihapus

Read jenisPakaian, harga
If select * from t_harga
Then
    If jenisPakaian <> t_harga.jenisPakaian
    Then
        Insert data(jenisPakaian, harga, kode)
    Else
        Update data where t_harga.jenisPakaian = jenisPakaian
If buttonDelete click
Then
Delete data t_harga.jenisPakaian = jenisPakaian
```

11. Algoritma status pencucian

```
Initial State : admin belum melihat status cuci
Final State  : admin telah melihat status cuci

If select * from t_keranjang
Then
    Write("Tabel Keranjang")
```

12. Algoritma mengelola data keranjang

Initial State : admin belum mengolah data keranjang

Final State : admin telah mengubah data keranjang

If select * from t_keranjang

Then

Write(“Tabel Keranjang”)

If btnEdit.clicked = true

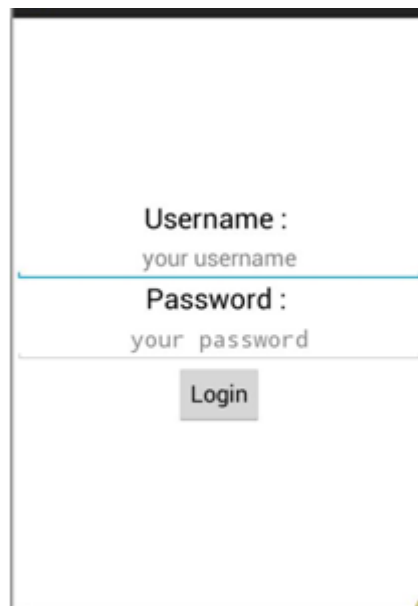
Then

Update t_keranjang set all data where t_keranjang.username =
username

Message(“Data berhasil di update”)

III.2.2 Interface Aplikasi Di Sisi Customer (Pelanggan)

1. Login



The image shows a login form with the following elements:

- Username :** followed by the placeholder text "your username" and a blue horizontal line.
- Password :** followed by the placeholder text "your password" and a grey horizontal line.
- A grey button labeled "Login" centered below the password field.

Gambar 32 Rancangan interface Login pelanggan

2. Menu utama



Gambar 33 Rancangan interface Menu utama pelanggan

3. Status pakaian

No.Ker	tanggal	status
119	2013-07-15	RENDAM
No.Ker	tanggal	status
118	2013-07-15	BELUM DICUCI

Gambar 34 Rancangan interface tampil status pencucian

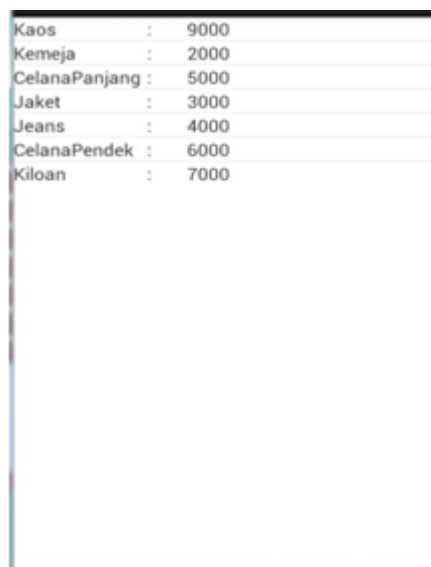
4. Ubah sandi



The image shows a user interface for changing a password. It consists of three input fields, each with a label and a placeholder text. The first field is labeled 'Password Lama :' with the placeholder 'Password Lama Anda'. The second field is labeled 'Password Baru :' with the placeholder 'Password Baru Anda'. The third field is labeled 'Confirm Password :' with the placeholder 'Ulang Pasword Baru Anda'. Below the input fields is a grey button labeled 'SIMPAN'.

Gambar 35 Rancangan interface ubah sandi pelanggan

5. Cek harga



Kaos	:	9000
Kemeja	:	2000
CelanaPanjang	:	5000
Jaket	:	3000
Jeans	:	4000
CelanaPendek	:	6000
Kiloan	:	7000

Gambar 36 Rancangan interface tampil list harga pelanggan

III.2.2.1 Interface disisi petugas laundry

1. Menu utama petugas



Gambar 37 Rancangan interface Menu utama petugas

2. Pilihan jenis pencucian



Gambar 38 Rancangan interface pilihan jenis pencucian

3. Input cucian per Kg

A screenshot of a user interface for entering laundry data per kilogram. The interface is contained within a rectangular border and includes the following fields and labels: 'Username :', 'Nama :', 'Alamat :', 'No. HP :', 'Total Kilo :', and 'Total Harga :'. Each label is followed by a horizontal input line. The 'Username :' field has a vertical cursor at the beginning of the line.

Gambar 39 Rancangan interface masukkan pakaian per Kg

4. Input cucian per Pcs

A screenshot of a user interface for entering laundry data per piece. The interface is contained within a rectangular border and includes the following fields and labels: 'Username :', 'Nama :', 'Masukkan Data Pakaian :', '> Kemeja :', '> Kaos :', '> Jaket :', '> Jeans :', '> C. Panjang :', '> C. Pendek :', 'Jumlah Pakaian :', and 'Total Harga :'. Each label is followed by a horizontal input line. The 'Username :' field has a vertical cursor at the beginning of the line.

Gambar 40 Rancangan interface masukkan pakaian per Pcs

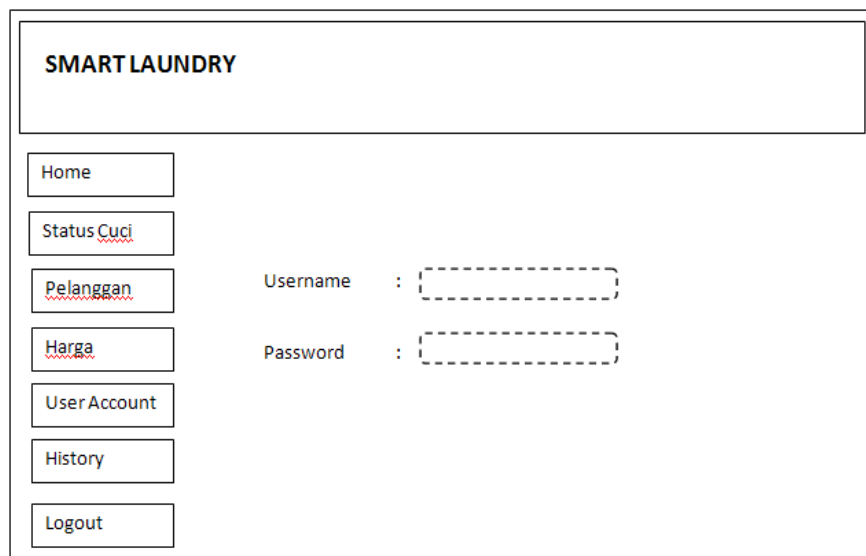
5. History cucian



Gambar 41 Rancangan interface history cucian

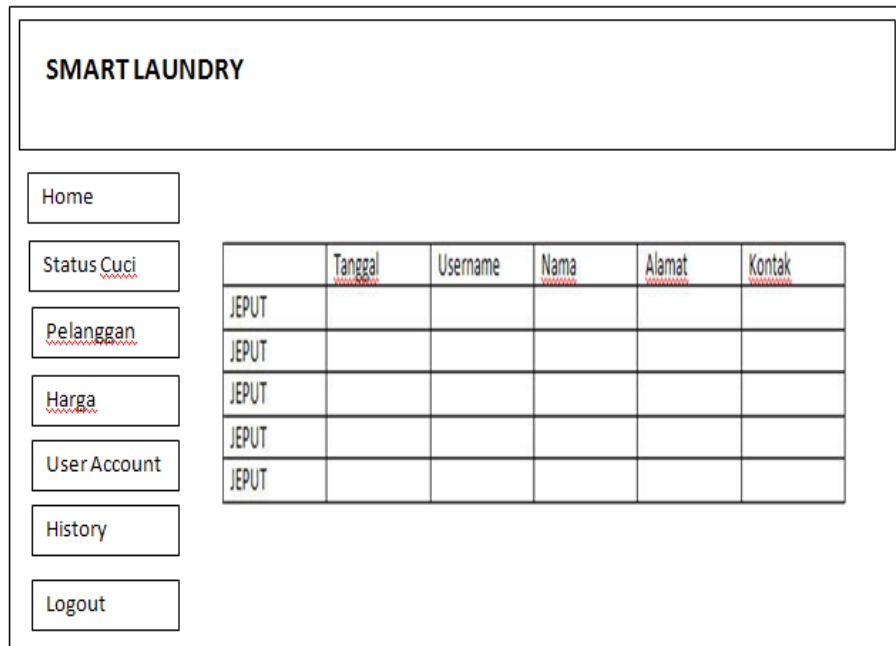
III.2.2.2 Interface disisi Administrator

1. Login



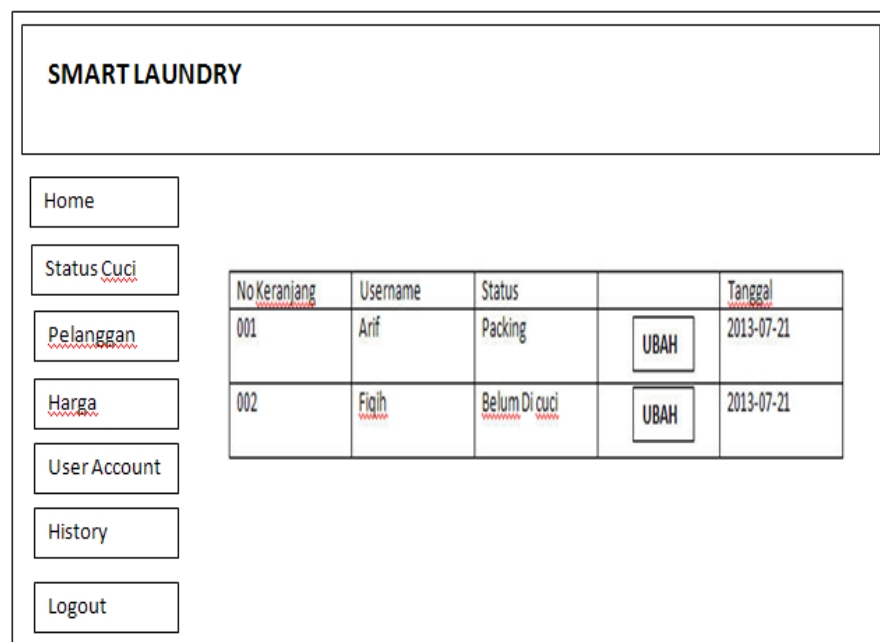
Gambar 42 Rancangan interface login admin

2. Home



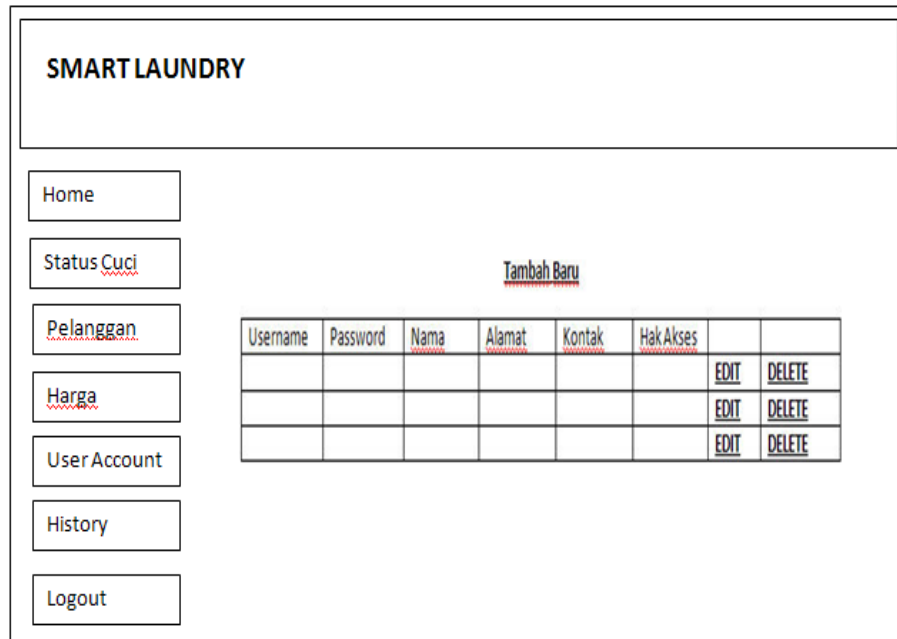
Gambar 43 Rancangan interface home page admin

3. Status cucian



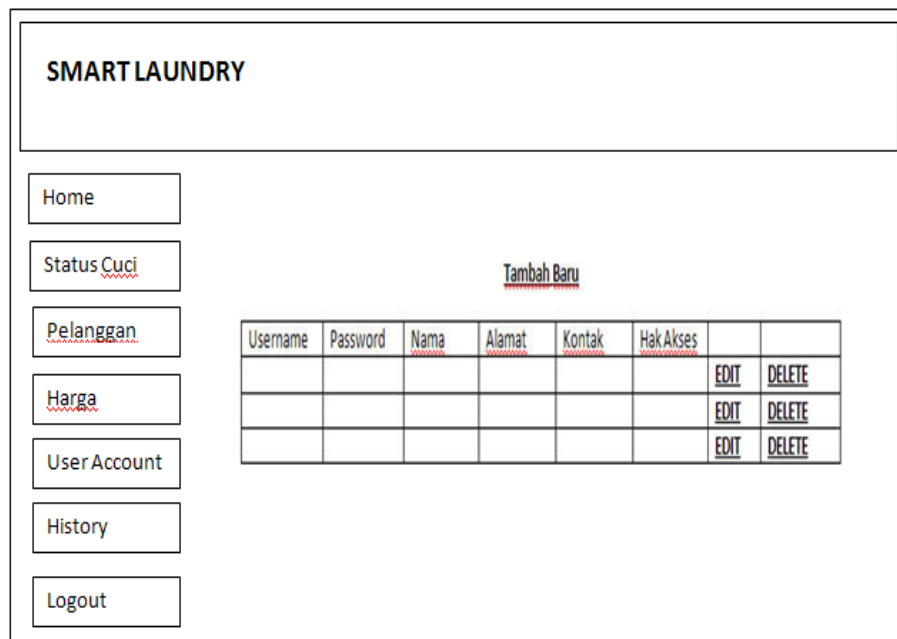
Gambar 44 Rancangan interface status cucian

4. Daftar User account



Gambar 45 Rancangan interface tampil user account

5. Daftar Data Pelanggan



Gambar 46 Rancangan interface tampil data pelanggan

6. Daftar harga

SMART LAUNDRY

- Home
- Status Cuci
- Pelanggan
- Harga
- User Account
- History
- Logout

Jenis Pakaian	Harga

Tambah **SIMPAN**

Gambar 47 Rancangan interface tampil daftar harga

7. Cetak kwitansi

SMART LAUNDRY

- Home
- Status Cuci
- Pelanggan
- Harga
- User Account
- History
- Logout

KWITANSI SMART LAUNDRY	
Nama : Arif ravindra	16-07-2013
Alamat : Bengkong	
Jenis Pakaian	Jumlah
Kiloan	3
Total Harga : Rp.21.000	
Tertanda	
SMART LAUNDRY	

Gambar 48 Rancangan interface cetak kwitansi

8. History laundry

SMART LAUNDRY

Home

Status Cuci

Pelanggan

Harga

User Account

History

Logout

No Keranjang	Username	Status	Tanggal
001	Sofian12	SELESAI	2013-06-25
003	UlfaCakep	SELESAI	2013-06-25

Gambar 49 Rancangan interface tampil history laundry

9. Olah data pelanggan

SMART LAUNDRY

Home

Status Cuci

Pelanggan

Harga

User Account

History

Logout

Username	:	Pelanggan123
Password	:	Pelanggan
Nama	:	Pelanggan
Alamat	:	Batam
Kontak	:	0983763553
SUBMIT		

Gambar 50 Rancangan interface kelola data pelanggan

Bab IV Implementasi dan Pengujian

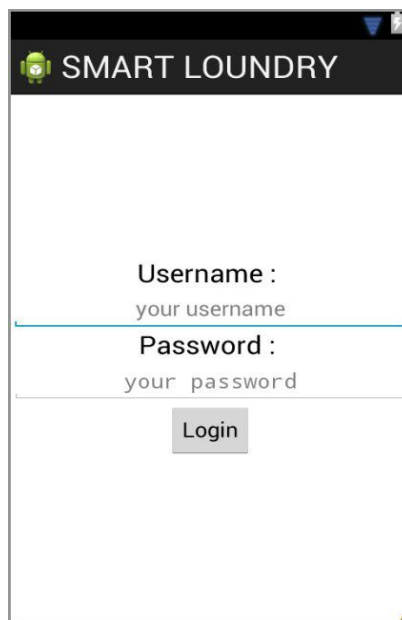
Setelah proses analisis dan perancangan dilakukan, maka tahap selanjutnya yaitu mengimplementasikan rancangan tersebut dalam bentuk program atau aplikasi yang sudah jadi dan pembuatan hasil uji program tersebut. Pada bab ini akan dipaparkan hasil implementasi aplikasi Smart laundry.

IV.1 Implementasi *interface* (Antarmuka)

Pada proses implementasi antar muka, yang akan di implementasikan yaitu antarmuka aplikasi pada *customer*, *petugas* dan *admin*.

IV.1.1 *Interface customer* (pelanggan)

IV.1.1.1 Login pelanggan



Gambar 51 Interface login pelanggan

IV.1.1.2 Menu utama pelanggan



Gambar 52 Interface menu utama pelanggan

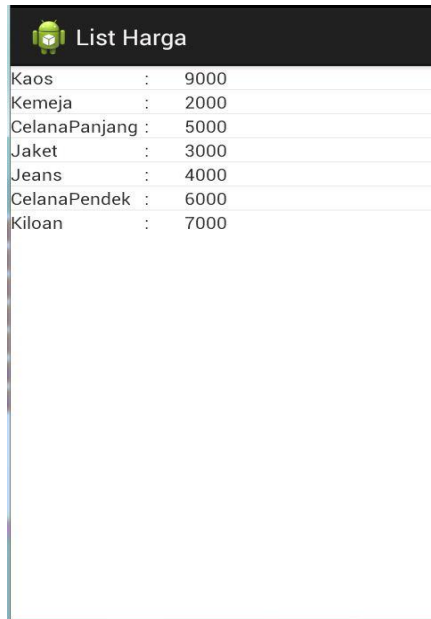
IV.1.1.3 Cek status cuci pelanggan

The screenshot shows the "Status Pencucian" interface. At the top, there is a black header bar with the Android logo and the text "Status Pencucian". Below the header, there is a table with three columns: "No.Ker", "tanggal", and "status". The table contains two rows of data.

No.Ker	tanggal	status
119	2013-07-15	RENDAM
118	2013-07-15	BELUM DICUCI

Gambar 53 Interface status cuci pelanggan

IV.1.1.4 Cek harga



List Harga	
Kaos	: 9000
Kemeja	: 2000
Celana Panjang	: 5000
Jaket	: 3000
Jeans	: 4000
Celana Pendek	: 6000
Kiloan	: 7000

Gambar 54 Interface tampil list harga

IV.1.1.5 Ubah sandi



SMART LOUNDRY

Password Lama :
Password Lama Anda

Password Baru :
Password Baru Anda

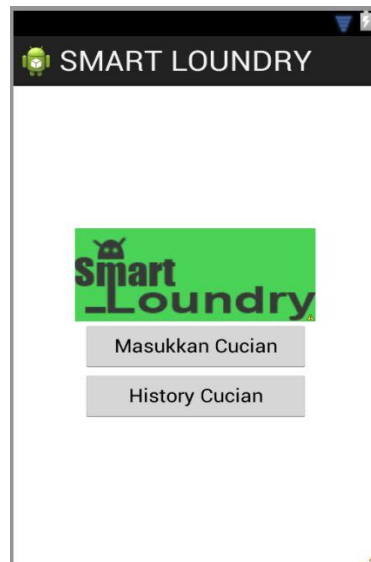
Confirm Password :
Ulang Password Baru Anda

SIMPAN

Gambar 55 Interface ubah sandi pelanggan

IV.1.2 *Interface* petugas

IV.1.2.1 Menu utama



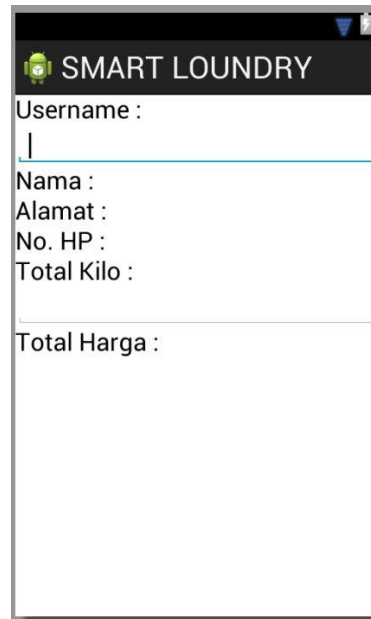
Gambar 56 Interface menu utama petugas

IV.1.2.2 Pilihan Masukkan Cucian



Gambar 57 Interface pilihan pencucian

IV.1.2.3 Masukkan cucian per Kg



The screenshot shows the 'SMART LOUNDRY' app interface. At the top, there is a header with the app name and a logo. Below the header, there are several input fields: 'Username :', 'Nama :', 'Alamat :', 'No. HP :', 'Total Kilo :', and 'Total Harga :'. The 'Total Kilo' field is currently empty, and the 'Total Harga' field is also empty.

Gambar 58 Interface masukkan cucian per Kg

IV.1.2.4 Masukkan cucian per Pcs



The screenshot shows the 'SMART LOUNDRY' app interface. At the top, there is a header with the app name and a logo. Below the header, there are several input fields: 'Username :', 'Nama :', 'Masukkan Data Pakaian :', '> Kemeja :', '> Kaos :', '> Jaket :', '> Jeans :', '> C. Panjang :', '> C. Pendek :', 'Jumlah Pakaian :', and 'Total Harga :'. The 'Masukkan Data Pakaian' section contains a list of clothing items with corresponding input fields for their quantity. The 'Total Harga' field is currently empty.

Gambar 59 Interface masukkan cucian per Pcs

IV.1.2.5 Lihat history laundry



Gambar 60 Interface tampil history laundry

IV.1.2.6 Interface isi keranjang



Gambar 61 Interface isi keranjang

IV.1.3 Interface Admin

IV.1.3.1 Login



Gambar 62 Interface login admin

IV.1.3.2 Home (Request)



Gambar 63 Interface home page admin

IV.1.3.3 Status Cuci

No Keranjang	Username	Status		Tanggal
120	fiqih	BELUM DICUCI	Ubah	2013-07-15
118	arif	BELUM DICUCI	Ubah	2013-07-15

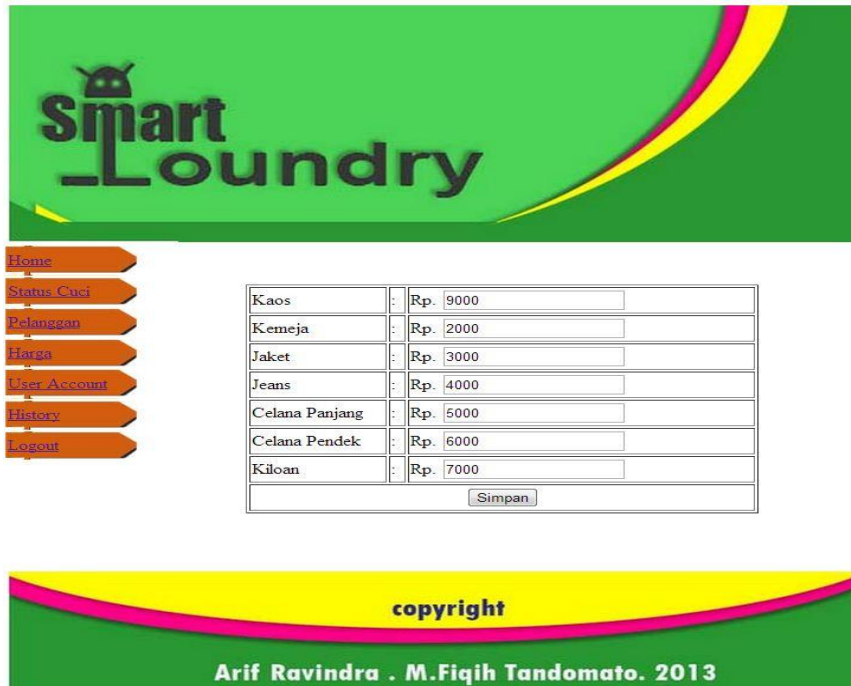
Gambar 64 Interface tampil status cuci

IV.1.3.4 Pelanggan

username	Password	Nama	Alamat	Kontak	Hak Akses		
arif	321	arif ravindra	bengkong	0811701892	PELANGGAN	EDIT	DELETE

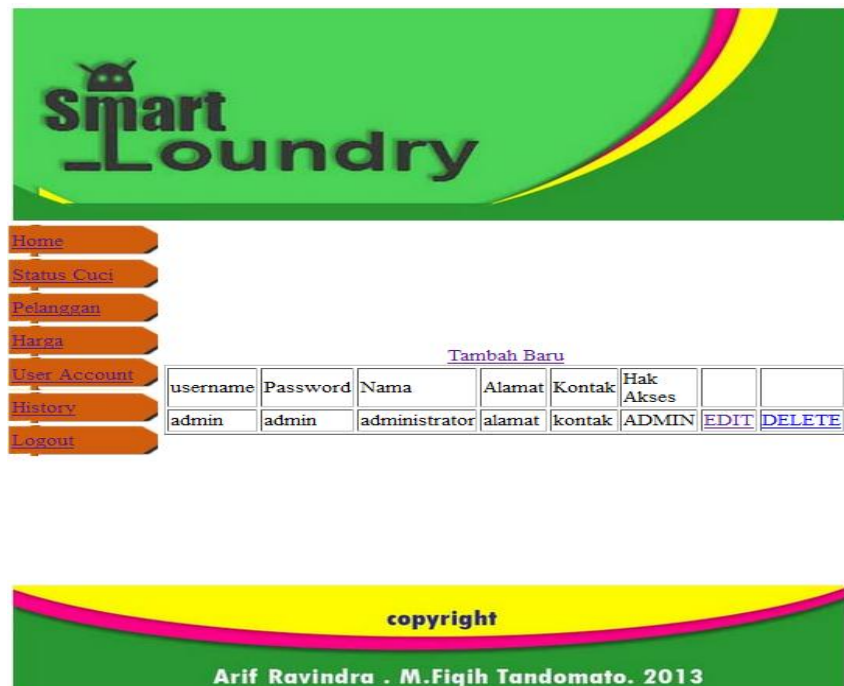
Gambar 65 Interface tampil data pelanggan

IV.1.3.5 Harga



Gambar 66 Interface tampil list harga

IV.1.3.6 User Account



Gambar 67 Interface tampil user account

IV.1.3.7 History

No Keranjang	Username	Status	Tanggal
119	arif	SELESAI	2013-07-15

Gambar 68 Interface tampil history

IV.1.3.8 Cetak Kwitansi

KWITANSI SMART LOUNDRY	
Nama : arif ravindra	16-07-2013
Alamat : bengkong	
Jenis Pakaian	Jumlah
Kiloan	3
Total harga : Rp.21000	
Tertanda :	
	
Print	

Gambar 69 Interface cetak kwitansi

IV.1.3.9 Kelola Data Pelanggan

The screenshot displays the 'Smart Laundry' web application interface for managing customer data. The page has a green header with the 'Smart Laundry' logo. On the left, there is a vertical navigation menu with the following items: Home, Status Cuci, Pelanggan, Harga, User Account, History, and Logout. The main content area contains a form for adding customer data with the following fields and values:

Username	: pelanggan
Password	: pelanggan
Nama	: nama pelanggan
Alamat	: alamat pelanggan
Kontak	: 08xxx
<input type="button" value="Submit"/>	

At the bottom of the page, there is a yellow footer with the text 'copyright'.

Gambar 70 Interface kelola data pelanggan

IV.2 Hasil Uji

Tabel 6 Hasil uji aplikasi

No	Kelas	Skenario	Data Uji	Target	Pengujian	
					Benar	Salah
1	Login Pelanggan	Pelanggan sebagai pemilik <i>username</i> dan <i>password</i> Pelanggan memasukkan <i>username</i> dan <i>password</i> yang mereka miliki	<p>Data Benar</p> <p><i>Username</i> dan <i>Password</i> benar. Hak Akses sebagai Pelanggan.</p> <p>Data Salah</p> <p>Login gagal.</p>	<p><i>Main Menu</i> tampil.</p> <p><i>Main Menu</i> tidak tampil.</p>		
2	<i>Request</i>	Pelanggan membuka aplikasi dan masuk ke <i>main menu</i> . Pelanggan	<p>Data Benar</p> <p>Username masuk ke basis data.</p> <p>Data Salah</p>	<p>Pesan berhasil keluar. Data dimasukkan ke basis data.</p>		

No	Kelas	Skenario	Data Uji	Target	Pengujian	
					Benar	Salah
		menekan tombol <i>request</i> .	Data tidak dimasukkan.	Pesan gagal keluar.		
3	Ubah Sandi	Pelanggan memasukkan <i>password</i> lama dan <i>Password</i> baru. Pelanggan menekan tombol simpan.	Data Benar Data sandi di table user diubah. Data Salah Data tidak dimasukkan.	Pesan berhasil ditampilkan. Data dimasukkan ke basis data. Pesan gagal keluar.		
4	Status Pencucian	Petugas menekan tombol status pencucian.	Data Benar Data status cuci terpilih berdasarkan <i>username</i> .	Data status pencucian tampil dalam bentuk table.		

No	Kelas	Skenario	Data Uji	Target	Pengujian	
					Benar	Salah
			Data Salah Tidak ada data yang terpilih.	Pesan gagal keluar.		
5	Masukkan Pakaian	Petugas memasukkan data pakaian. Pelanggan menekan tombol simpan.	Data Benar Data diisi dengan benar dan masuk ke dalam basis data. Data Salah Data tidak masuk ke basis data.	Pesan berhasil ditampilkan. Pesan gagal ditampilkan.		
6	<i>History Laundry</i>	Petugas menekan tombol lihat history.	Data Benar Data history terpilih. Data Salah Data tidak ada yang terpilih.	Data history ditampilkan dalam bentuk table. Pesan gagal ditampilkan.		

No	Kelas	Skenario	Data Uji	Target	Pengujian	
					Benar	Salah
7	Login Admin	Admin sebagai pemilik <i>username</i> dan <i>password</i> Admin memasukkan <i>username</i> dan <i>password</i> yang mereka miliki	<p>Data Benar</p> <p><i>Username</i> dan <i>Password</i> benar. Hak Akses sebagai Admin.</p> <p>Data Salah</p> <p>Login gagal.</p>	<p><i>Main Menu</i> tampil.</p> <p><i>Main Menu</i> tidak tampil.</p>		
8	Menampilkan <i>user account</i> .	Admin memilih menu <i>user account</i> .	<p>Data Benar</p> <p>Data <i>user account</i> terpilih.</p>	Data <i>user account</i> ditampilkan dalam bentuk		

No	Kelas	Skenario	Data Uji	Target	Pengujian	
					Benar	Salah
			Data Salah Tidak ada data yang terpilih.	tabel. Tidak ada data yang tampil.		
9	Mengelola <i>user account</i> .	Admin memilih memasukkan data baru atau memilih data yang akan diolah.	Data Benar Data <i>user account</i> diisi dengan benar. Data <i>user account</i> masuk ke basis data. Data Salah Data tidak diisi dengan benar. Data tidak masuk ke basis data.	Pesan berhasil ditampilkan. Pesan gagal ditampilkan.		
10	Menampilkan data pelanggan.	Admin memilih menu pelanggan	Data Benar Data pelanggan terpilih. Data Salah	Data <i>user account</i> ditampilkan dalam bentuk tabel.		

No	Kelas	Skenario	Data Uji	Target	Pengujian	
					Benar	Salah
			Tidak ada data yang terpilih.	Tidak ada data yang tampil.		
11	Mengelola data pelanggan.	Admin memilih memasukkan data baru atau memilih data yang akan diolah.	<p>Data Benar Data pelanggan diisi dengan benar. Data pelanggan masuk ke basis data.</p> <p>Data Salah Data tidak diisi dengan benar. Data tidak masuk ke basis data.</p>	<p>Pesan berhasil ditampilkan.</p> <p>Pesan gagal ditampilkan.</p>		
12	Status Pencucian ADMIN	Admin menekan tombol status pencucian.	<p>Data Benar Data status cuci terpilih berdasarkan <i>username</i>.</p> <p>Data Salah</p>	Data ditampilkan dalam bentuk tabel.		

No	Kelas	Skenario	Data Uji	Target	Pengujian	
					Benar	Salah
			Tidak ada data yang terpilih.	Pesan gagal ditampilkan.		
13	Cetak Kwitansi	Admin menekan tombol cetak.	<p>Data Benar Data total harga diambil dari tabel keranjang.</p> <p>Data Salah Tidak ada data yang diambil.</p>	<p>Kwitansi dicetak.</p> <p>Kwitansi tidak dicetak.</p>		
14	Mengelola data keranjang.	Admin memilih ubah data, tambah data, maupun hapus data.	<p>Data Benar Data keranjang disini dengan benar dan diolah.</p> <p>Data Salah Data tidak ada yang diolah.</p>	<p>Pesan berhasil ditampilkan.</p> <p>Pesan gagal ditampilkan.</p>		
15	Menampilkan list harga laundry.	Admin memilih menu harga.	<p>Data benar Data list harga terpilih.</p>	Data harga laundry ditampilkan dalam bentuk		

No	Kelas	Skenario	Data Uji	Target	Pengujian	
					Benar	Salah
			Data salah Tidak ada data yang terpilih.	table. Pesan gagal ditampilkan.		
17	Mengelola list harga.	Admin memilih data list harga yang akan diubah. Admin mengubah data. Admin menekan tombol simpan.	Data benar Data list harga dimasukkan dengan benar. Data list diolah. Data salah Tidak ada data yang diolah.	Pesan berhasil ditampilkan. Pesan gagal ditampilkan.		

Bab V Kesimpulan dan Saran

V.1 Kesimpulan

1. Aplikasi dapat menyediakan informasi proses *laundry* untuk pelanggan.
2. Aplikasi ini dapat mengirimkan pesan ke komputer server yang berisi pemanggilan petugas jemput pakaian.
3. Aplikasi dapat memberikan informasi harga terkini.

V.2 Saran

Setelah melakukan pengujian serta evaluasi terhadap aplikasi ini, maka penulis berharap aplikasi ini dapat dikembangkan dengan saran-saran pengembangan sebagai berikut:

1. Pembuatan program dapat juga menggunakan teknik parsing data JSON+REST

DAFTAR PUSTAKA

1. Arisandi, Dedi , 2011, *Use Case Diagram*. Jurnal Pembelajaran APSI
2. Developer, Android . ADT Plugin
<http://developer.android.com/tools/sdk/eclipse-adt.html>
Diakses pada tanggal 13 Februari 2013
3. Developer,Android, Get the android SDK
<http://developer.android.com/sdk/index.html>
Diakses pada tanggal 15 Februari 2013
4. Madcoms, 2013, *Aplikasi web database dengan dreamweaver dan php-mysql*, Yogyakarta, Andi.
5. Organisation, JSON, Introduction JSON
<http://www.json.org/json-id.html>
Diakses pada tanggal 3 Juli 2013
6. Safaat, Nazruddin, 2012 *Pemrograman aplikasi mobile smartphone dan tablet pc berbasis android*, Bandung, Informatika bandung.
7. Sianipar,RH, 2013, *Teori dan implementasi Java*, Bandung, Informatika bandung.
8. Siddik, Mahmud. Compile dan run file java
<http://mahmudsiddik.com/download/materi/java/166-compile-dan-run-file-java-di-command-prompt.html>
Diakses pada tanggal 24 April 2013
9. Siti syamsiah, 2009, *Mengenal Pemrograman HTML*, Makalah Pemrograman, Teknik Informatika BSI.
10. ST, Wina Noviani Fatimah, 2011, *Pengenalan Eclipse*, Jurnal pengenalan dan installasi eclipse
11. Virtual hotel institute, Laundry department
<http://virtualhotelinstitute.com/page/40262/laundry-department.html>
Diakses pada tanggal 10 Februari 2013

12. Wikipedia, Unfied Modeling Language

http://id.wikipedia.org/wiki/Unified_Modeling_Language

Diakses pada tanggal 6 Mei 2013