PERANCANGAN APLIKASI LAUNDRY MENGGUNAKAN JAVA

PROJECT WORK

*Diajukan Sebagai Salah Satu Syarat Untuk Menyelesaikan Mata Pelajaran Pemograman Berorientasi Objek Di SMK Negeri 2 GUGUAK*



OLEH :

Nama : Al Azim

NIS : 170700714

PROGRAM STUDI KEAHLIAN TEKNIK KOMPUTER

DAN INFORMATIKA

KOMPETENSI KEAHLIAN REKAYASA PERANGKAT LUNAK

SMK NEGERI 2 KECAMATAN GUGUAK

KABUPATEN LIMAPULUH KOTA

SUMATERA BARAT

2020

BAB I

LANDASAN TEORI

* 1. Konsep Dasar
     1. Program

Menurut KBBI (Kamus Besar Bahasa Indonesia) menjelaskan bahwa program adalah “rancangan mengenai asas serta usaha yang akan dijalankan atau urutan perintah yang diberikan pada komputer untuk membuat fungsi atau tugas tertentu”.

Beberapa langkah yang harus diperhatikan oleh programmer sebelum mengimplementasikan suatu permasalahan dalam program yaitu:

1. Mendefinisikan masalah
2. Mencari solusi untuk masalah
3. Memilih teknik pemecahan masalah dan algoritma
4. Menulis program
5. Melakukan test dan debugging
6. Melakukan dokumentasi
7. Melakukan pemeliharaan
   * 1. Aplikasi

Aplikasi perangkat lunak (bahasa Inggris: software application) adalah suatu subkelas perangkat lunak komputer yang memanfaatkan kemampuan komputer langsung untuk melakukan suatu tugas yang diinginkan pengguna. Biasanya dibandingkan dengan perangkat lunak sistem yang mengintegrasikan berbagai kemampuan komputer, tetapi tidak secara langsung menerapkan kemampuan tersebut untuk mengerjakan suatu tugas yang menguntungkan pengguna.

* 1. Teori Program
     1. Java

Java adalah bahasa pemrograman yang multi platform dan multi device. Sekali anda menuliskan sebuah program dengan menggunakan Java, anda dapat menjalankannya hampir di semua komputer dan perangkat lain yang support Java, dengan sedikit perubahan atau tanpa perubahan sama sekali dalam kodenya. Aplikasi dengan berbasis Java ini dikompulasikan ke dalam p-code dan bisa dijalankan dengan Java Virtual Machine. Fungsionalitas dari Java ini dapat berjalan dengan platform sistem operasi yang berbeda karena sifatnya yang umum dan non-spesifik.



Gambar 1.1 Logo Java

Sekarang ini Java menjadi sebuah bahasa pemrograman yang populer dan dimanfaatkan secara luas untuk pengembangan perangkat lunak. Kebanyakan perangkat lunak yang menggunakan Java adalah ponsel feature dan ponsel pintar atau smartphone.

Kelebihan dan kekurangan Java. Setelah membahas mengenai pengertian java, selanjutnya kita membahas mengenai kelebihan dan kekurangan java. Kelebihan Java yang pertama tentu saja multiplatform. Java dapat dijalankan dalam beberapa platform komputer dan sistem operasi yang berbeda. Hal ini sesuai dengan slogannya yangs udah dibahas sebelumnya. Yang kedua adalah OOP atau Object Oriented Programming. Java memiliki library yang lengkap. Library disini adalah sebuah kumpulan dari program yang disertakan dalam Java. Hal ini akan memudahkan pemrograman menjadi lebih mudah. Kelengkapan library semakin beragam jika ditambah dengan karya komunitas Java.

Setiap hal pasti memiliki kelebihan dan kekurangan. Kekurangan yang dimiliki oleh Java adalah pada satu slogannya, takni “Tulis sekali dan jalankan dimana saja” ternyata tidak sepenuhnya benar. Beberapa hal harus disesuaikan jika dijalankan pada platform yang berbeda. Misalnya untuk J2SE dengan platform SWT-AWT bridge tidak dapat berfungsi di Mac OS X. Kekurangan lainnya adalah kemudahan aplikasi Java didekompilasi. Dekompilasi adalah suatu proses membalikkan sebuah aplikasi menjadi kode sumbernya. Hal ini memungkinkan terjadi pada Java karena berupa bytecode yang menyimpan bahasa tingkat tinggi. Hal ini terjadi pula pada platform .NET dari Microsoft sehingga program yang dihasilkan mudah dibajak kodenya karena sulit untuk disembunyikan.

Kekurangan Java yang lain adalah penggunaan memori yang cukup banyak, lebih besar daripada bahasa tingkat tinggi sebelum generasi Java. Namun hal ini memang sesuai dengan fitur beragam yang dimiliki oleh Java. Masalah memori ini juga tidak dialami oleh semua pengguna aplikasi Java. Mereka yang sudah menggunakan perangkat keras dengan teknologi terbaru tidak merasakan kelambatan dan konsumsi memori Java yang tinggi. Lain halnya dengan mereka yang menggunakan teknologi lama atau komputer yang sudah berumur tua lebih dari empat tahun akan merasakan adanya kelambatan. Namun apapun kelemahan yang dimiliki Java, faktanya adalah Java merupakan bahasa pemrograman yang populer dan digunakan di seluruh dunia saat ini.

* + 1. Basis Data

Basis data atau database adalah kumpulan informasi yang disimpan di dalam komputer secara sistematik sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut. Perangkat lunak yang digunakan untuk mengelola dan memanggil kueri (query) basis data disebut sistem manajemen basis data (database management system, DBMS). Sistem basis data dipelajari dalam ilmu informasi.

Konsep dasar dari basis data adalah kumpulan dari catatan-catatan, atau potongan dari pengetahuan. Sebuah basis data memiliki penjelasan terstruktur dari jenis fakta yang tersimpan di dalamnya.

Secara konsep basis data atau database adalah kumpulan dari data-data yang membentuk suatu berkas (file) yang saling berhubungan (relation) dengan tatacara yang tertentu untuk membentuk data baru atau informasi. Atau basis data (database) merupakan kumpulan dari data yang saling berhubungan (relasi) antara satu dengan yang lainnya yang diorganisasikan berdasarkan skema atau struktur tertentu.

Basis data penting dalam pengolaan data dalam menggunakan komputer, karena bebrapa alasan yaitu sebagai komponen utama atau penting dalam sistem informasi, karena merupakan dasar dalam menyediakan informasi. Dan menentukan kualitas informasi yang cepat, akurat, relevan, sehingga informasi yang disajikan tidak basi. Informasi dapat dikatakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya mendapatkannya.

Terdapat tiga jenis bahasa komputer yang digunakan saat kita ingin membangun dan memanipulasi sebuah basis data, yaitu: Data Definition Language (DDL), Data Manipulation Language (DML), Data Control Language (DCL).

Komponen-komponen sistem basis data adalah:

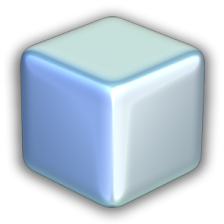
Hardware, sebagai pendukung operasi pengolahan data seperti CPU, memori, disk, terminal, dan sebagainya. Software sistem operasi , (Windows 9x, Windows 2000/XP, Linux, Unix.) Software pengelola basis data (DBMS) seperti MS-Access, SQL, Oracle. Software program aplikasi misalnya Visual Basic, Delphi, Visual Foxpro. Basis data (semua data yang diperlukan, dipelihara, dikelola oleh sistem Basis Data) dan Pemakai/pengguna basis data (user).

* + 1. Netbeans IDE

NetBeans mengacu pada kedua platform kerangka untuk aplikasi desktop Java, dan sebuah lingkungan pengembangan terpadu (IDE) untuk pengembangan dengan Java , JavaScript , PHP , Python , Ruby , Groovy , C, C + + , Scala , Clojure , dan lain-lain.

Semula, aplikasi NetBeans IDE ini diperuntukkan bagi pengembangan dalam Java. Namun, aplikasi ini juga mendukung program-program pembuatan bahasa lain secara khusus seperti PHP, C/C++ dan HTML5.

NetBeans adalah alat lintas serambi serta penerapannya dijalankan pada Microsoft Windows, Mac OS X, Linux, Solaris dan serambi-serambi lainnya yang mendukung JVM yang sepadan.



Gambar 1.2 Icon Netbeans

NetBeans IDE ditulis dalam Java dan berjalan di mana-mana mana JVM diinstal, termasuk Windows, Mac OS, Linux, dan Solaris.Sebuah JDK diperlukan untuk pengembangan fungsionalitas Jawa, tetapi tidak diperlukan untuk pembangunan di bahasa pemrograman lain.

Platform NetBeans memungkinkan aplikasi untuk dikembangkan dari satu set modular komponen software yang disebut modul. Aplikasi berbasis platform NetBeans (termasuk IDE NetBeans) dapat diperpanjang oleh pengembang pihak ketiga .

NetBeans dimulai pada tahun 1996 sebagai Xelfi (kata bermain pada Delphi ), Java IDE proyek mahasiswa di bawah bimbingan Fakultas Matematika dan Fisika di Charles University di Praha . Pada tahun 1997 Staněk Romawi membentuk perusahaan sekitar proyek tersebut dan menghasilkan versi komersial NetBeans IDE hingga kemudian dibeli oleh Sun Microsystems pada tahun 1999. Komunitas NetBeans sejak terus tumbuh, berkat individu dan perusahaan yang menggunakan dan berkontribusi dalam proyek ini.

NetBeans IDE 6.0 memperkenalkan dukungan untuk mengembangkan modul IDE dan aplikasi klien kaya berdasarkan platform NetBeans, Java Swing GUI builder (sebelumnya dikenal sebagai "Proyek Matisse"), meningkatkan CVS dukungan, WebLogic 9 dan JBoss 4 dukungan, dan perangkat tambahan banyak editor. NetBeans 6 is available in official repositories of major Linux distributions. NetBeans 6 tersedia dalam repositori resmi dari distribusi Linux utama.

Selain itu, NetBeans Enterprise Pack mendukung pengembangan aplikasi Java EE 5 perusahaan, termasuk SOA alat desain visual, skema XML tools, web orkestrasi layanan (untuk BPEL), dan UML modeling. The NetBeans IDE Bundle for C/C++ supports C/C++ development. The NetBeans IDE Bundle untuk C / C + + mendukung C / C + + pembangunan.

Hosting pengembang sumber terbuka proyek di kenai.com tambahan manfaat dari instant messaging dan pelacakan masalah integrasi dan navigasi kanan dalam IDE, dukungan untuk pengembangan aplikasi web dengan PHP 5.3 dan kerangka Symfony, dan kode selesai diperbaiki, layout, petunjuk dan navigasi dalam proyek JavaFX.

Dirilis pada bulan Juni 2010, menambahkan dukungan untuk OSGi , Spring Framework 3.0, Java EE injeksi ketergantungan (JSR-299), Zend Framework untuk PHP , dan navigasi kode lebih mudah (seperti "Apakah / ditimpa Penerapan" penjelasan), format , petunjuk, dan refactoring di beberapa bahasa.

Versi beta dari NetBeans 8.2 kini telah tersedia, hal tersebut diumumkan oleh pengembang NetBeans. Pada rilis ini, pemutakhiran IDE Java akan didukung dengan Java SE 8, penambahan beberapa kakas baru dan peningkatan fitur sebelumnya.

* + 1. MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (bahasa Inggris: database management system) atau DBMS yang multialur, multipengguna, dengan sekitar 6 juta instalasi di seluruh dunia. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis di bawah lisensi GNU General Public License (GPL), tetapi mereka juga menjual di bawah lisensi komersial untuk kasus-kasus di mana penggunaannya tidak cocok dengan penggunaan GPL.

MySQL adalah sebuah implementasi dari sistem manajemen basis data relasional (RDBMS) yang didistribusikan secara gratis di bawah lisensi GPL (General Public License). Setiap pengguna dapat secara bebas menggunakan MySQL, namun dengan batasan perangkat lunak tersebut tidak boleh dijadikan produk turunan yang bersifat komersial. SQL (Structured Query Language) adalah sebuah konsep pengoperasian basisdata, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis.



Gambar 1.3 MySQL

MySQL pada awalnya diciptakan pada tahun 1979, oleh Michael "Monty" Widenius, seorang programmer komputer asal Swedia. Monty mengembangkan sebuah sistem database sederhana yang dinamakan UNIREG yang menggunakan koneksi low-level ISAM database engine dengan indexing.

Hal paling mendasar yang menjadikan MySQL pilihan utama sebagai database yang digunakan adalah karena MySQL menggunakan Lisensi GPL dan multiplatform, sehingga lebih disukai para mahasiswa karena tidak membutuhkan biaya besar dalam membuat aplikasi serta tidak harus tergantung pada OS Windows ataupun Linux karena dapat dijalankan pada kedua OS tersebut dan beberapa OS lainnya.

Terdapat beberapa API (Application Programming Interface) tersedia yang memungkinkan aplikasi-aplikasi komputer yang ditulis dalam berbagai bahasa pemrograman untuk dapat mengakses basis data MySQL antara lain: bahasa pemrograman C, C++, C#, bahasa pemrograman Eiffel, bahasa pemrograman Smalltalk, bahasa pemrograman Java, bahasa pemrograman Lisp, Perl, PHP, bahasa pemrograman Python, Ruby, REALbasic dan Tcl. Sebuah antarmuka ODBC memanggil MyODBC yang memungkinkan setiap bahasa pemrograman yang mendukung ODBC untuk berkomunikasi dengan basis data MySQL. Kebanyakan kode sumber MySQL dalam ANSI C.

* + 1. XAMPP

XAMPP adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program.

Fungsinya adalah sebagai server yang berdiri sendiri (localhost), yang terdiri atas program Apache HTTP Server, MySQL database, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Nama XAMPP merupakan singkatan dari X (empat sistem operasi apapun), Apache, MySQL, PHP dan Perl. Program ini tersedia dalam GNU General Public License dan bebas, merupakan web server yang mudah digunakan yang dapat melayani tampilan halaman web yang dinamis.



Gambar 1.4 Logo XAMPP

XAMPP dikembangkan dari sebuah tim proyek bernama Apache Friends, yang terdiri dari Tim Inti (Core Team), Tim Pengembang (Development Team) & Tim Dukungan (Support Team).

XAMPP adalah singkatan yang masing-masing hurufnya adalah:

X: Program ini dapat dijalankan dibanyak sistem operasi, seperti Windows, Linux, Mac OS, dan Solaris.

A: Apache, merupakan aplikasi web server. Tugas utama Apache adalah menghasilkan halaman web yang benar kepada user berdasarkan kode PHP yang dituliskan oleh pembuat halaman web. jika diperlukan juga berdasarkan kode PHP yang dituliskan,maka dapat saja suatu database diakses terlebih dahulu (misalnya dalam MySQL) untuk mendukung halaman web yang dihasilkan.

M: MySQL, merupakan aplikasi database server. Perkembangannya disebut SQL yang merupakan kepanjangan dari Structured Query Language. SQL merupakan bahasa terstruktur yang digunakan untuk mengolah database. MySQL dapat digunakan untuk membuat dan mengelola database beserta isinya. Kita dapat memanfaatkan MySQL untuk menambahkan, mengubah, dan menghapus data yang berada dalam database.

P: PHP, bahasa pemrograman web. Bahasa pemrograman PHP merupakan bahasa pemrograman untuk membuat web yang bersifat server-side scripting. PHP memungkinkan kita untuk membuat halaman web yang bersifat dinamis. Sistem manajemen basis data yang sering digunakan bersama PHP adalah MySQl. namun PHP juga mendukung sistem manajement database Oracle, Microsoft Access, Interbase, d-base, PostgreSQL, dan sebagainya.

P: Perl, bahasa pemrograman untuk segala keperluan, dikembangkan pertama kali oleh Larry Wall di mesin Unix. Perl dirilis pertama kali pada tanggal 18 Desember 1987 ditandai dengan keluarnya Perl 1. Pada versi-versi selanjutnya, Perl tersedia pula untuk berbagai sistem operasi varian Unix (SunOS, Linux, BSD, HP-UX), juga tersedia untuk sistem operasi seperti DOS, Windows,PowerPC, BeOS, VMS, EBCDIC, dan PocketPC.

BAB II

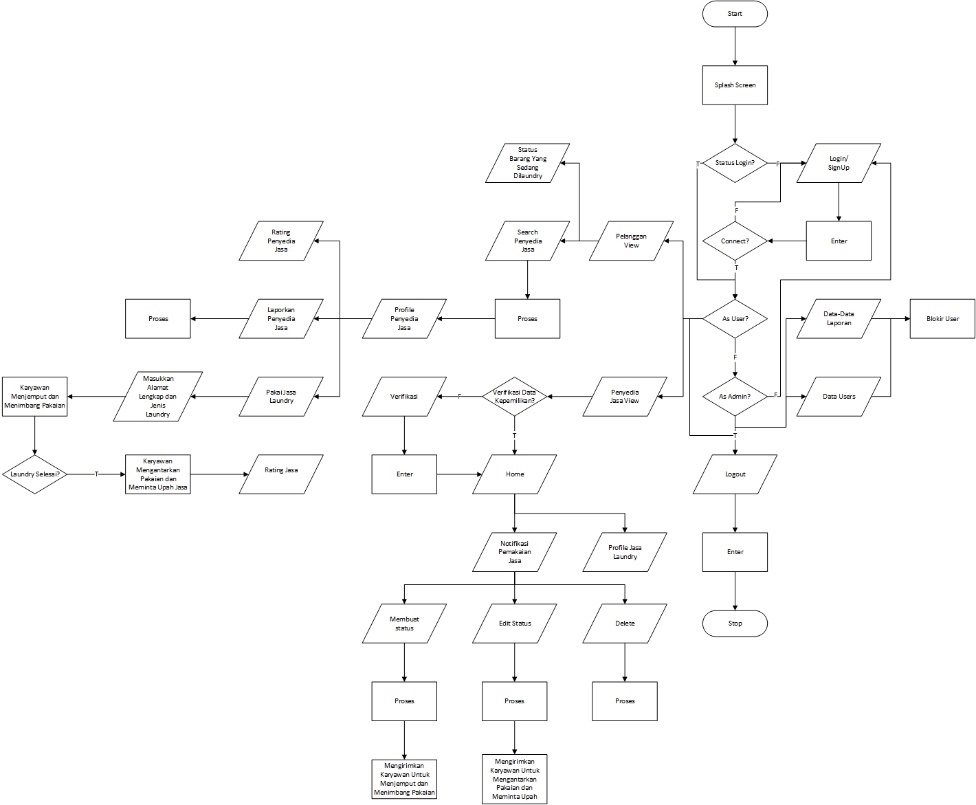
PERANCANGAN SISTEM

* 1. Flowchart
     1. Definisi Flowchart

Flowchart atau diagram alir adalah sebuah jenis diagram yang mewakili algoritme, alir kerja atau proses, yang menampilkan langkah-langkah dalam bentuk simbol-simbol grafis, dan urutannya dihubungkan dengan panah. Diagram ini mewakili ilustrasi atau penggambaran penyelesaian masalah. Diagram alir digunakan untuk menganalisis, mendesain, mendokumentasi atau memanajemen sebuah proses atau program di berbagai bidang.

Diagram alir digunakan untuk mendesain dan mendokumentasi proses atau program sederhana. Seperti jenis diagram lainnya, diagram ini membantu menggambarkan apa yang sedang terjadi dan dengan demikian membantu mengerti sebuah proses. Dan mungkin saja menentukan kekurangan fitur, atau bagian yang kurang jelas didalam sebuah proses. Terdapat beberapa bentuk diagram alir, dan setiap bentuk memiliki urutan dan peranan masing-masing.

* + 1. Gambar Flowchart



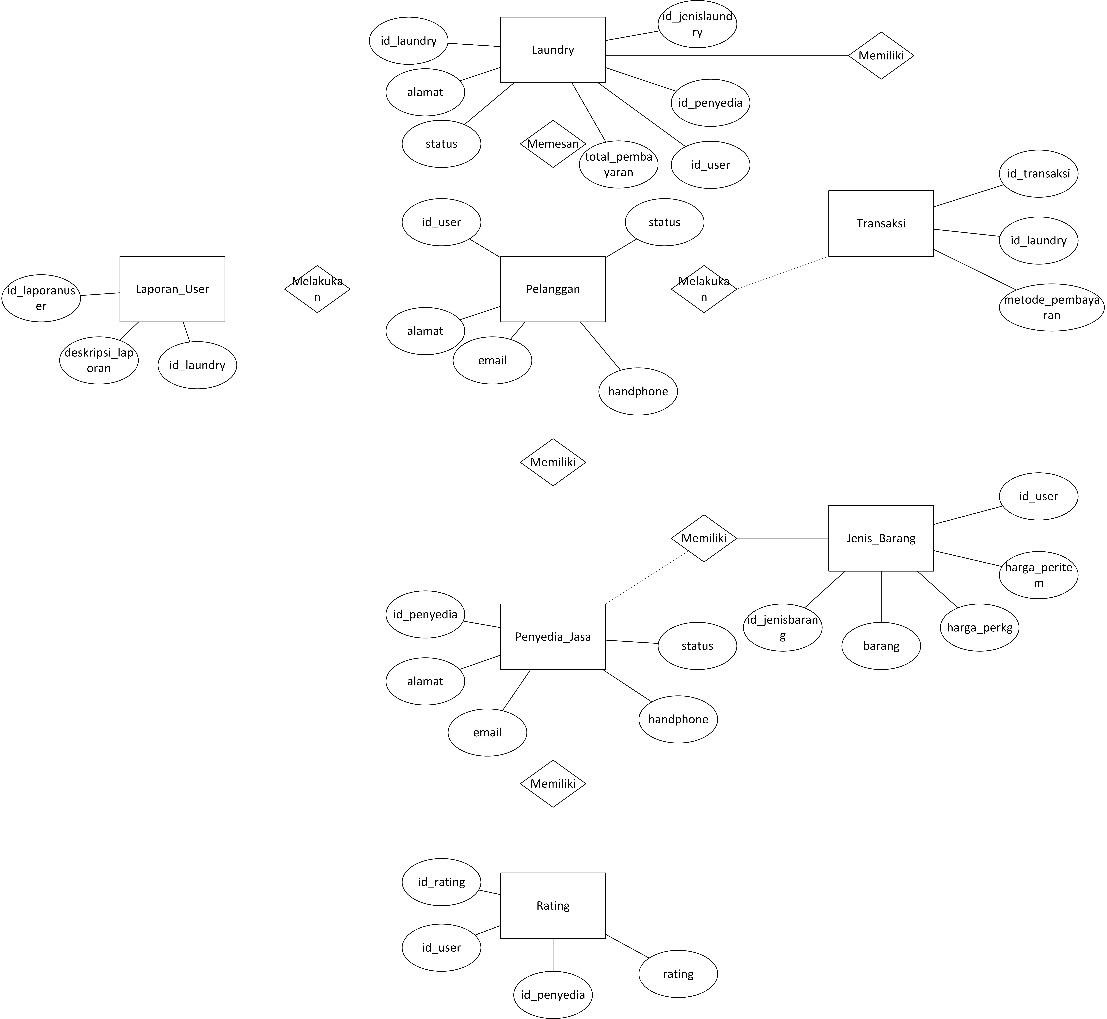
|  |
| --- |
| Gambar 2.1 Flowchart Laundry |

* 1. ERD(Entity Relationship Diagram)
     1. Definisi ERD

Diagram Hubungan Entitas atau entity relationship diagram merupakan model data berupa notasi grafis dalam pemodelan data konseptual yang menggambarkan hubungan antara penyimpan. Model data sendiri merupakan sekumpulan cara, peralatan untuk mendeskripsikan data-data yang hubungannya satu sama lain, semantiknya, serta batasan konsistensi. Model data terdiri dari model hubungan entitas dan model relasional. Diagram hubungan entitas ditemukan oleh Peter Chen dalam buku Entity Relational Model-Toward a Unified of Data. Chen mencoba merumuskan dasar-dasar model dan setelah itu dikembangkan dan dimodifikai oleh Chen dan banyak pakar lainnya. Pada saat itu diagram hubungan entitas dibuat sebagai bagian dari perangkat lunak yang juga merupakan modifikasi khusus, karena tidak ada bentuk tunggal dan standar dari diagram hubungan entitas.

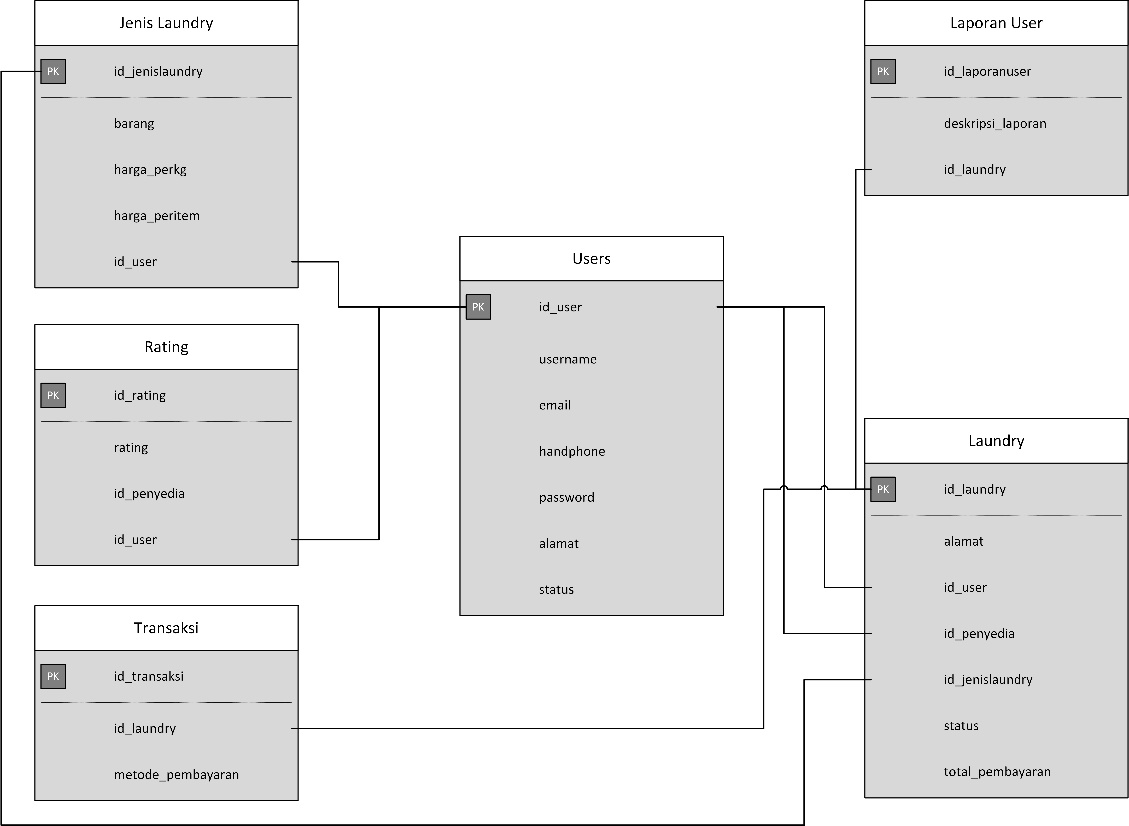
Diagram hubungan entitas digunakan untuk mengkonstruksikan model data konseptual, memodelkan struktur data dan hubungan antar data dan mengimplementasikan basis data secara logika maupun secara fisik dengan '''DBMS''' (''Database Management system'').

* + 1. Gambar ERD



|  |
| --- |
| Gambar 2.2 ERD Laundry |

* 1. Rancangan Database dan Table
     1. Rancangan Database



|  |
| --- |
| Gambar 2.3 Perancangan Database Laundry |

* + 1. Rancangan Table

1. Jenis Laundry

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NO | Field Name | Tipe | Panjang | Key |
| 1 | Id\_jenislaundry | INT | 11 | Primary Key |
| 2 | barang | VARCHAR | 45 |  |
| 3 | Harga\_perkg | VARCHAR | 12 |  |
| 4 | Harga\_peritem | VARCHAR | 12 |  |
| 5 | Id\_user | INT | 11 | Foreign Key |

1. Laporan User

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NO | Field Name | Tipe | Panjang | Key |
| 1 | Id\_laporanuser | INT | 11 | Primary Key |
| 2 | Deskripsi\_laporam | TEXT |  |  |
| 3 | Id\_laundry | INT | 11 | Foreign Key |

1. Laundry

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NO | Field Name | Tipe | Panjang | Key |
| 1 | Id\_laundry | INT | 11 | Primary Key |
| 2 | Alamat | TEXT |  |  |
| 3 | Id\_user | INT | 11 | Foreign Key |
| 4 | Id\_penyedia | INT | 11 | Foreign Key |
| 5 | Id\_jenislaundry | INT | 11 | Foreign Key |
| 6 | Status | INT | 11 |  |
| 7 | Total\_pembayaran | VARCHAR | 35 |  |

1. Rating

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NO | Field Name | Tipe | Panjang | Key |
| 1 | Id\_rating | INT | 11 | Primary Key |
| 2 | Rating | ENUM | ‘1’,’2’,’3’,’4’,’5’ |  |
| 3 | Id\_user | INT | 11 | Foreign Key |
| 4 | Id\_penyedia | INT | 11 | Foreign Key |

1. Transaksi

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NO | Field Name | Tipe | Panjang | Key |
| 1 | Id\_transaksi | INT | 11 | Primary Key |
| 2 | Id\_laundry | INT | 11 | Foreign Key |
| 3 | Metode\_pembayaran | ENUM | 'dana', 'ovo', 'gopay' |  |

1. Users

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NO | Field Name | Tipe | Panjang | Key |
| 1 | Id\_user | INT | 11 | Primary Key |
| 2 | Username | VARCHAR | 45 |  |
| 3 | Email | VARCHAR | 45 |  |
| 4 | Handphone | VARCHAR | 20 |  |
| 5 | Password | TEXT |  |  |
| 6 | Alamat | TEXT |  |  |
| 7 | Status | INT | 11 |  |

* 1. Rancangan User
     1. Macam – Macam User
* Pengguna/User
* Penyedia
* Admin
  + 1. Pembagian Hak - Hak Akses

1. Hak-Hak Pengguna

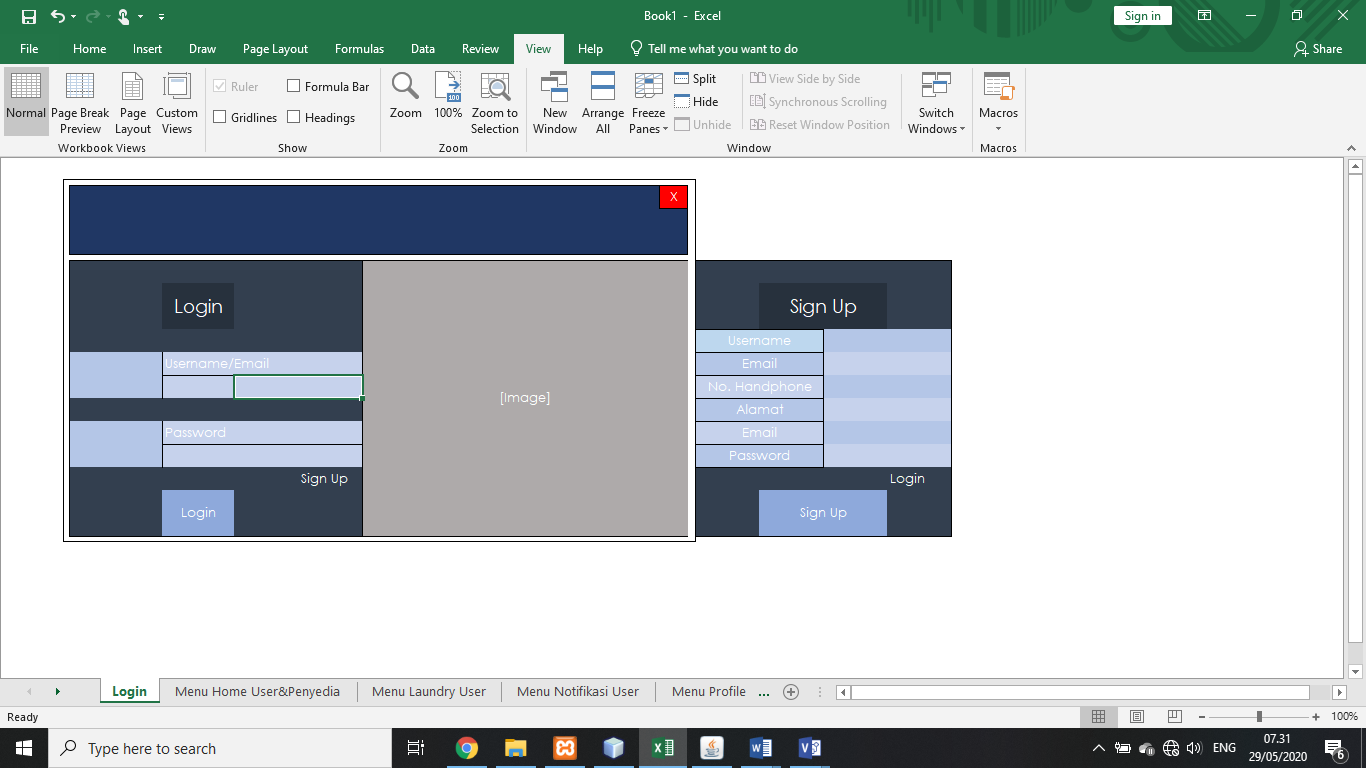
* Memesan jasa laundry.
* Melaporkan penyedia.
* Membayar jasa dan mengubah status.
* Melihat detail penyedia.
* Mengedit profile serta mengganti password.
* Melamar sebagai penyedia.

1. Hak-Hak Penyedia

* Melihat total uang pemasukan.
* CRUD jenis laundry miliknya.
* Melihat detail pelanggannya.
* Mengubah status laundry.
* Memasukkan total harga dari laundry pelanggannya.
* Mengedit profile serta mengganti password.

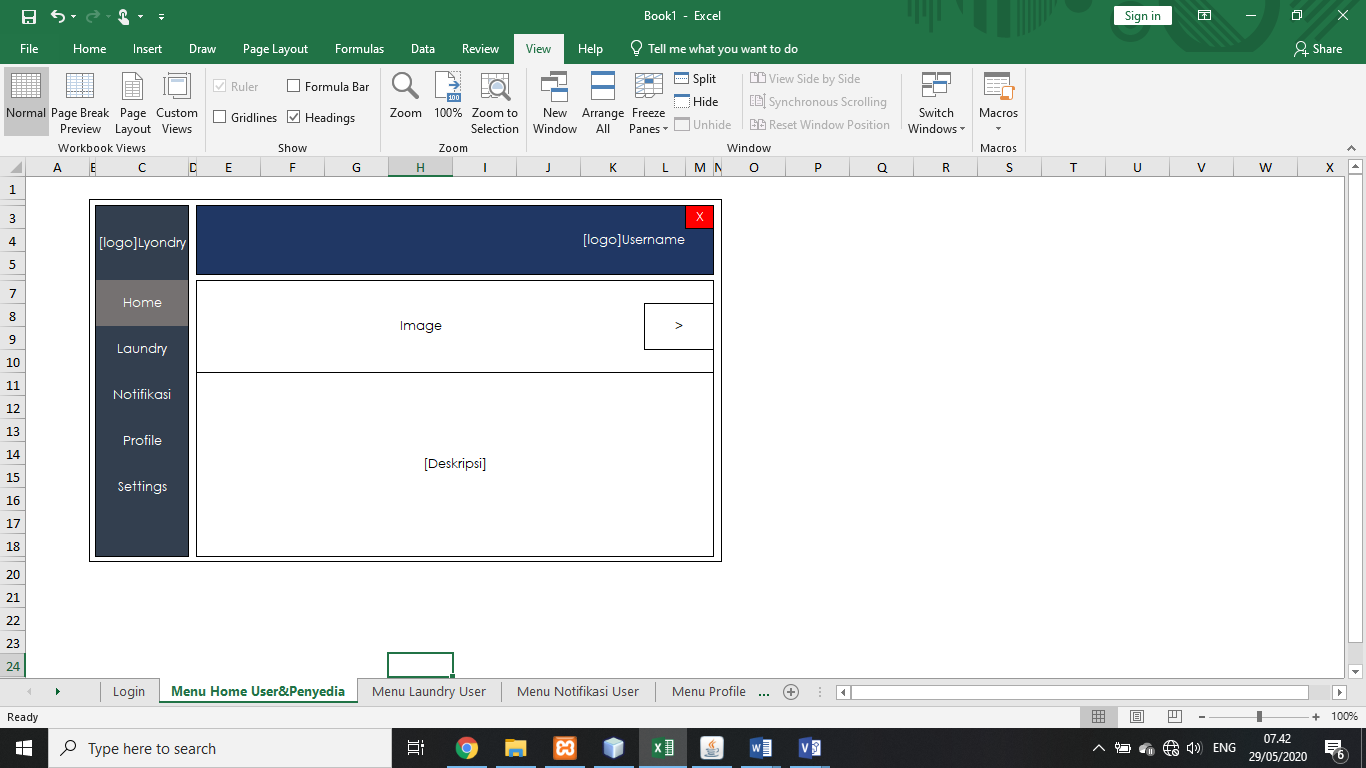
1. Hak-Hak Admin

* Melihat seluruh data User.
* Memblock user.
* Melihat dan menghapus laporan.
  1. User Interface
     1. Login



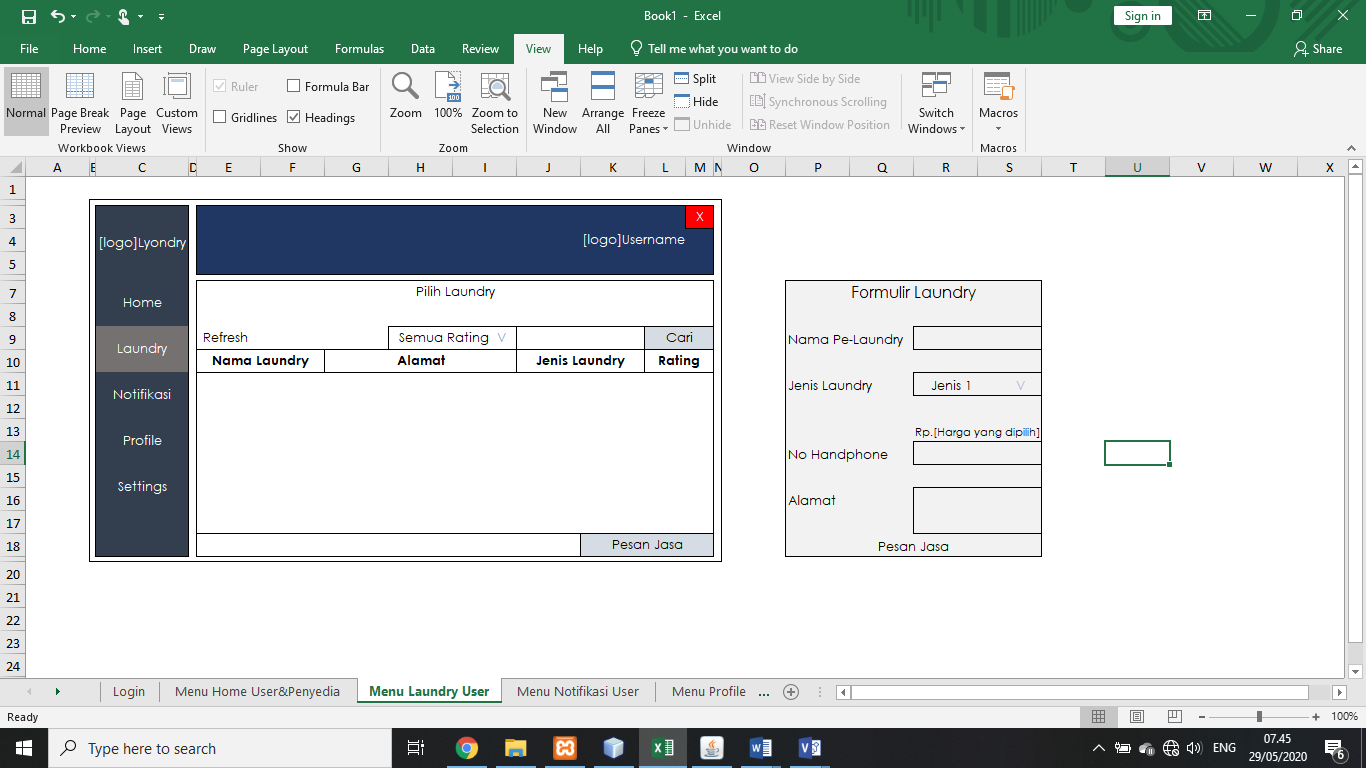
|  |
| --- |
| Gambar 2.4 Halaman Login |

* + 1. Menu Home User&Penyedia



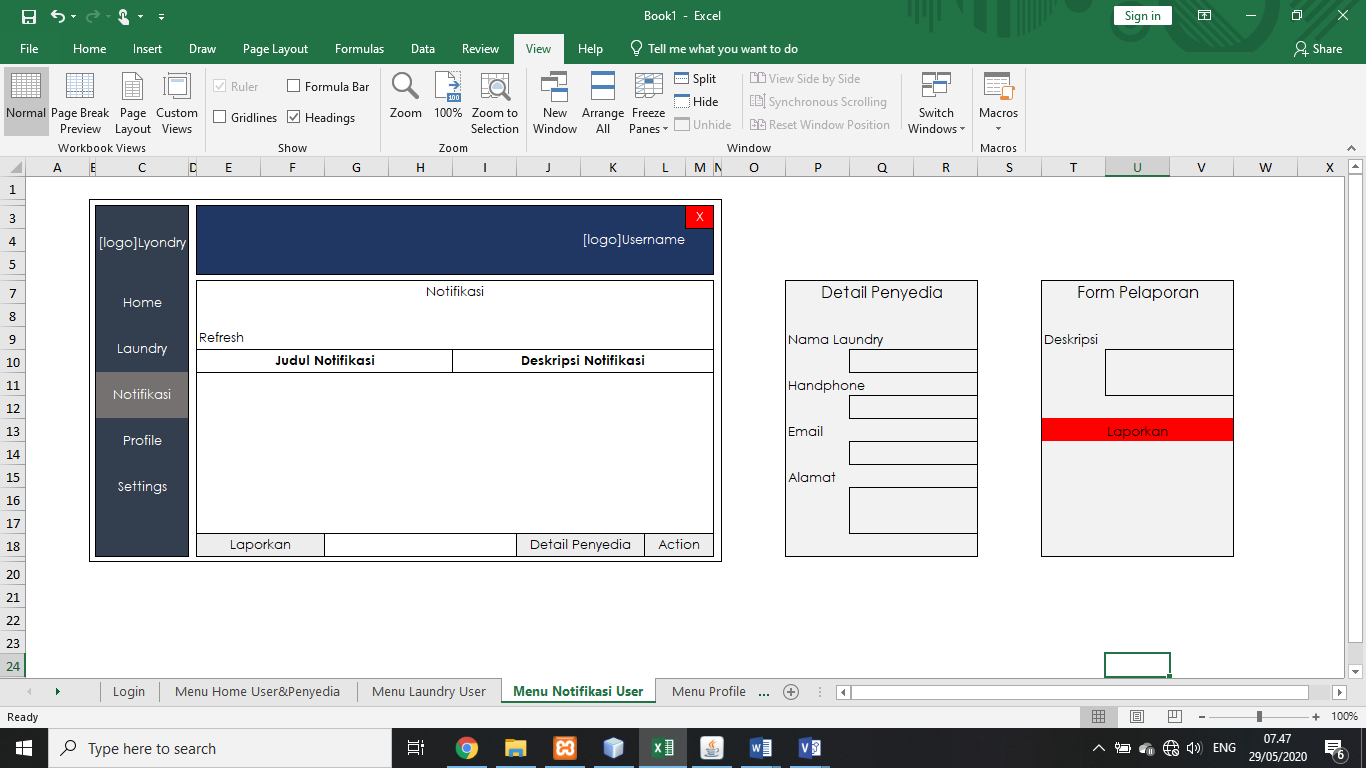
|  |
| --- |
| Gambar 2.5 Menu Home User&Penyedia |

* + 1. Menu Laundry User



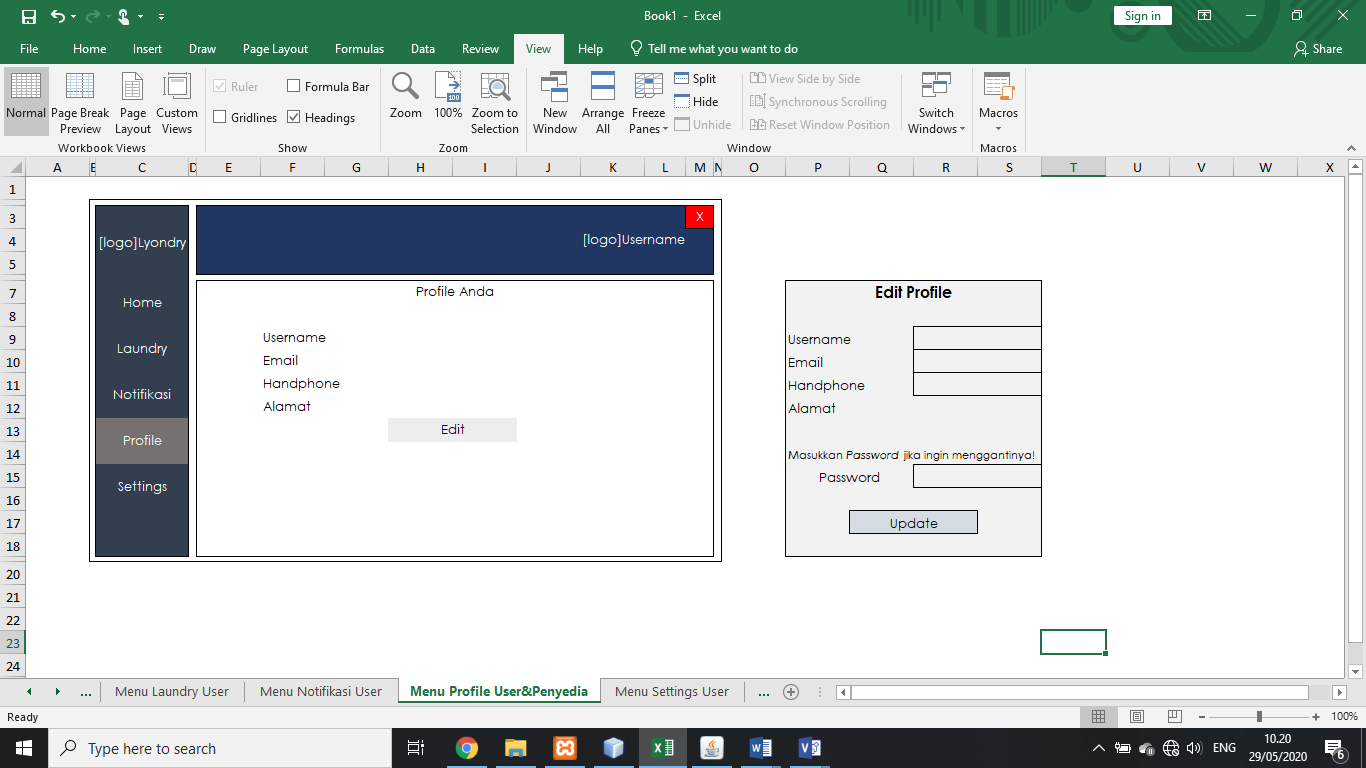
|  |
| --- |
| Gambar 2.6 Menu Laundry User |

* + 1. Menu Notifikasi User



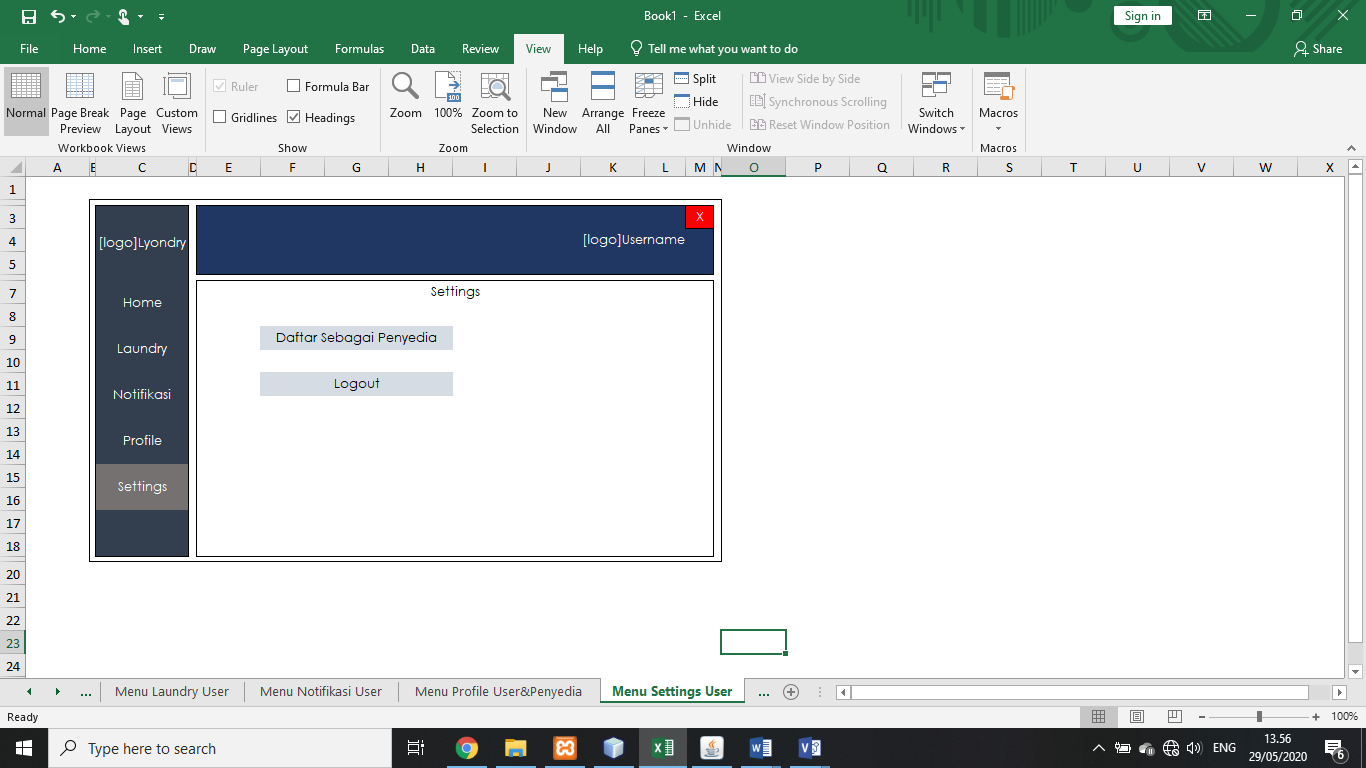
|  |
| --- |
| Gambar 2.7 Menu Notifikasi User |

* + 1. Menu Profile User&Penyedia



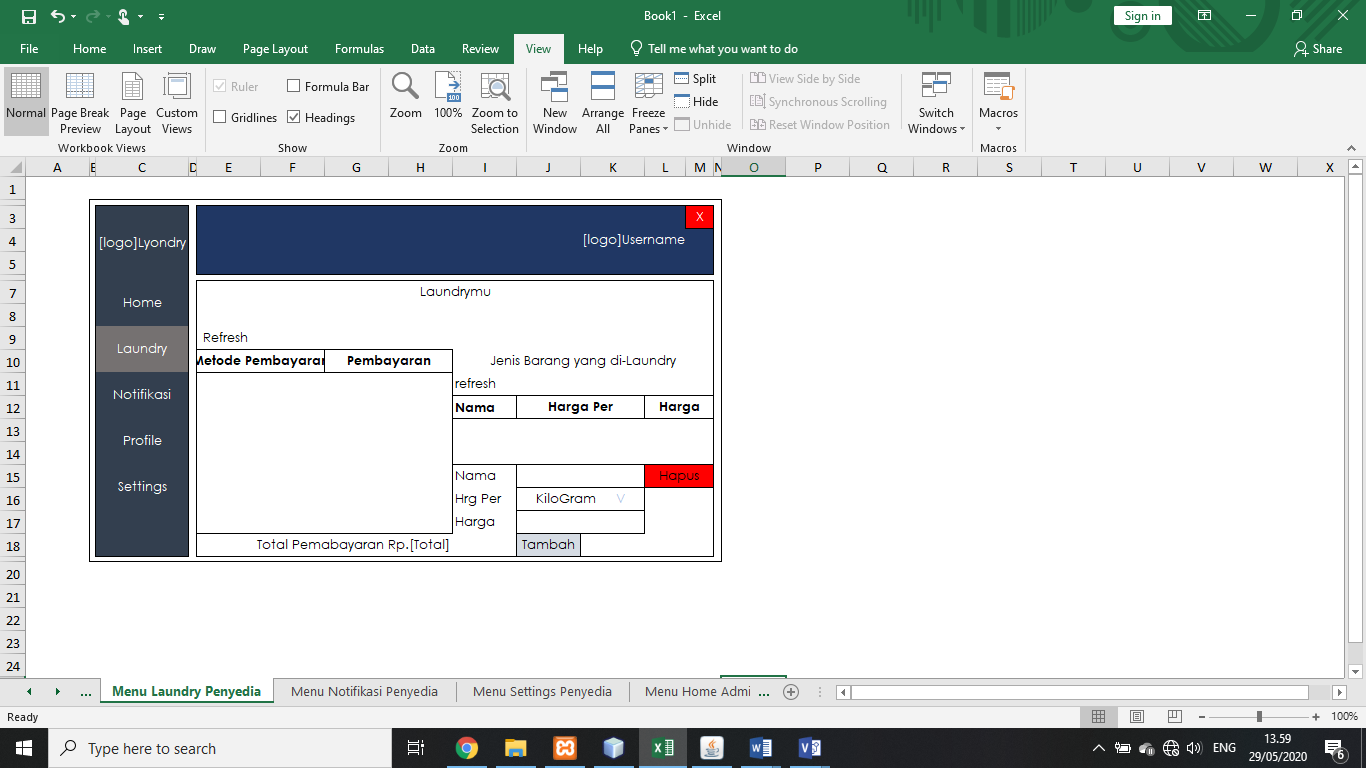
|  |
| --- |
| Gambar 2.8 Menu Profile User&Penyedia |

* + 1. Menu Settings User



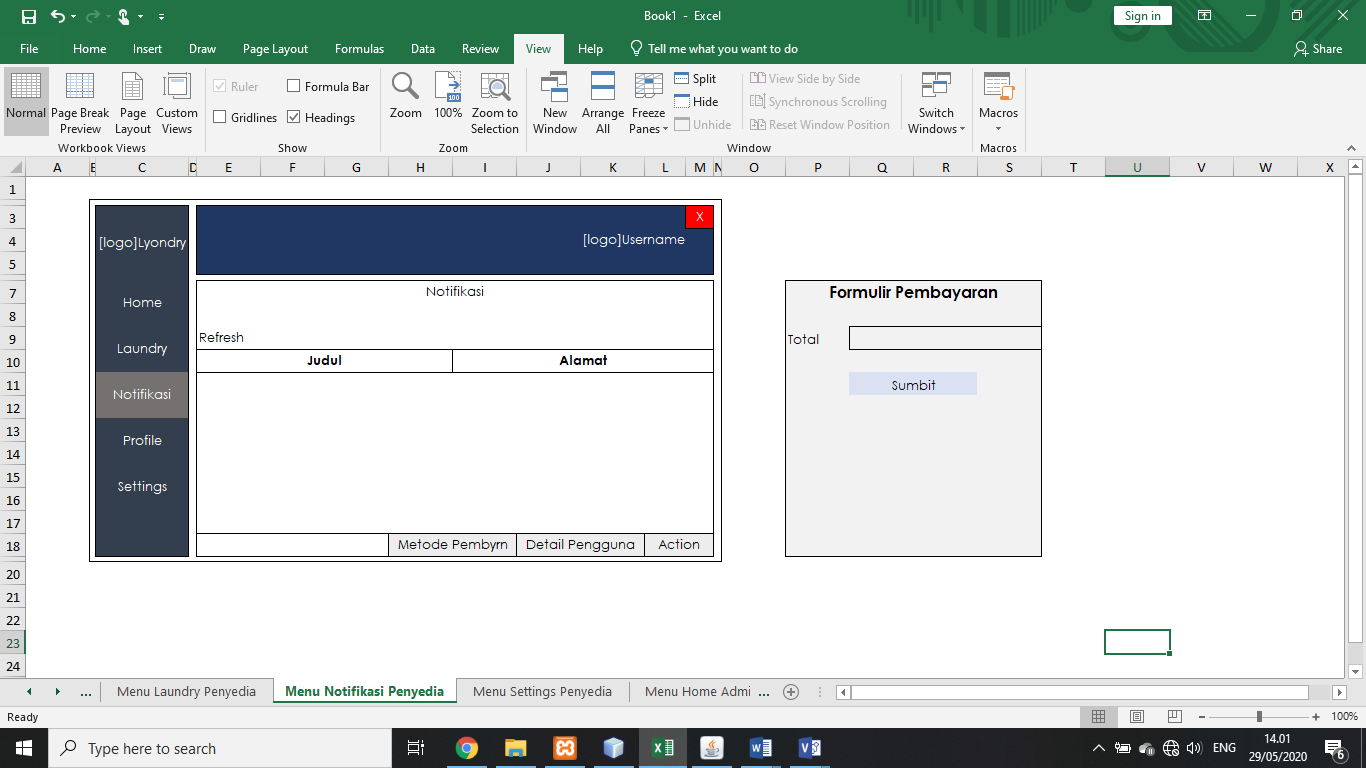
|  |
| --- |
| Gambar 2.9 Menu Settings User |

* + 1. Menu Laundry Peyedia



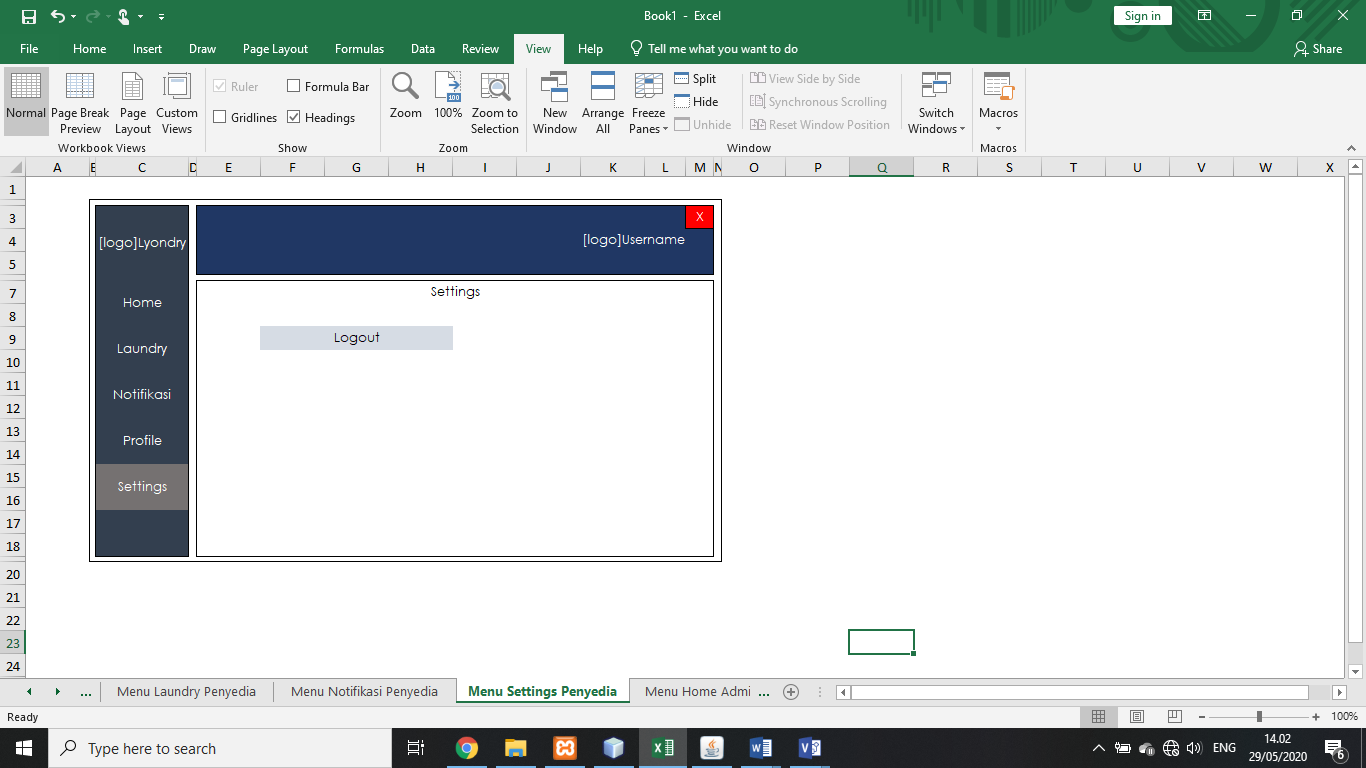
|  |
| --- |
| Gambar 2.10 Menu Laundry Penyedia |

* + 1. Menu Notifikasi Penyedia



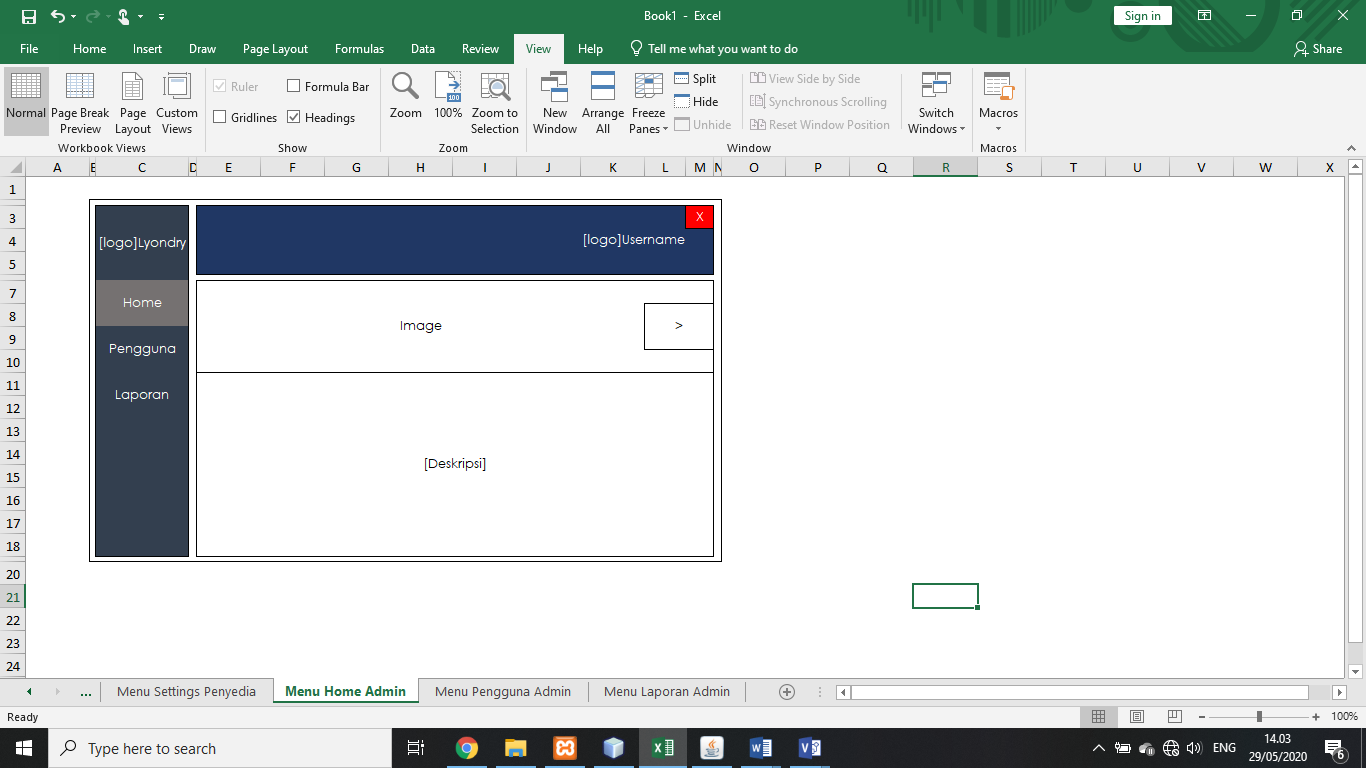
|  |
| --- |
| Gambar 2.11 Menu Notifkasi Penyedia |

* + 1. Menu Settings Penyedia



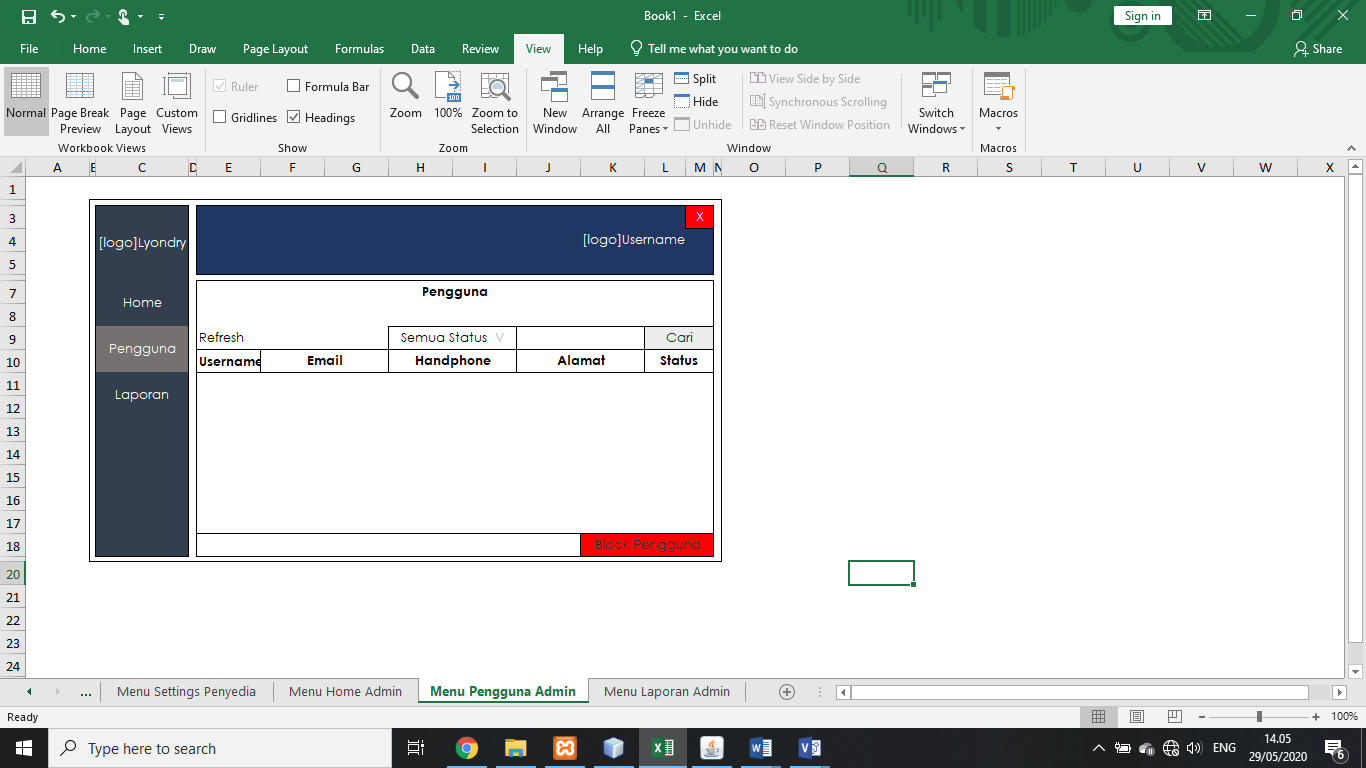
|  |
| --- |
| Gambar 2.12 Menu Settings Penyedia |

* + 1. Menu Home Admin



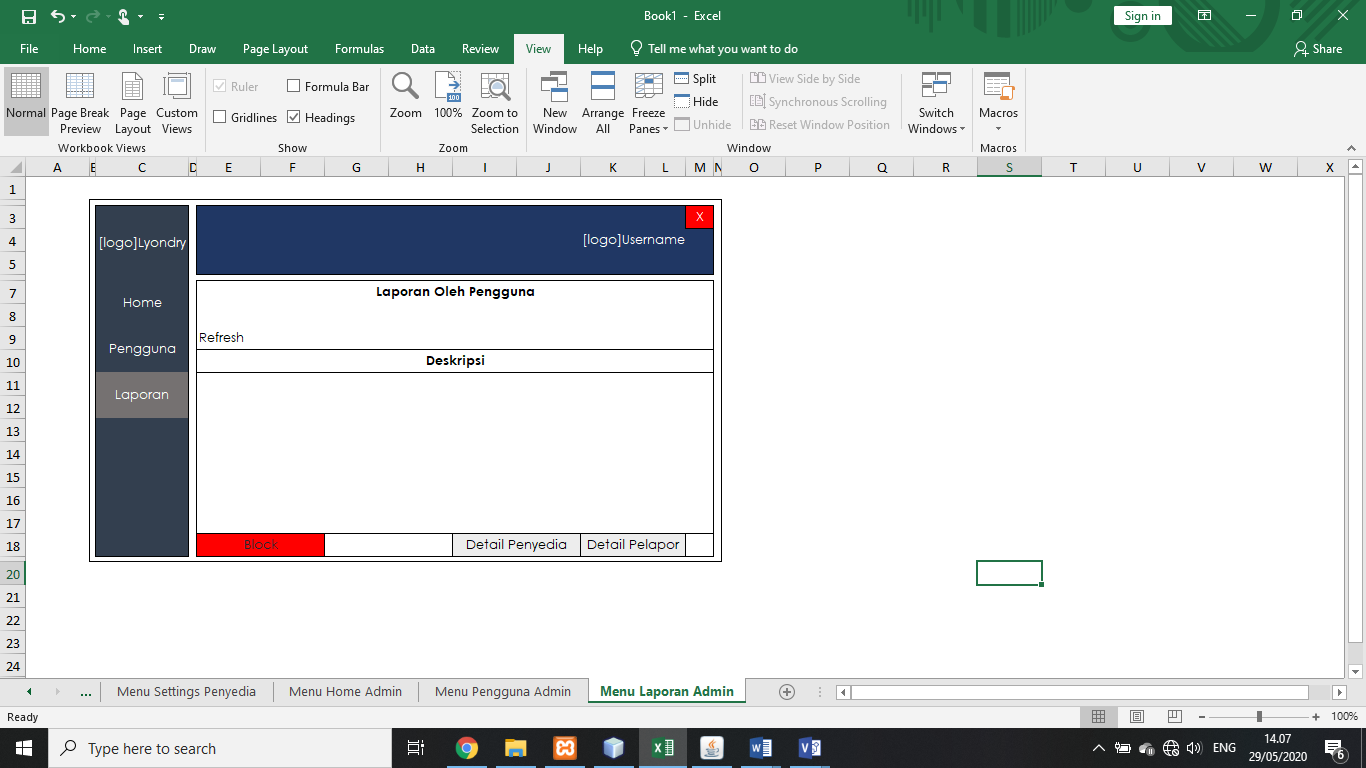
|  |
| --- |
| Gambar 2.13 Menu Home Admin |

* + 1. Menu Pengguna Admin



|  |
| --- |
| Gambar 2.14 Menu Pengguna Admin |

* + 1. Menu Laporan Admin



|  |
| --- |
| Gambar 2.15 Menu Laporan Admin |

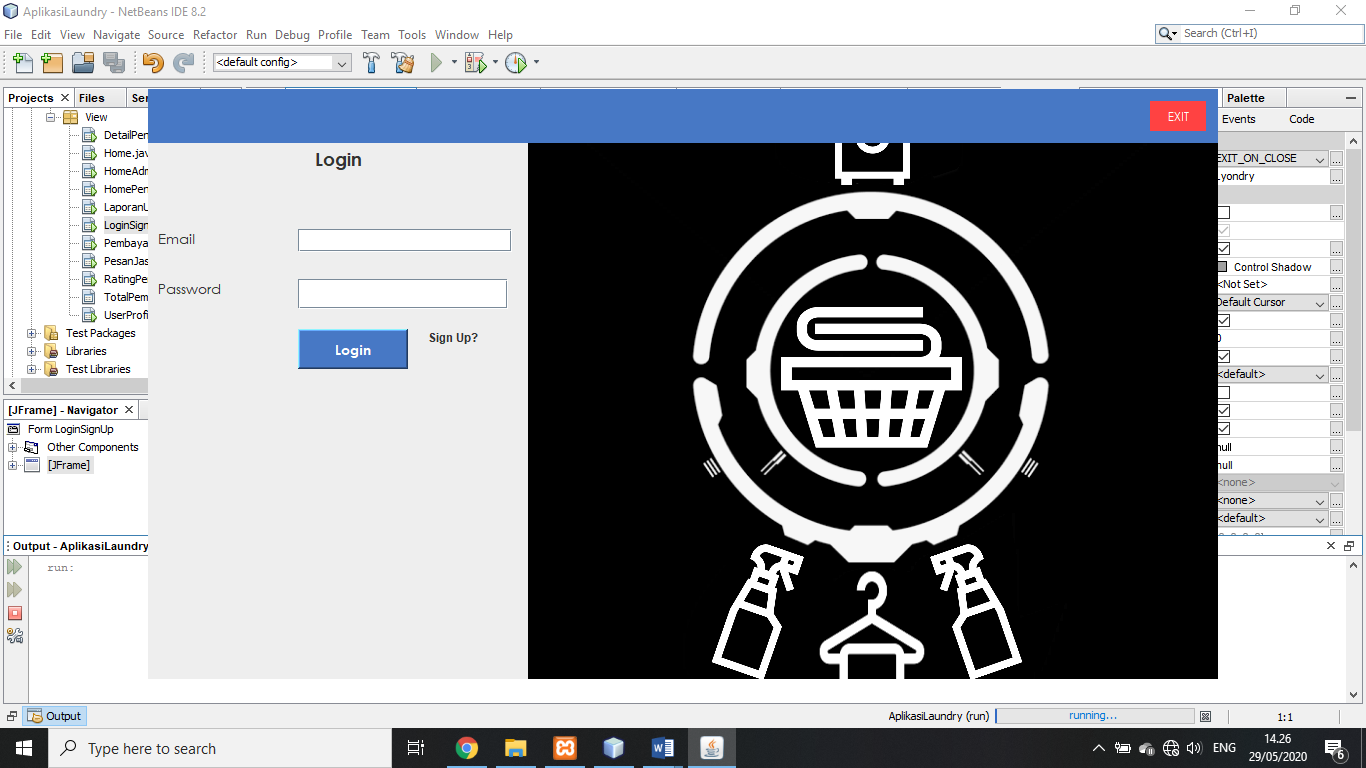
BAB III

HASIL DAN PEMBAHASAN

* 1. Implementasi

Setelah melakukan perancangan terhadap sistem langkah selanjutnya yaitu implementasi sistem. Implementasi sistem akan mengubah bentuk dari perancangan menjadi bentuk pemrograman.

* + 1. Login Form

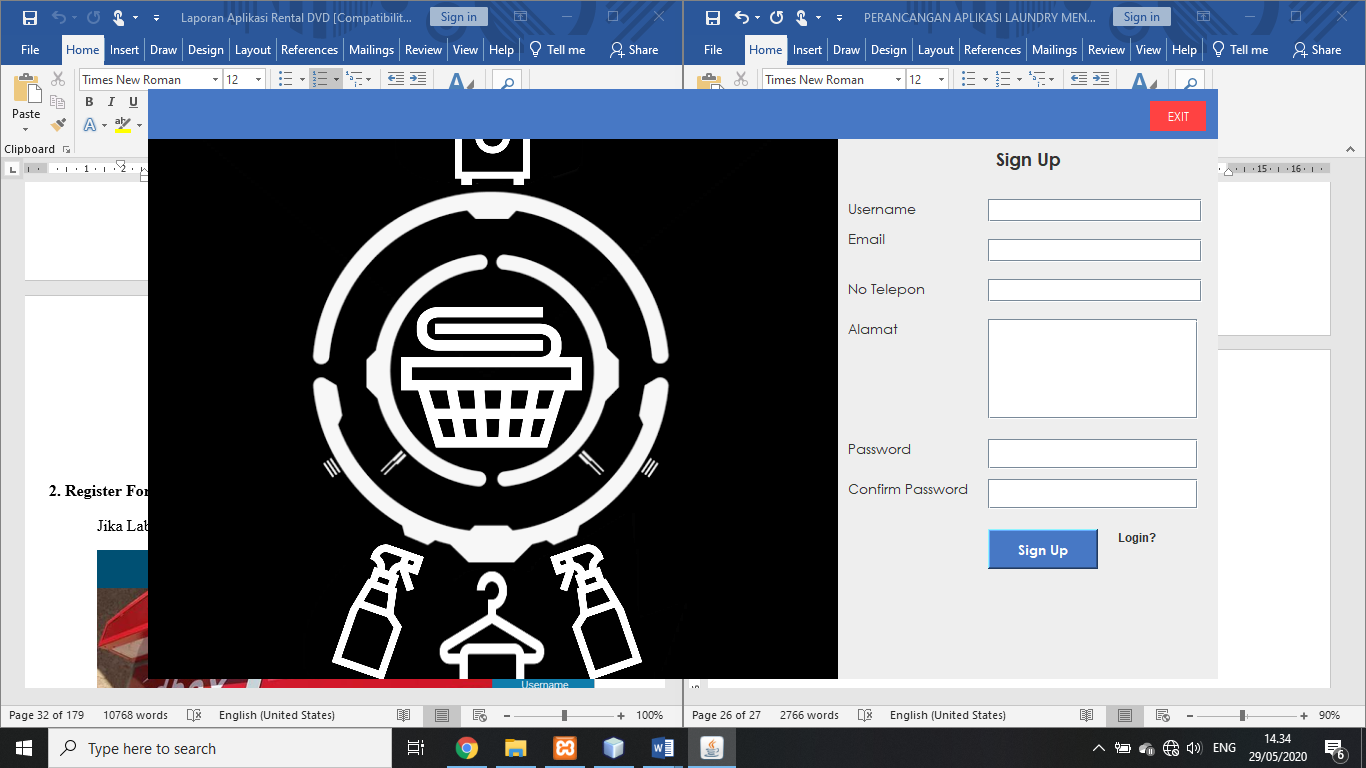


|  |
| --- |
| Gambar 3.1 Login Form |

Gambar di atas adalah form pertama kali muncul setelah aplikasi dijalankan. Form tersebutkan akan dilanjutkan jika Email dan Password terdaftar.

Urutkan langkah-langkah di bawah ini:

1. Masukkan Email dan Password.
2. Jika terdaftar maka Form baru akan ditampilkan.
3. Jika tidak terdaftar maka akan ada peringatan “Email atau Password yang Anda masukkan salah!”
   * 1. Sign Up Form

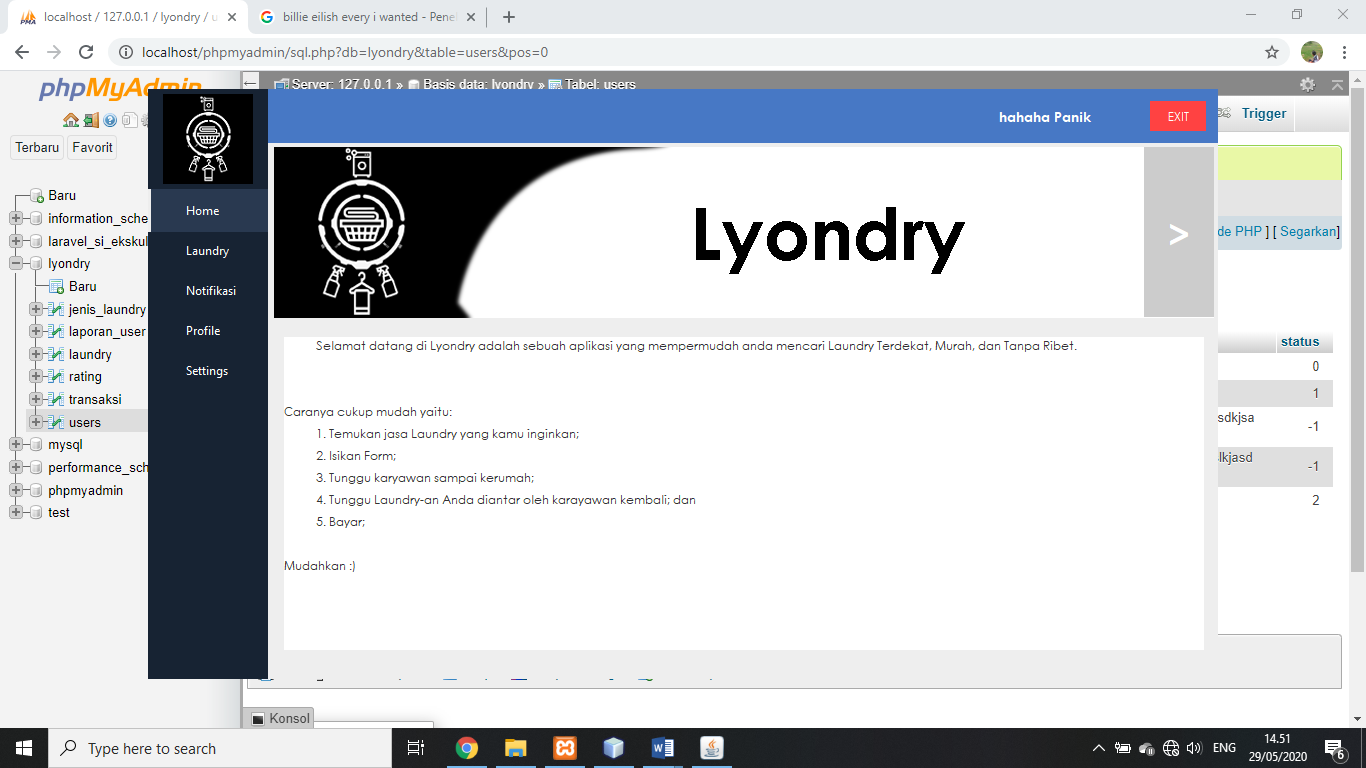


|  |
| --- |
| Gambar 3.2 Sign Up Form |

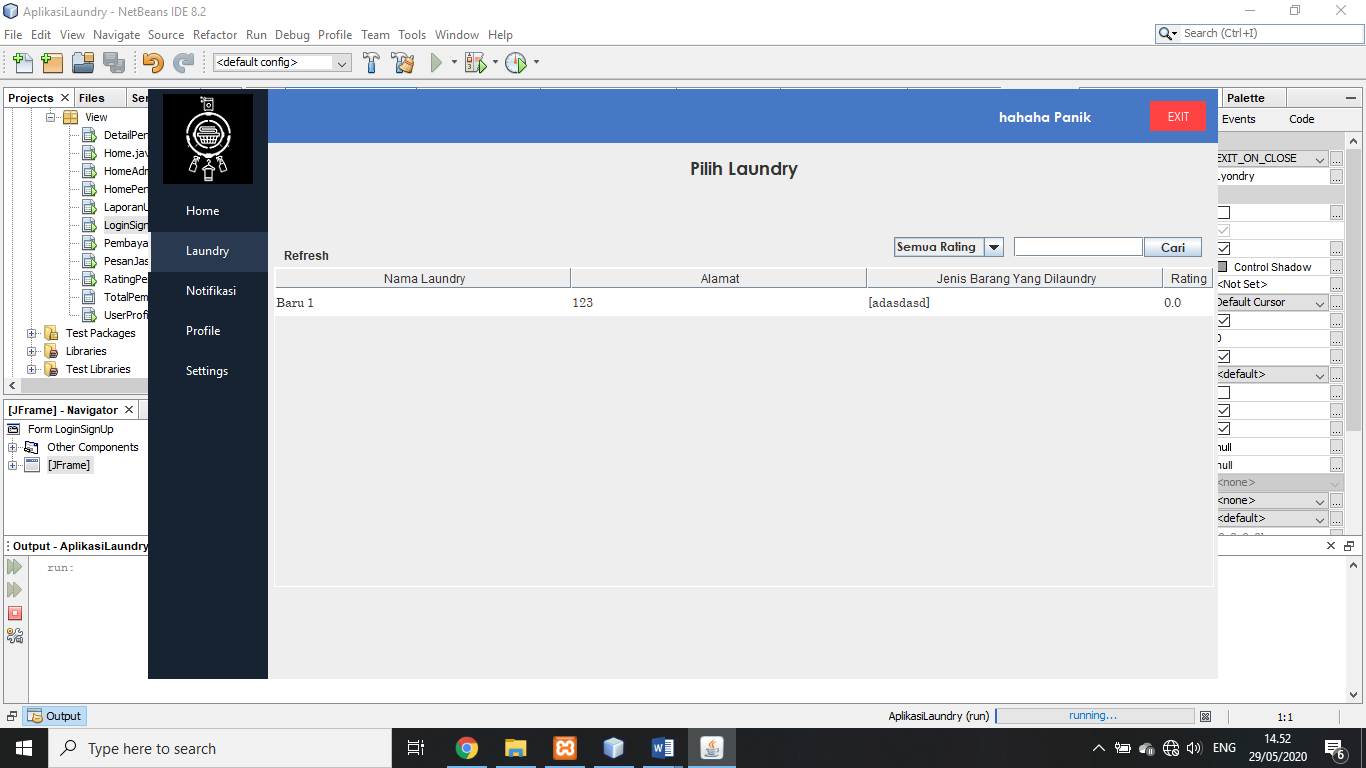
Jika label “Sign Up” pada Form Login ditekan maka akan terjadi animasi pertukaran form yaitu Form Sign Up akan ditampilkan sedangkan Form Login akan disembunyikan.

Urutkan langkah-langkah sebagai berikut:

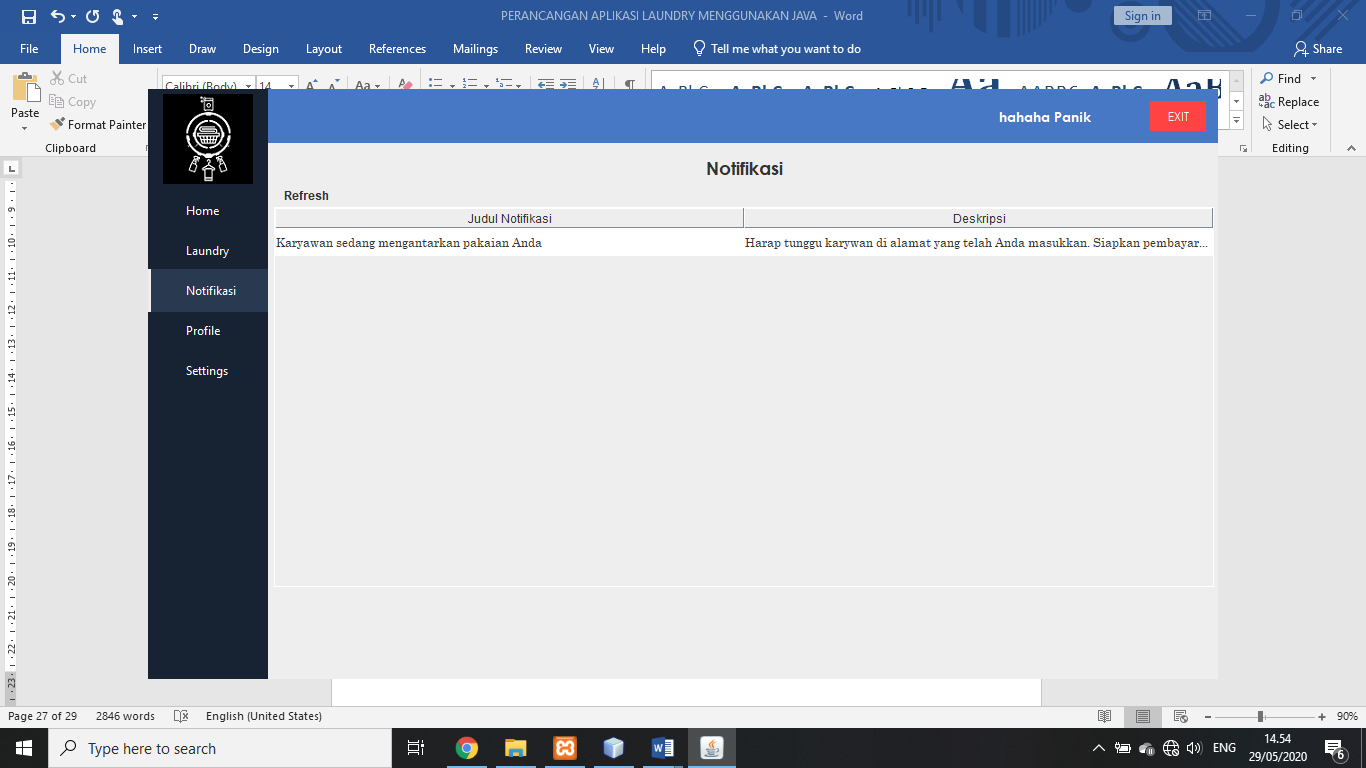
1. Masukkan data, jika data yang anda masukkan cocok setelah anda menekan tombol Sign Up maka anda langsung diarahkan ke view User.
2. Jika salah maka akan muncul peringatan dimana letak kesalahan Anda.
   * 1. View User



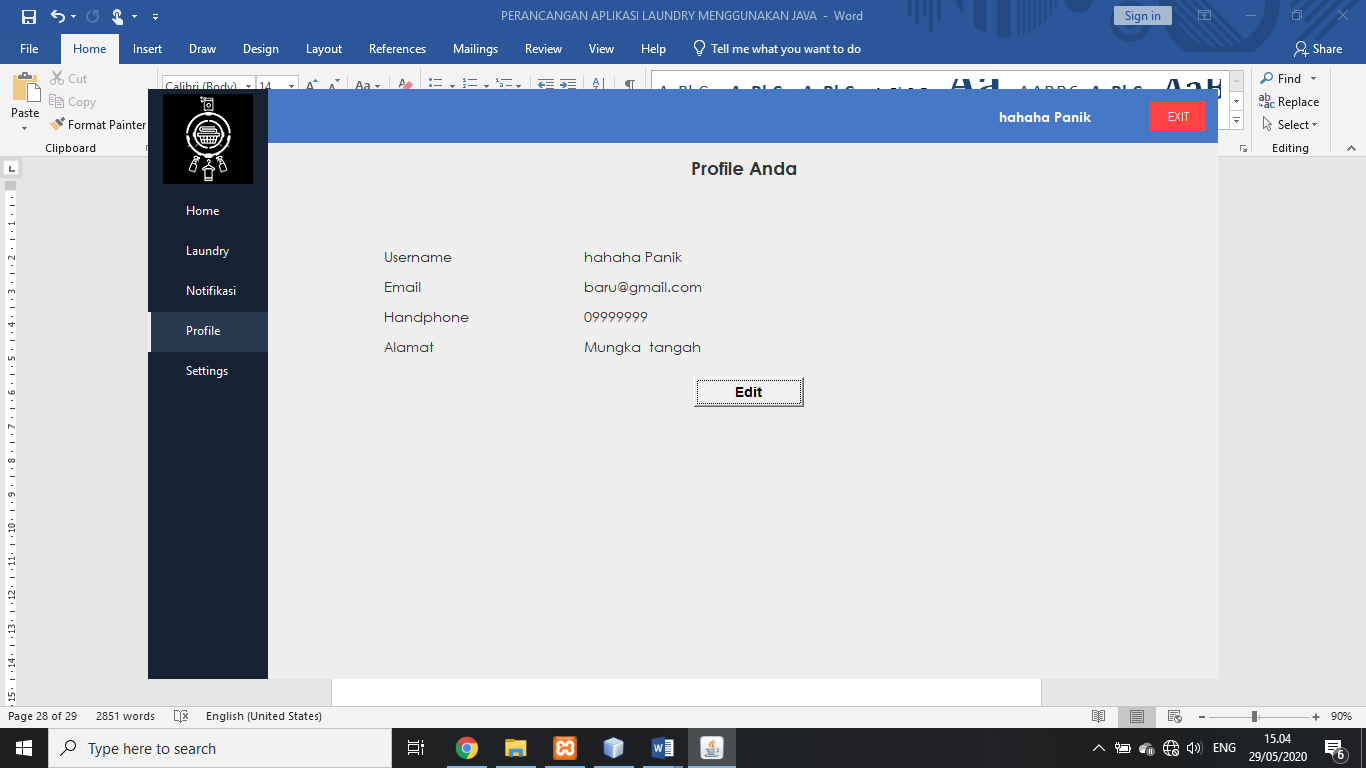
|  |
| --- |
| Gambar 3.3 Menu Home User |



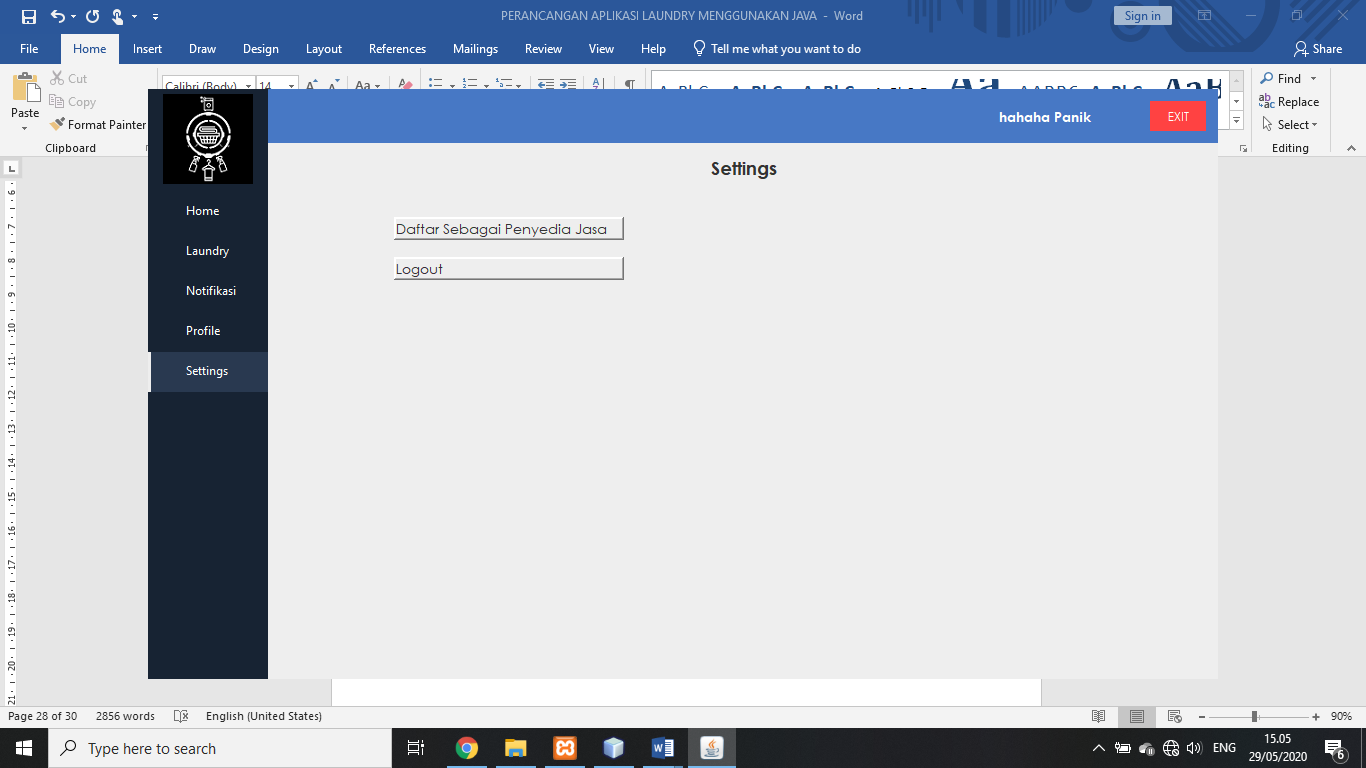
|  |
| --- |
| Gambar 3.4 Menu Laundry User |



|  |
| --- |
| Gambar 3.5 Menu Notifikasi User |



|  |
| --- |
| Gambar 3.6 Menu Profile User |

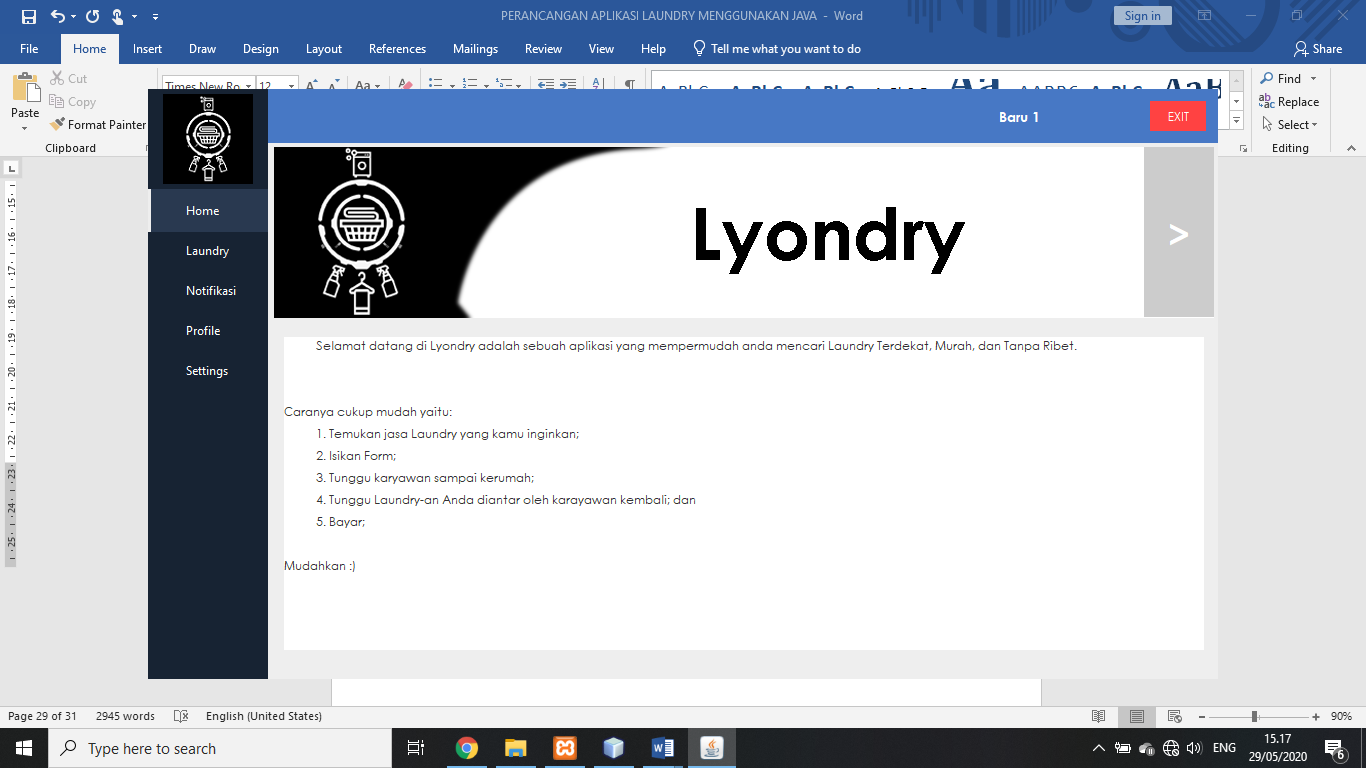


|  |
| --- |
| Gambar 3.7 Menu Settings User |

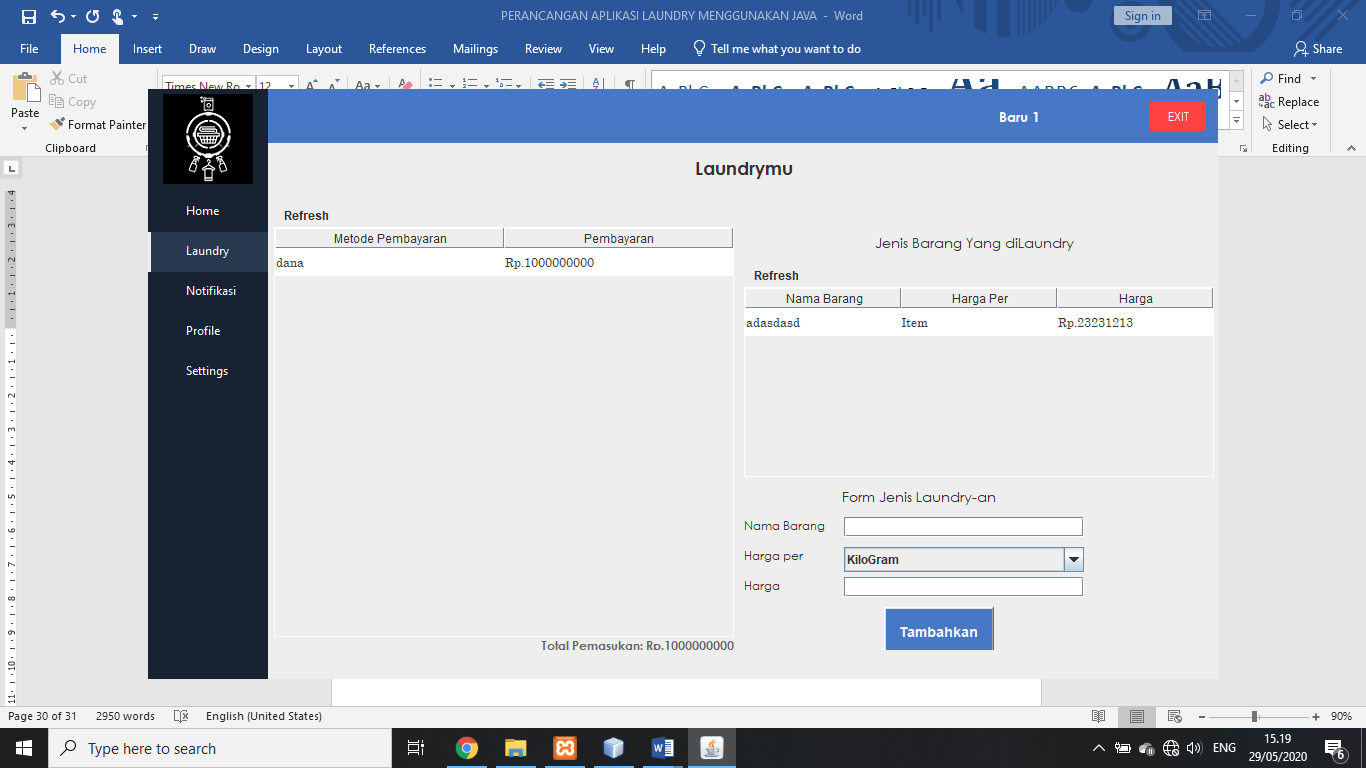
Gambar-gambar diatas merupakan tampilan setiap menu di View User.

Urutkan langkah-langkah di bawah ini:

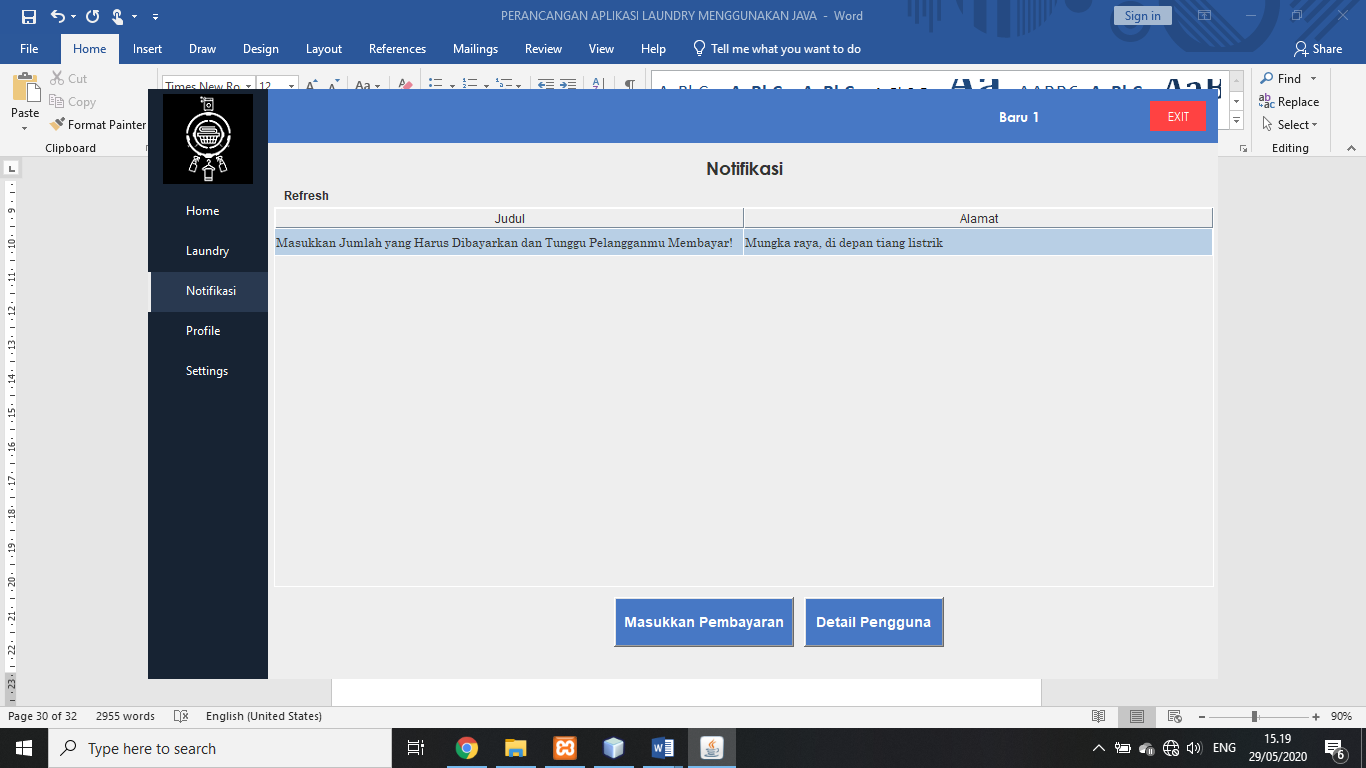
1. Ditampilan Home User bisa membaca bagaimana memesan Jasa Laundry di Aplikasi ini.
2. Ditampilan Laundry User bisa memesan Jasa Laundry.
3. Ditampilan Notifikasi User bisa melihat update-tan status Laundry-annya dan akhirnya membayar. Disini User juga bisa melihat detail Laundry Penyedia atau melaporkan Penyedia.
4. Ditampilan Profile User bisa melihat detail profilenya, atau mengedit profile serta passwordnya.
5. Ditampilan Settings User bisa melamar sebagai Penyedia dan Logout.
   * 1. View Penyedia



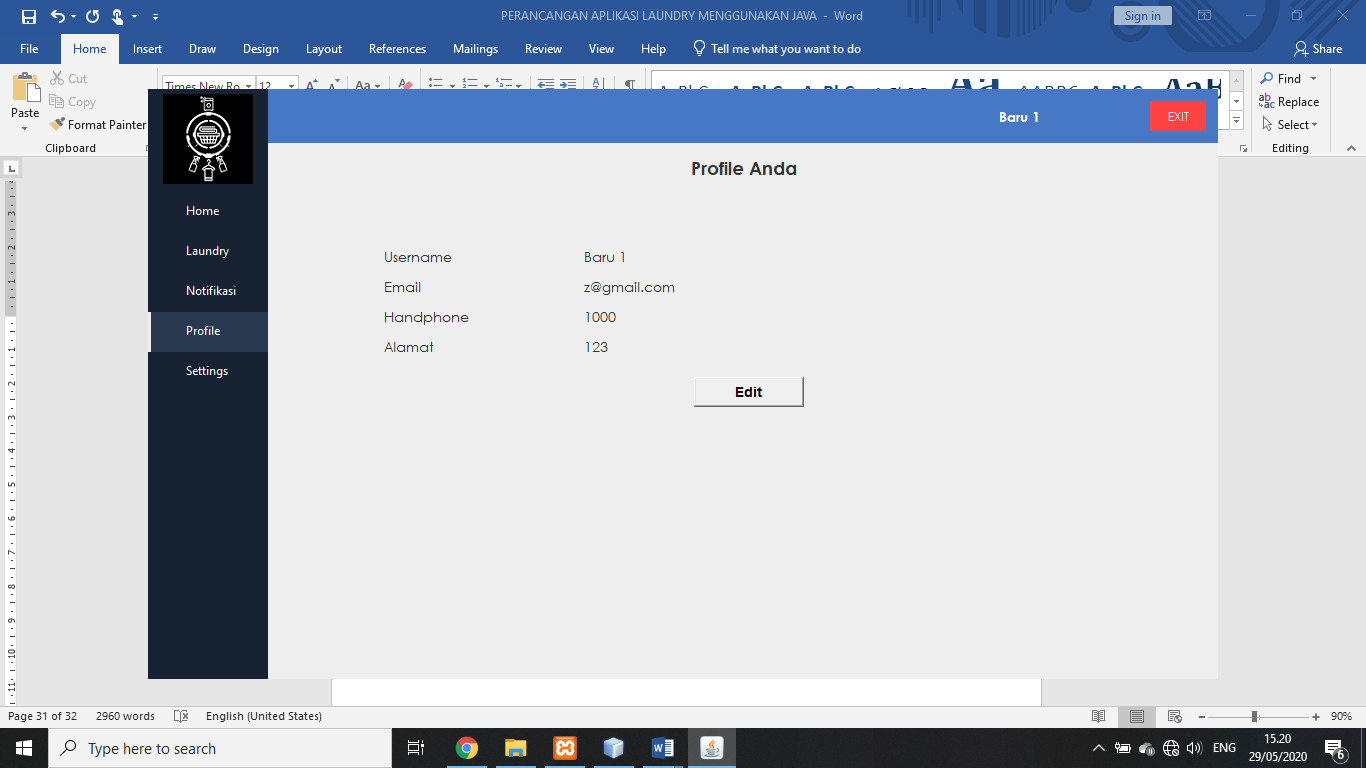
|  |
| --- |
| Gambar 3.8 Menu Home Penyedia |



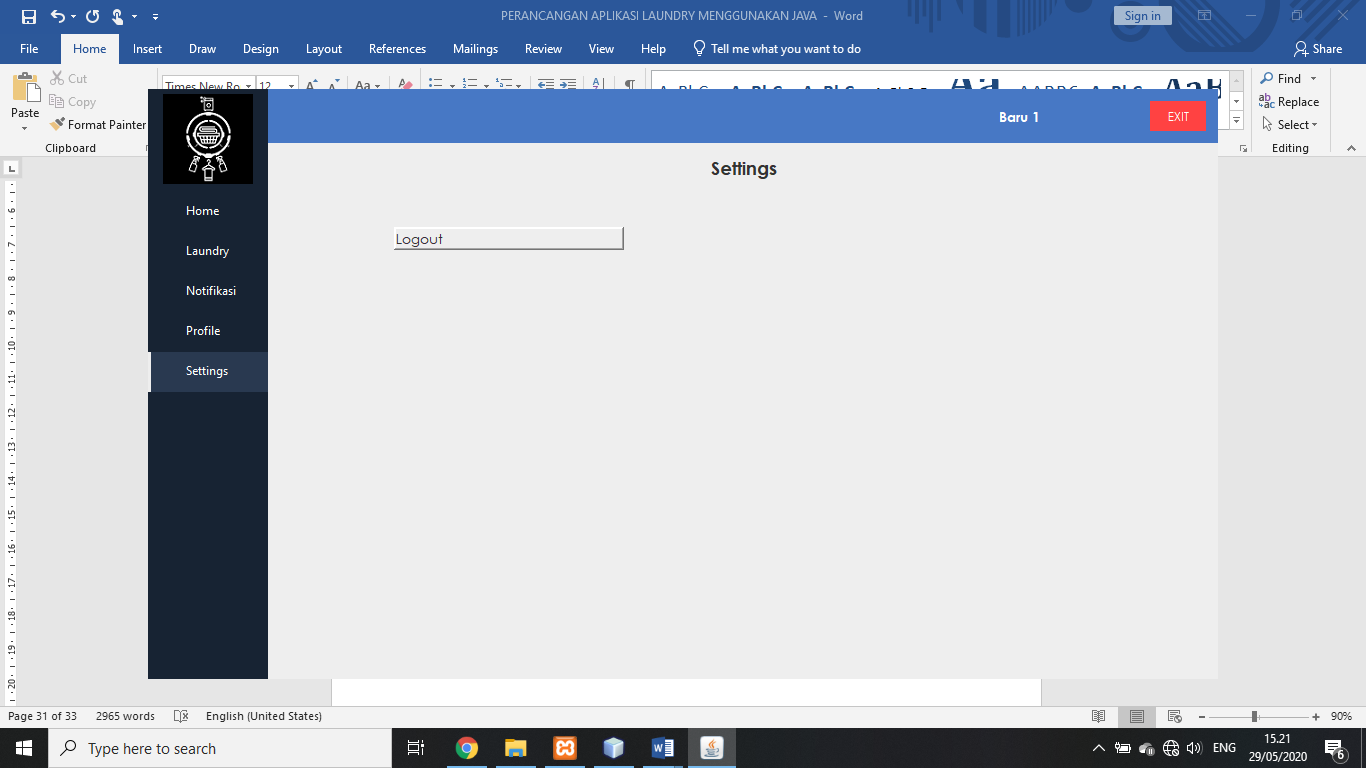
|  |
| --- |
| Gambar 3.9 Menu Laundry Penyedia |



|  |
| --- |
| Gambar 3.10 Menu Notifikasi Penyedia |



|  |
| --- |
| Gambar 3.11 Menu Profile Penyedia |

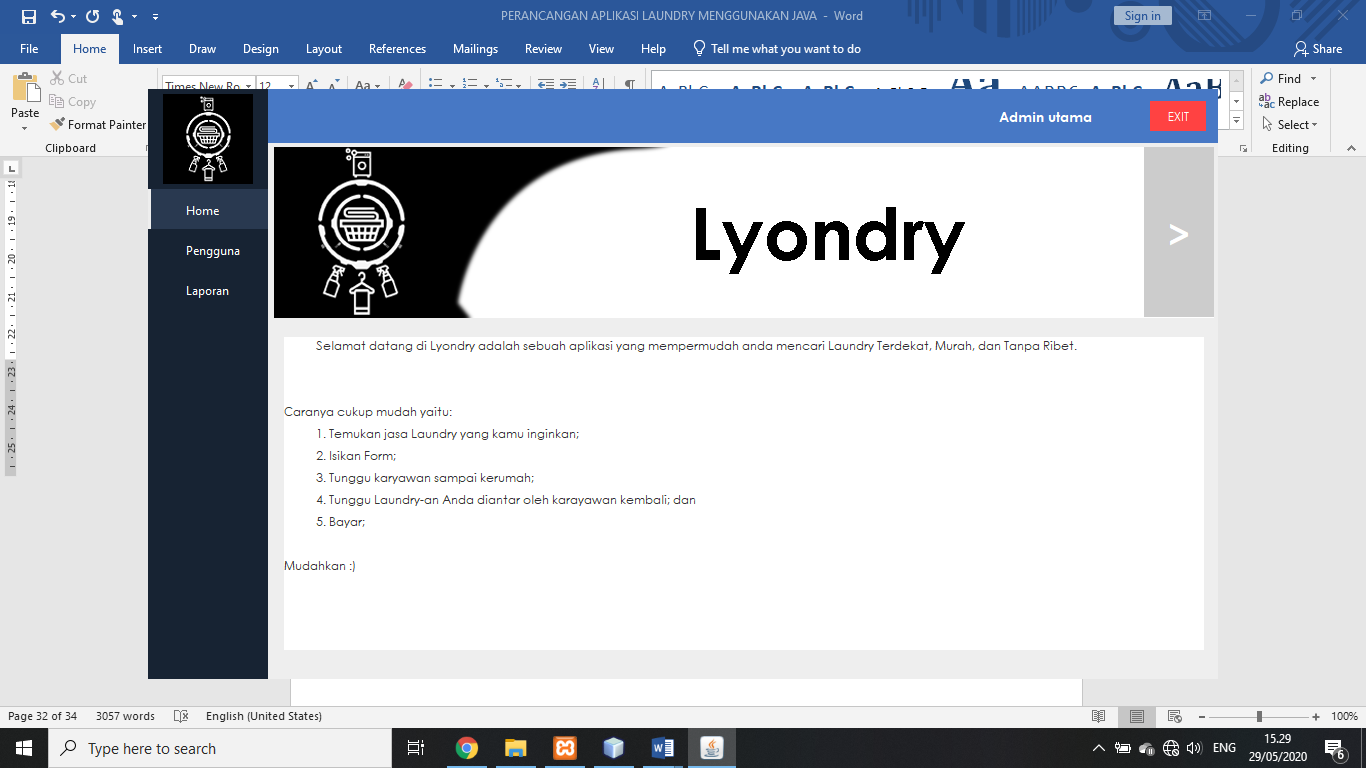


|  |
| --- |
| Gambar 3.12 Menu Settings Penyedia |

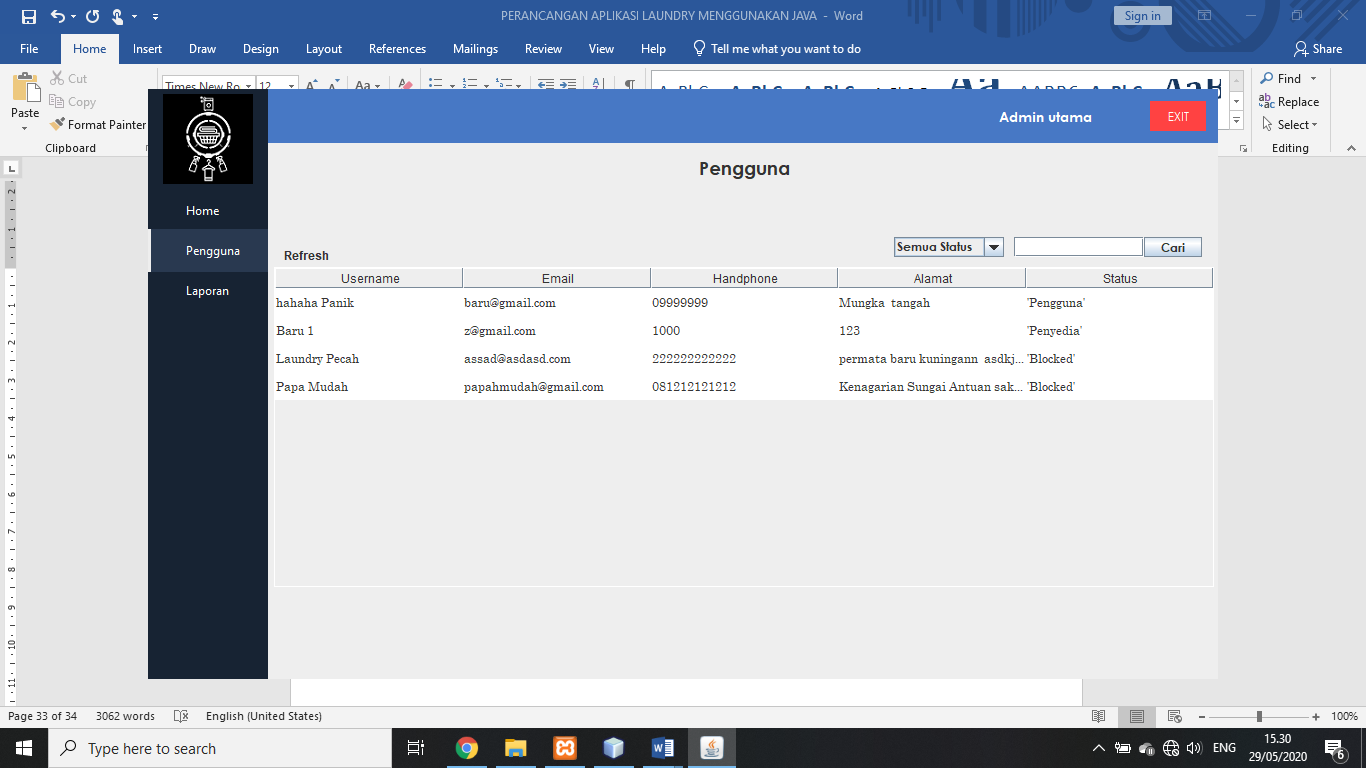
Gambar-gambar diatas merupakan tampilan setiap menu di View Penyedia.

Urutkan langkah-langkah di bawah ini:

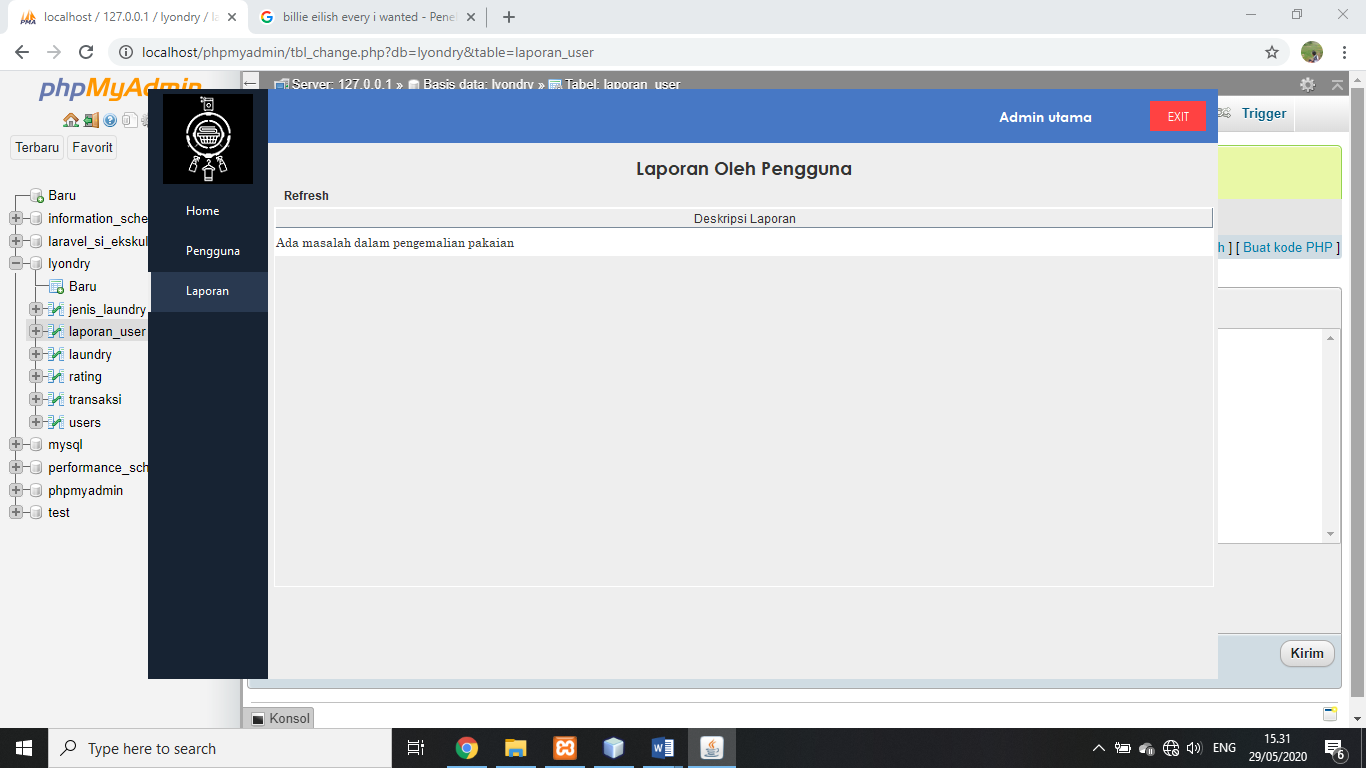
1. Ditampilan Home Penyedia bisa membaca bagaimana memesan Jasa Laundry di Aplikasi ini.
2. Ditampilan Laundry Penyedia bisa melihat pemasukan totalnya dan CRUD jenis laundry-nya.
3. Ditampilan Notifikasi Penyedia bisa melakukan update status Laundry dan akhirnya memasukkan total yang harus dibayar oleh User. Disini Penyedia juga bisa melihat detail Pelanggannya.
4. Ditampilan Profile Penyedia bisa melihat detail profilenya, atau mengedit profile serta passwordnya.
5. Ditampilan Settings Penyedia bisa melakukan Logout.
   * 1. View Admin



|  |
| --- |
| Gambar 3.13 Menu Home Admin |



|  |
| --- |
| Gambar 3.14 Menu Pengguna Admin |



|  |
| --- |
| Gambar 3.15 Menu Laporan Admin |

Gambar-gambar diatas merupakan tampilan setiap menu di View Admin.

Urutkan langkah-langkah di bawah ini:

1. Ditampilan Home Admin bisa membaca bagaimana memesan Jasa Laundry di Aplikasi ini.
2. Ditampilan Pengguna Admin bisa melihat semua user dan statusnya kecuali user yang statusnya admin. Serta bisa memblock akun yang berstatus pengguna dan penyedia.
3. Ditampilan Laporan Admin bisa melihat laporan ke penyedia oleh pengguna. Melihat detail pelapor dan yang melapor.

LAMPIRAN

Source Code:

1. Models
2. Connection.java

package Model;

import java.sql.DriverManager;

import java.sql.SQLException;

import javax.swing.JOptionPane;

public class Connection {

private static java.sql.Connection koneksi;

public static java.sql.Connection getkoneksi(){

if(koneksi==null){

try {

String url = new String();

String user = new String();

String password = new String();

url="jdbc:mysql://localhost:3306/lyondry";

user="root";

password="";

DriverManager.registerDriver(new com.mysql.jdbc.Driver());

koneksi=DriverManager.getConnection(url,user,password);

} catch (SQLException t) {

System.out.println("Oops! No internet");

JOptionPane.showMessageDialog(null, "Oops! No internet");

System.exit(0);

}

}

return koneksi;

}

}

1. JenisLaundry.java

package Model;

import java.sql.ResultSet;

import java.sql.Statement;

import javax.swing.JOptionPane;

public class JenisLaundry {

public static ResultSet getJenisLaundry(String id\_penyedia){

try{

Statement st = Model.Connection.getkoneksi().createStatement();

String queryJL = "SELECT \* FROM jenis\_laundry WHERE id\_user='"+id\_penyedia+"'";

ResultSet JL = st.executeQuery(queryJL);

return JL;

} catch(Exception e){

{System.err.println(e.getMessage());}

}

return null;

}

public static ResultSet getDataJenisLaundry(){

try{

Statement st = Model.Connection.getkoneksi().createStatement();

String queryJL = "SELECT \* FROM jenis\_laundry ORDER BY id\_jenislaundry DESC";

ResultSet JL = st.executeQuery(queryJL);

return JL;

} catch(Exception e){

{System.err.println(e.getMessage());}

}

return null;

}

public void fullDeleteJenisLaundy(String id\_jenislaundry){

try{

Statement st = Model.Connection.getkoneksi().createStatement();

String ls = "DELETE FROM jenis\_laundry WHERE id\_jenislaundry='"+id\_jenislaundry+"'";

if(!st.execute(ls)){

JOptionPane.showMessageDialog(null, "Jenis Laundry berhasil dihapus!");

}

}catch(Exception e){

JOptionPane.showMessageDialog(null, "Tampaknya ada masalah dalam proses penghapusan");

}

}

public void addJenisLaundry(String id\_penyedia, String barang, String harga\_perkg, String harga\_peritem){

try{

Statement st = Model.Connection.getkoneksi().createStatement();

String rg = "";

if(harga\_perkg == "NULL"){

rg = "INSERT INTO jenis\_laundry(barang, harga\_perkg, harga\_peritem, id\_user) "

+" VALUES ('"

+barang+"',NULL,'"

+harga\_peritem+"','"

+id\_penyedia+"')";

}else{

rg = "INSERT INTO jenis\_laundry(barang, harga\_perkg, harga\_peritem, id\_user) "

+" VALUES ('"

+barang+"','"

+harga\_perkg+"',NULL,'"

+id\_penyedia+"')";

}

if(!st.execute(rg)){

JOptionPane.showMessageDialog(null, "Berhasil ditambahkan");

}else{

JOptionPane.showMessageDialog(null, "Gagal ditambahkan");

}

}catch(Exception e){

JOptionPane.showMessageDialog(null, e);

}

}

}

1. LaporanUserM.java

package Model;

import java.sql.ResultSet;

import java.sql.Statement;

import javax.swing.JOptionPane;

public class LaporanUserM {

public void reportPenyedia(String id\_laundry, String deskripsi){

try{

Statement st = Model.Connection.getkoneksi().createStatement();

String rg = "INSERT INTO laporan\_user (id\_laundry,deskripsi\_laporan) VALUES ('"

+id\_laundry+"','"+deskripsi+"')";

String ls = "UPDATE laundry SET status = '-1' WHERE id\_laundry='"+id\_laundry+"'";

if(!st.execute(rg) && !st.execute(ls)){

JOptionPane.showMessageDialog(null, "Penyedia berhasil dilaporkan!");

}

}catch(Exception e){

JOptionPane.showMessageDialog(null, "Tampaknya Anda sudah melaporkan penyedia ini");

}

}

public void deleteNotif(String id\_laundry){

try{

Statement st = Model.Connection.getkoneksi().createStatement();

String ls = "UPDATE laundry SET id\_user = NULL WHERE id\_laundry='"+id\_laundry+"'";

if(!st.execute(ls)){

JOptionPane.showMessageDialog(null, "Notif berhasil dihapus!");

}

}catch(Exception e){

JOptionPane.showMessageDialog(null, "Tampaknya ada masalah dalam proses penghapusan");

}

}

public static ResultSet getLaporan(){

try{

Statement st = Model.Connection.getkoneksi().createStatement();

String queryLaporan = "SELECT laporan\_user.id\_laporanuser, laundry.id\_penyedia, laundry.id\_user, laporan\_user.deskripsi\_laporan FROM laporan\_user LEFT JOIN laundry ON laporan\_user.id\_laundry = laundry.id\_laundry ORDER BY id\_laporanuser DESC";

ResultSet laporan = st.executeQuery(queryLaporan);

return laporan;

} catch(Exception e){

{System.err.println(e.getMessage());}

}

return null;

}

public void fullDeleteNotif(String id\_laporanuser){

try{

Statement st = Model.Connection.getkoneksi().createStatement();

String ls = "DELETE FROM laporan\_user WHERE id\_laporanuser='"+id\_laporanuser+"'";

if(!st.execute(ls)){

JOptionPane.showMessageDialog(null, "Laporan berhasil dihapus!");

}

}catch(Exception e){

JOptionPane.showMessageDialog(null, "Tampaknya ada masalah dalam proses penghapusan");

}

}

}

1. Laundry.java

package Model;

import java.sql.ResultSet;

import java.sql.Statement;

import javax.swing.JOptionPane;

public class Laundry {

public static Integer statusForm = 0;

public void submit(String usernamePelaundry, String id\_jenislaundry, String handphone, String alamat, String id\_penyedia){

try{

Statement st = Model.Connection.getkoneksi().createStatement();

ResultSet rs = st.executeQuery("select \* from users where username='"+usernamePelaundry+"' AND handphone='"+handphone+"'");

if(rs.next()){

String rg = "INSERT INTO laundry(alamat, id\_user, id\_penyedia, id\_jenislaundry, status) "

+ "VALUE ('"

+alamat+"','"

+rs.getString("id\_user")+"','"

+id\_penyedia+"','"

+id\_jenislaundry+"','0')";

System.out.println(rs.getString("id\_user"));

if(!st.execute(rg)){

JOptionPane.showMessageDialog(null, "Form sudah dikirimkan silahkan lihat notifikasi anda!");

}else{

JOptionPane.showMessageDialog(null, "Masukkan form dengan benar");

}

statusForm = 1;

}

}catch(Exception e){

statusForm = 0;

JOptionPane.showMessageDialog(null, "Maaf Anda tidak memesan jasa Laundry 2 kali.");

}

}

public ResultSet getLaundry(String usernamePelaundry, String handphone){

ResultSet rs = null;

try{

Statement st = Model.Connection.getkoneksi().createStatement();

ResultSet rsu = st.executeQuery("SELECT \* FROM users where username='"+usernamePelaundry+"' AND handphone='"+handphone+"'");

if(rsu.next()){

rs = st.executeQuery("SELECT laundry.id\_laundry,laundry.id\_penyedia,users.username,laundry.status "

+"FROM laundry LEFT JOIN users ON laundry.id\_penyedia = users.id\_user "

+"WHERE laundry.id\_user='"+rsu.getString("id\_user")+"' ORDER BY id\_laundry DESC");

}

}catch(Exception e){System.out.println("Error saat pengambilan data deskripsi\t"+e);}

return rs;

}

public void setStatus(String id\_laundry, String status){

try{

Statement st = Model.Connection.getkoneksi().createStatement();

String ls = "UPDATE laundry SET status = '"+status+"' WHERE id\_laundry='"+id\_laundry+"'";

if(!st.execute(ls)){

JOptionPane.showMessageDialog(null, "Notif berhasil diubah!");

}

}catch(Exception e){

JOptionPane.showMessageDialog(null, "Tampaknya ada masalah dalam proses pengubahan");

}

}

public ResultSet getDataLaundry(String id\_penyedia){

ResultSet rs = null;

try{

Statement st = Model.Connection.getkoneksi().createStatement();

rs = st.executeQuery("SELECT laundry.id\_laundry,laundry.id\_user,laundry.status,laundry.alamat "

+"FROM laundry WHERE laundry.id\_penyedia='"+id\_penyedia+"' ORDER BY id\_laundry DESC");

}catch(Exception e){System.out.println("Error saat pengambilan data deskripsi\t");}

return rs;

}

public void setTotalPembayaran(String id\_laundry, String total\_pembayaran){

try{

Statement st = Model.Connection.getkoneksi().createStatement();

String ls = "UPDATE laundry SET total\_pembayaran = '"+total\_pembayaran+"' WHERE id\_laundry='"+id\_laundry+"'";

if(!st.execute(ls)){

JOptionPane.showMessageDialog(null, "Berhasil diubah!");

}

}catch(Exception e){

JOptionPane.showMessageDialog(null, "Tampaknya ada masalah dalam proses pengubahan");

}

}

}

1. Rating.java

package Model;

import java.sql.ResultSet;

import java.sql.Statement;

import javax.swing.JOptionPane;

public class Rating {

public static ResultSet getAverageRatingPenyedia(String id\_penyedia){

try{

Statement st = Model.Connection.getkoneksi().createStatement();

String queryR = "SELECT AVG(rating) FROM rating WHERE id\_penyedia='"+id\_penyedia+"'";

ResultSet R = st.executeQuery(queryR);

return R;

} catch(Exception e){

{System.err.println(e.getMessage());}

}

return null;

}

public void ratingPenyedia(String bintang, String id\_penyedia, String usernamePelaundry, String handphone){

try{

Statement st = Model.Connection.getkoneksi().createStatement();

ResultSet rs = st.executeQuery("select \* from users where username='"+usernamePelaundry+"' AND handphone='"+handphone+"'");

if(rs.next()){

String queryR = "INSERT INTO rating (rating,id\_penyedia,id\_user) VALUES ('"+

bintang+"','"+

id\_penyedia+"','"+

rs.getString("id\_user")+"');";

if(!st.execute(queryR)){

JOptionPane.showMessageDialog(null, "Terima Kasih, jangan lupa memesan Jasa Laundry lagi di Lyoky.");

}

}

} catch(Exception e){

{System.err.println(e.getMessage());}

}

}

}

1. Transaksi.java

package Model;

import java.sql.ResultSet;

import java.sql.Statement;

import javax.swing.JOptionPane;

public class Transaksi {

public ResultSet totalPembayaran(String id\_laundry){

try{

Statement st = Model.Connection.getkoneksi().createStatement();

ResultSet rs = st.executeQuery("SELECT \* FROM laundry WHERE id\_laundry='"+id\_laundry+"'");

return rs;

}catch(Exception e){

}

return null;

}

public void tambahkanPembayaran(String id\_laundry, String metode\_pembayaran, String total\_pembayaran){

try{

Statement st = Model.Connection.getkoneksi().createStatement();

String rg = "INSERT INTO transaksi(id\_laundry, metode\_pembayaran) "

+ "VALUE ('"

+id\_laundry+"','"

+metode\_pembayaran+"')";

if(!st.execute(rg)){

JOptionPane.showMessageDialog(null, "Anda akan membayar sebesar Rp."+total\_pembayaran);

new Model.Laundry().setStatus(id\_laundry, "3");

}

}catch(Exception e){}

}

public ResultSet getDataLaporanPembayaran(){

try{

Statement st = Model.Connection.getkoneksi().createStatement();

ResultSet rs = st.executeQuery("SELECT laundry.id\_penyedia, transaksi.metode\_pembayaran, laundry.total\_pembayaran FROM transaksi LEFT JOIN laundry ON transaksi.id\_laundry = laundry.id\_laundry ORDER BY id\_transaksi DESC");

return rs;

}catch(Exception e){}

return null;

}

}

1. User.java

package Model;

import java.sql.ResultSet;

import java.sql.Statement;

import javax.swing.JOptionPane;

public class User{

public int loginStatus;

public int statusLaundry = -1;

public void signUp(String username, String email, String handphone, String password, String alamat){

try{

Statement st = Model.Connection.getkoneksi().createStatement();

ResultSet rs = st.executeQuery("select \* from users where email='"+email+"' OR handphone='"+handphone+"'");

String rg = "INSERT INTO users(username, email, handphone, password, alamat, status) "

+ "VALUE ('"

+username+"','"

+email+"','"

+handphone+"','"

+password+"','"

+alamat+"','"

+"0')";

if(rs.next()){

JOptionPane.showMessageDialog(null, "Email atau Nomor Handphone yang Anda masukkan sudah dipakai!");

loginStatus = -2;

}else{

if(!st.execute(rg)){

JOptionPane.showMessageDialog(null, "Sekarang username "+username+" telah didaftarkan!!");

setProfile(username,email,handphone,alamat);

loginStatus = 0;

} else {

loginStatus = -2;

JOptionPane.showMessageDialog(null, "Harap periksa Koneksi!");

}

}

}catch(Exception e){

loginStatus = -2;

JOptionPane.showMessageDialog(null, "Harap periksa Koneksi!");

}

}

public void Login(String email, String password){

try{

Statement st = Model.Connection.getkoneksi().createStatement();

ResultSet rs = st.executeQuery("select \* from users where email='"+email+"' AND password='"+password+"'"

);

if(rs.next()){

if(rs.getInt("status") == 2){

loginStatus = 2;

}else if(rs.getInt("status") == 1){

loginStatus = 1;

}else if(rs.getInt("status") == 0){

loginStatus = 0;

}else{

loginStatus = -1;

}

if(rs.getInt("status") == 0){

setProfile(rs.getString("username"),rs.getString("email"),rs.getString("handphone"),rs.getString("alamat"));

}else if(rs.getInt("status") == 1){

setProfilePenyedia(rs.getString("username"),rs.getString("email"),rs.getString("handphone"),rs.getString("alamat"));

}else if(rs.getInt("status") == 2){

setProfileAdmin(rs.getString("username"),rs.getString("email"),rs.getString("handphone"),rs.getString("alamat"));

}

}else{

JOptionPane.showMessageDialog(null, "Email atau Password yang anda masukkan salah!");

loginStatus = -2;

}

}catch(Exception e){

JOptionPane.showMessageDialog(null, "Harap periksa Koneksi!");

}

}

public static ResultSet getUserPenyedia(String userLogin, String cari){

try{

Statement st = Model.Connection.getkoneksi().createStatement();

String queryPenyedia = "";

if(cari.equals(null) || cari.equals("")){

queryPenyedia = "SELECT \* FROM users WHERE NOT email='"+userLogin+"' AND status='1'";

}else{

queryPenyedia = "SELECT \* FROM users WHERE NOT email='"+userLogin+"' AND status='1' AND (username LIKE '%"+cari+"%' OR alamat LIKE '%"+cari+"%')";

}

ResultSet penyedia = st.executeQuery(queryPenyedia);

return penyedia;

} catch(Exception e){

{System.err.println(e.getMessage());}

}

return null;

}

private void setProfile(String username, String email, String handphone, String alamat){

Controller.UserHome.username = username;

Controller.UserHome.email = email;

Controller.UserHome.handphone = handphone;

Controller.UserHome.alamat = alamat;

}

private void setProfileAdmin(String username, String email, String handphone, String alamat){

Controller.AdminHome.username = username;

Controller.AdminHome.email = email;

Controller.AdminHome.handphone = handphone;

Controller.AdminHome.alamat = alamat;

}

private void setProfilePenyedia(String username, String email, String handphone, String alamat){

Controller.PenyediaHome.username = username;

Controller.PenyediaHome.email = email;

Controller.PenyediaHome.handphone = handphone;

Controller.PenyediaHome.alamat = alamat;

}

public static ResultSet detailPenyedia(String id\_penyedia){

ResultSet rs = null;

try{

Statement st = Model.Connection.getkoneksi().createStatement();

String queryPenyedia = "SELECT \* FROM users WHERE id\_user='"+id\_penyedia+"'";

rs = st.executeQuery(queryPenyedia);

}catch(Exception e){}

return rs;

}

public void editProfile(String emailLama, String handphoneLama, String username, String email, String handphone, String password, String alamat, String status){

try{

Statement st = Model.Connection.getkoneksi().createStatement();

Statement sta = Model.Connection.getkoneksi().createStatement();

ResultSet rse = null;

ResultSet rsh = null;

if(!emailLama.equals(email)){

rse = st.executeQuery("SELECT \* FROM users WHERE email='"+email+"'");

}

if(!handphoneLama.equals(handphone)){

rsh = sta.executeQuery("SELECT \* FROM users WHERE handphone='"+handphone+"'");

}

String rg = "";

if(password.equals("")){

rg = "UPDATE users SET username='"+username+

"', email='"+email+

"', handphone='"+handphone+

"', alamat='"+alamat+

"', status='"+status+

"' WHERE email='"+emailLama+"'";

}else{

rg = "UPDATE users SET username='"+username+

"', email='"+email+

"', handphone='"+handphone+

"', password='"+password+

"', alamat='"+alamat+

"', status='"+status+

"' WHERE email='"+emailLama+"'";

}

if(rse != null && rse.next()){

JOptionPane.showMessageDialog(null, "Email yang Anda masukkan sudah dipakai!");

}else if(rsh != null && rsh.next()){

JOptionPane.showMessageDialog(null, "Nomor Handphone yang Anda masukkan sudah dipakai!");

}else{

if(!st.execute(rg)){

JOptionPane.showMessageDialog(null, "Berhasil diubah!");

setProfile(username,email,handphone,alamat);

} else {

JOptionPane.showMessageDialog(null, "Harap periksa Koneksi!");

}

}

}catch(Exception e){}

}

public void cekLaundry(String emailLama, String handphoneLama){

try{

Statement st = Model.Connection.getkoneksi().createStatement();

ResultSet rse = st.executeQuery("SELECT \* FROM users WHERE email='"+emailLama+"' AND handphone='"+handphoneLama+"'");

if(rse.next()){

ResultSet rsr = st.executeQuery("SELECT \* FROM laundry WHERE id\_user='"+rse.getString("id\_user")+"'");

if(rsr.next()){

statusLaundry = 1;

}else{

statusLaundry = 0;

}

}

}catch(Exception e){}

}

public static ResultSet getPengguna(String cari){

try{

Statement st = Model.Connection.getkoneksi().createStatement();

String queryPengguna = "";

if(cari.equals(null) || cari.equals("")){

queryPengguna = "SELECT \* FROM users WHERE NOT status='2'";

}else{

queryPengguna = "SELECT \* FROM users WHERE NOT status='2' AND (username LIKE '%"+cari+"%' OR alamat LIKE '%"+cari+"%' OR email LIKE '%"+cari+"%' OR handphone LIKE '%"+cari+"%')";

}

ResultSet penyedia = st.executeQuery(queryPengguna);

return penyedia;

} catch(Exception e){

{System.err.println(e.getMessage());}

}

return null;

}

public void setStatus(String id\_user, String status){

try{

Statement st = Model.Connection.getkoneksi().createStatement();

String ls = "UPDATE users SET status = '"+status+"' WHERE id\_user='"+id\_user+"'";

if(!st.execute(ls)){

JOptionPane.showMessageDialog(null, "Status berhasil diubah!");

}

}catch(Exception e){

JOptionPane.showMessageDialog(null, "Tampaknya ada masalah dalam proses pengubahan");

}

}

public ResultSet getIdUser(String email, String handphone){

try{

Statement st = Model.Connection.getkoneksi().createStatement();

ResultSet rs = st.executeQuery("SELECT \* FROM users WHERE email='"+email+"' AND handphone='"+handphone+"'");

return rs;

}catch(Exception e){}

return null;

}

}

1. Views
2. DetailPenyedia.java

package View;

import java.awt.Dimension;

import java.awt.Toolkit;

import Controller.DetailPenyediaC;

import javax.swing.JLabel;

import javax.swing.JTextArea;

public class DetailPenyedia extends javax.swing.JFrame {

private DetailPenyediaC DetailPenyediaC;

public DetailPenyedia() {

initComponents();

Dimension layar = Toolkit.getDefaultToolkit().getScreenSize();

this.setLocation(layar.width / 2 - this.getSize().width / 2, layar.height / 2 - this.getSize().height / 2);

DetailPenyediaC = new DetailPenyediaC(this);

}

private void formWindowClosed(java.awt.event.WindowEvent evt) {

for (double i = 1.0; i >=1.0;i = i - 0.2)

{

String val = i+ "";

float f = Float.valueOf(val);

this.setOpacity(f);

try{

Thread.sleep(50);

}

catch(Exception e){}

}

new View.Home().setVisible(true);

}

private void formWindowOpened(java.awt.event.WindowEvent evt) {

//

}

public JLabel username(){

return jLabel6;

}

public JLabel handphone(){

return jLabel7;

}

public JLabel email(){

return jLabel8;

}

public JTextArea alamat(){

return jTextArea1;

}

1. Home.java

package View;

import AppPackage.AnimationClass;

import java.awt.Color;

import java.awt.Dimension;

import java.awt.Toolkit;

import javax.swing.JPanel;

import javax.swing.table.DefaultTableModel;

import javax.swing.JLabel;

import Controller.UserHome;

import Controller.PesanJasa;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.JOptionPane;

import javax.swing.JTable;

public class Home extends javax.swing.JFrame {

public static DefaultTableModel modeltab1, modeltab2;

int xx,xy;

private UserHome UserHome;

private PesanJasa PesanJasa;

public static String username;

private Integer ratingLaundry = 0;

private String cariLaundry = "";

public Home() {

initComponents();

setColor(btn\_1,ind\_1,show\_1);

ind\_1.setOpaque(true);

Dimension layar = Toolkit.getDefaultToolkit().getScreenSize();

this.setLocation(layar.width / 2 - this.getSize().width / 2, layar.height / 2 - this.getSize().height / 2);

UserHome = new UserHome(this);

}

private void btn\_1MousePressed(java.awt.event.MouseEvent evt) {

setColor(btn\_1,ind\_1,show\_1);

}

private void btn\_2MouseReleased(java.awt.event.MouseEvent evt) {

setColor(btn\_2,ind\_2,show\_2);

}

private void btn\_3MousePressed(java.awt.event.MouseEvent evt) {

setColor(btn\_3,ind\_3,show\_3);

}

private void btn\_4MousePressed(java.awt.event.MouseEvent evt) {

setColor(btn\_4,ind\_4,show\_4);

}

private void btn\_5MouseReleased(java.awt.event.MouseEvent evt) {

setColor(btn\_5,ind\_5,show\_5);

}

private void jPanel2MousePressed(java.awt.event.MouseEvent evt) {

xx = evt.getX();

xy = evt.getY();

}

private void jPanel2MouseDragged(java.awt.event.MouseEvent evt) {

int x = evt.getXOnScreen();

int y = evt.getYOnScreen();

this.setLocation(x-xx,y-xy);

}

private void jLabel1MouseClicked(java.awt.event.MouseEvent evt) {

for (double i = 1.0; i >=0.0;i = i - 0.2)

{

String val = i+ "";

float f = Float.valueOf(val);

this.setOpacity(f);

try{

Thread.sleep(50);

}

catch(Exception e){}

}

System.exit(0);

}

private void formWindowOpened(java.awt.event.WindowEvent evt) {

for (double i = 0.0; i <=1.0;i = i + 0.2)

{

String val = i+ "";

float f = Float.valueOf(val);

this.setOpacity(f);

try{

Thread.sleep(50);

}

catch(Exception e){}

}

}

private void jLabel3MouseClicked(java.awt.event.MouseEvent evt) {

previousImage();

}

private void jPanel4MouseEntered(java.awt.event.MouseEvent evt) {

jPanel4.setBackground(new Color(71,120,197));

}

private void jPanel4MouseExited(java.awt.event.MouseEvent evt) {

jPanel4.setBackground(new Color(255,66,66));

}

private void jLabel1MouseEntered(java.awt.event.MouseEvent evt) {

jPanel4.setBackground(new Color(71,120,197));

}

private void jLabel1MouseExited(java.awt.event.MouseEvent evt) {

jPanel4.setBackground(new Color(255,66,66));

}

private void jLabel32MouseEntered(java.awt.event.MouseEvent evt) {

jLabel32.setText("Logout");

}

private void jLabel32MouseExited(java.awt.event.MouseEvent evt) {

jLabel32.setText(username);

}

private void jLabel32MouseClicked(java.awt.event.MouseEvent evt) {

new View.LoginSignUp().setVisible(true);

this.dispose();

}

private void jLabel29MouseClicked(java.awt.event.MouseEvent evt) {

new View.LoginSignUp().setVisible(true);

this.dispose();

}

private void formWindowActivated(java.awt.event.WindowEvent evt) {

Thread p = new Thread(){

public void run(){

for(;;){

try{

if(jTable1.getSelectedRow() > -1){

button4.setVisible(true);

}else{

button4.setVisible(false);

}

if(jTable2.getSelectedRow() > -1){

if(jTable2.getValueAt(jTable2.getSelectedRow(), 4).toString().equals("0")){

button5.setVisible(true);

button2.setVisible(true);

button6.setVisible(false);

}else if(jTable2.getValueAt(jTable2.getSelectedRow(), 4).toString().equals("1")){

button5.setVisible(true);

button2.setVisible(true);

button6.setVisible(false);

}else if(jTable2.getValueAt(jTable2.getSelectedRow(), 4).toString().equals("2")){

button5.setVisible(true);

button2.setVisible(true);

button6.setVisible(true);

button6.setLabel("Bayar");

}else if(jTable2.getValueAt(jTable2.getSelectedRow(), 4).toString().equals("3")){

button2.setVisible(true);

button6.setVisible(true);

button6.setLabel("Rating&Hapus");

button5.setVisible(false);

}else if(jTable2.getValueAt(jTable2.getSelectedRow(), 4).toString().equals("-1")){

button6.setVisible(true);

button6.setLabel("Hapus");

button5.setVisible(false);

button2.setVisible(false);

}

}else{

button5.setVisible(false);

button2.setVisible(false);

button6.setVisible(false);

}

sleep(100);

}catch(InterruptedException ex){

Logger.getLogger(View.Home.class.getName()).log(Level.SEVERE, null, ex);

}

}

}

};

p.start();

jLabel32.setText(username);

}

private void button4ActionPerformed(java.awt.event.ActionEvent evt) {

if(jTable1.getSelectedRow() >= 0 && modeltab1.getValueAt(jTable1.getSelectedRow(), 3) != "[]"){

UserHome.openPesanJasaForm(jTable1.getValueAt(jTable1.getSelectedRow(), 0).toString());

}else if(modeltab1.getValueAt(jTable1.getSelectedRow(), 3) == "[]"){

JOptionPane.showMessageDialog(null, "Tampaknya Penyedia belum memasukkan jenis laundrynya.");

}

}

private void jComboBox1ItemStateChanged(java.awt.event.ItemEvent evt) {

if(jComboBox1.getSelectedIndex() == 0){

ratingLaundry = 0;

}else if(jComboBox1.getSelectedIndex() == 1){

ratingLaundry = 2;

}else if(jComboBox1.getSelectedIndex() == 2){

ratingLaundry = 3;

}else if(jComboBox1.getSelectedIndex() == 3){

ratingLaundry = 4;

}else if(jComboBox1.getSelectedIndex() == 4){

ratingLaundry = 5;

}

UserHome.setTabelPenyedia(ratingLaundry,cariLaundry);

}

private void jTextField1InputMethodTextChanged(java.awt.event.InputMethodEvent evt) {

cariLaundry = jTextField1.getText();

System.out.println(cariLaundry);

UserHome.setTabelPenyedia(ratingLaundry,cariLaundry);

}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

cariLaundry = jTextField1.getText();

UserHome.setTabelPenyedia(ratingLaundry,cariLaundry);

}

private void jLabel33MouseClicked(java.awt.event.MouseEvent evt) {

UserHome.setTabelDeskripsi();

}

private void jTable2MouseClicked(java.awt.event.MouseEvent evt) {

}

private void button2ActionPerformed(java.awt.event.ActionEvent evt) {

UserHome.openDetailPenyedia(jTable2.getValueAt(jTable2.getSelectedRow(), 1).toString());

}

private void jLabel34MouseClicked(java.awt.event.MouseEvent evt) {

cariLaundry = jTextField1.getText();

UserHome.setTabelPenyedia(ratingLaundry,cariLaundry);

}

private void button5ActionPerformed(java.awt.event.ActionEvent evt) {

UserHome.openLaporanUser(jTable2.getValueAt(jTable2.getSelectedRow(), 0).toString());

}

private void button6ActionPerformed(java.awt.event.ActionEvent evt) {

if(button6.getLabel() == "Hapus"){

UserHome.deleteNotif(jTable2.getValueAt(jTable2.getSelectedRow(), 0).toString());

}else if(button6.getLabel() == "Bayar"){

UserHome.openPembayaran(jTable2.getValueAt(jTable2.getSelectedRow(), 0).toString());

}else if(button6.getLabel() == "Rating&Hapus"){

UserHome.openRatePenyedia(jTable2.getValueAt(jTable2.getSelectedRow(), 0).toString(), jTable2.getValueAt(jTable2.getSelectedRow(), 1).toString());

UserHome.deleteNotif(jTable2.getValueAt(jTable2.getSelectedRow(), 0).toString());

}

UserHome.setTabelDeskripsi();

}

private void button3ActionPerformed(java.awt.event.ActionEvent evt) {

UserHome.openEditProfile();

}

private void jLabel30MouseClicked(java.awt.event.MouseEvent evt) {

UserHome.openChangeToPenyedia();

}

private void setColor(JPanel pane, JPanel indicator, JPanel showing)

{

pane.setBackground(new Color(41,57,80));

indicator.setOpaque(true);

showing.setEnabled(true);

showing.setVisible(true);

showing.setOpaque(true);

JPanel[] panels = {btn\_1,btn\_2,btn\_3,btn\_4,btn\_5};

JPanel[] indicators = {ind\_1,ind\_2,ind\_3,ind\_4,ind\_5};

JPanel[] shows = {show\_1,show\_2,show\_3,show\_4,show\_5};

for(int i=0;i<panels.length;i++){

if(panels[i] != pane){

panels[i].setBackground(new Color(23,35,51));

}

}

for(int i=0;i<indicators.length;i++){

if(indicators[i] != indicator){

indicators[i].setOpaque(false);

}

}

for(int i=0;i<shows.length;i++){

if(shows[i] != showing){

shows[i].setEnabled(false);

shows[i].setVisible(false);

shows[i].setOpaque(false);

}

}

}

int potition = -1;

private void previousImage(){

try{

AnimationClass anim = new AnimationClass();

if(imgHome.getY() == 0 && potition == -1){

anim.jLabelYUp(imgHome.getY(), -200, 2, 1, imgHome);

potition = 1;

}else if(potition == 1){

anim.jLabelYDown(imgHome.getY(), 0, 2, 1, imgHome);

potition = -2;

}else if(potition == -2){

anim.jLabelYDown(imgHome.getY(), 200, 2, 1, imgHome);

potition = -1;

}else{

anim.jLabelYUp(imgHome.getY(), 0, 2, 1, imgHome);

}

anim.wait(10,1);

}catch(Exception e){}

}

public JLabel setUser(){

return jLabel32;

}

public JLabel setUsername(){

return jLabel25;

}

public JLabel setEmail(){

return jLabel26;

}

public JLabel setHandphone(){

return jLabel27;

}

public JLabel setAlamat(){

return jLabel24;

}

public JTable tabelPenyedia(){

return jTable1;

}

public JTable tabelDeskripsi(){

return jTable2;

}

1. HomeAdmin.java

package View;

import AppPackage.AnimationClass;

import java.awt.Color;

import java.awt.Dimension;

import java.awt.Toolkit;

import javax.swing.JPanel;

import javax.swing.table.DefaultTableModel;

import javax.swing.JLabel;

import Controller.AdminHome;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.JOptionPane;

import javax.swing.JTable;

public class HomeAdmin extends javax.swing.JFrame {

public static DefaultTableModel modeltab1, modeltab2;

int xx,xy;

private AdminHome AdminHome;

public static String username;

private Integer CBstatus = 0;

private String cari = "";

public HomeAdmin() {

initComponents();

setColor(btn\_1,ind\_1,show\_1);

ind\_1.setOpaque(true);

Dimension layar = Toolkit.getDefaultToolkit().getScreenSize();

this.setLocation(layar.width / 2 - this.getSize().width / 2, layar.height / 2 - this.getSize().height / 2);

this.AdminHome = new AdminHome(this);

}

private void btn\_1MousePressed(java.awt.event.MouseEvent evt) {

setColor(btn\_1,ind\_1,show\_1);

}

private void btn\_2MouseReleased(java.awt.event.MouseEvent evt) {

setColor(btn\_2,ind\_2,show\_2);

}

private void btn\_3MousePressed(java.awt.event.MouseEvent evt) {

setColor(btn\_3,ind\_3,show\_3);

}

private void jPanel2MousePressed(java.awt.event.MouseEvent evt) {

xx = evt.getX();

xy = evt.getY();

}

private void jPanel2MouseDragged(java.awt.event.MouseEvent evt) {

int x = evt.getXOnScreen();

int y = evt.getYOnScreen();

this.setLocation(x-xx,y-xy);

}

private void jLabel1MouseClicked(java.awt.event.MouseEvent evt) {

for (double i = 1.0; i >=0.0;i = i - 0.2)

{

String val = i+ "";

float f = Float.valueOf(val);

this.setOpacity(f);

try{

Thread.sleep(50);

}

catch(Exception e){}

}

System.exit(0);

}

private void formWindowOpened(java.awt.event.WindowEvent evt) {

for (double i = 0.0; i <=1.0;i = i + 0.2)

{

String val = i+ "";

float f = Float.valueOf(val);

this.setOpacity(f);

try{

Thread.sleep(50);

}

catch(Exception e){}

}

}

private void jLabel3MouseClicked(java.awt.event.MouseEvent evt) {

previousImage();

}

private void jPanel4MouseEntered(java.awt.event.MouseEvent evt) {

jPanel4.setBackground(new Color(71,120,197));

}

private void jPanel4MouseExited(java.awt.event.MouseEvent evt) {

jPanel4.setBackground(new Color(255,66,66));

}

private void jLabel1MouseEntered(java.awt.event.MouseEvent evt) {

jPanel4.setBackground(new Color(71,120,197));

}

private void jLabel1MouseExited(java.awt.event.MouseEvent evt) {

jPanel4.setBackground(new Color(255,66,66));

}

private void jLabel32MouseEntered(java.awt.event.MouseEvent evt) {

jLabel32.setText("Logout");

}

private void jLabel32MouseExited(java.awt.event.MouseEvent evt) {

jLabel32.setText(username);

}

private void jLabel32MouseClicked(java.awt.event.MouseEvent evt) {

new View.LoginSignUp().setVisible(true);

this.dispose();

}

private void formWindowActivated(java.awt.event.WindowEvent evt) {

Thread p = new Thread(){

public void run(){

for(;;){

try{

if(jTable1.getSelectedRow() > -1 && jTable1.getValueAt(jTable1.getSelectedRow(), 5) != "'Blocked'"){

button4.setVisible(true);

}else{

button4.setVisible(false);

}

if(jTable2.getSelectedRow() > -1){

button2.setVisible(true);

button3.setVisible(true);

button5.setVisible(true);

}else{

button2.setVisible(false);

button3.setVisible(false);

button5.setVisible(false);

}

sleep(100);

}catch(InterruptedException ex){

Logger.getLogger(View.HomeAdmin.class.getName()).log(Level.SEVERE, null, ex);

}

}

}

};

p.start();

jLabel32.setText(username);

}

private void jComboBox1ItemStateChanged(java.awt.event.ItemEvent evt) {

cari = jTextField1.getText();

switch (jComboBox1.getSelectedIndex()) {

case 0:

CBstatus = 0;

break;

case 1:

CBstatus = 1;

break;

case 2:

CBstatus = 2;

break;

case 3:

CBstatus = 3;

break;

default:

break;

}

AdminHome.setTabelPengguna(CBstatus, cari);

}

private void jTextField1InputMethodTextChanged(java.awt.event.InputMethodEvent evt) {

cari = jTextField1.getText();

}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

cari = jTextField1.getText();

AdminHome.setTabelPengguna(CBstatus, cari);

}

private void jLabel33MouseClicked(java.awt.event.MouseEvent evt) {

AdminHome.setTabelLaporan();

}

private void jTable2MouseClicked(java.awt.event.MouseEvent evt) {

}

private void jLabel34MouseClicked(java.awt.event.MouseEvent evt) {

cari = jTextField1.getText();

AdminHome.setTabelPengguna(CBstatus, cari);

}

private void button4ActionPerformed(java.awt.event.ActionEvent evt) {

AdminHome.setStatus(jTable1.getValueAt(jTable1.getSelectedRow(), 0).toString(), "-1");

AdminHome.setTabelPengguna(CBstatus, cari);

}

private void button5ActionPerformed(java.awt.event.ActionEvent evt) {

AdminHome.setStatus(jTable2.getValueAt(jTable2.getSelectedRow(), 1).toString(), "-1");

AdminHome.deleteLaporan(jTable2.getValueAt(jTable2.getSelectedRow(), 0).toString());

AdminHome.setTabelLaporan();

}

private void button2ActionPerformed(java.awt.event.ActionEvent evt) {

AdminHome.getDataUserViaJOptionPane(jTable2.getValueAt(jTable2.getSelectedRow(), 1).toString());

}

private void button3ActionPerformed(java.awt.event.ActionEvent evt) {

AdminHome.getDataUserViaJOptionPane(jTable2.getValueAt(jTable2.getSelectedRow(), 2).toString());

}

private void setColor(JPanel pane, JPanel indicator, JPanel showing)

{

pane.setBackground(new Color(41,57,80));

indicator.setOpaque(true);

showing.setEnabled(true);

showing.setVisible(true);

showing.setOpaque(true);

JPanel[] panels = {btn\_1,btn\_2,btn\_3};

JPanel[] indicators = {ind\_1,ind\_2,ind\_3};

JPanel[] shows = {show\_1,show\_2,show\_3};

for(int i=0;i<panels.length;i++){

if(panels[i] != pane){

panels[i].setBackground(new Color(23,35,51));

}

}

for(int i=0;i<indicators.length;i++){

if(indicators[i] != indicator){

indicators[i].setOpaque(false);

}

}

for(int i=0;i<shows.length;i++){

if(shows[i] != showing){

shows[i].setEnabled(false);

shows[i].setVisible(false);

shows[i].setOpaque(false);

}

}

}

int potition = -1;

private void previousImage(){

try{

AnimationClass anim = new AnimationClass();

if(imgHome.getY() == 0 && potition == -1){

anim.jLabelYUp(imgHome.getY(), -200, 2, 1, imgHome);

potition = 1;

}else if(potition == 1){

anim.jLabelYDown(imgHome.getY(), 0, 2, 1, imgHome);

potition = -2;

}else if(potition == -2){

anim.jLabelYDown(imgHome.getY(), 200, 2, 1, imgHome);

potition = -1;

}else{

anim.jLabelYUp(imgHome.getY(), 0, 2, 1, imgHome);

}

anim.wait(10,1);

}catch(Exception e){}

}

public JLabel setUser(){

return jLabel32;

}

public JTable tabelPengguna(){

return jTable1;

}

public JTable tabelLaporan(){

return jTable2;

}

1. HomePenyedia.java

package View;

import AppPackage.AnimationClass;

import java.awt.Color;

import java.awt.Dimension;

import java.awt.Toolkit;

import javax.swing.JPanel;

import javax.swing.table.DefaultTableModel;

import javax.swing.JLabel;

import Controller.PenyediaHome;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.JComboBox;

import javax.swing.JOptionPane;

import javax.swing.JTable;

import javax.swing.JTextField;

public class HomePenyedia extends javax.swing.JFrame {

public static DefaultTableModel modeltab1, modeltab2, modeltab3;

int xx,xy;

private PenyediaHome PenyediaHome;

public static String username;

private Integer ratingLaundry = 0;

public HomePenyedia() {

initComponents();

setColor(btn\_1,ind\_1,show\_1);

ind\_1.setOpaque(true);

Dimension layar = Toolkit.getDefaultToolkit().getScreenSize();

this.setLocation(layar.width / 2 - this.getSize().width / 2, layar.height / 2 - this.getSize().height / 2);

PenyediaHome = new PenyediaHome(this);

}

private void btn\_1MousePressed(java.awt.event.MouseEvent evt) {

setColor(btn\_1,ind\_1,show\_1);

}

private void btn\_2MouseReleased(java.awt.event.MouseEvent evt) {

setColor(btn\_2,ind\_2,show\_2);

}

private void btn\_3MousePressed(java.awt.event.MouseEvent evt) {

setColor(btn\_3,ind\_3,show\_3);

}

private void btn\_4MousePressed(java.awt.event.MouseEvent evt) {

setColor(btn\_4,ind\_4,show\_4);

}

private void btn\_5MouseReleased(java.awt.event.MouseEvent evt) {

setColor(btn\_5,ind\_5,show\_5);

}

private void jPanel2MousePressed(java.awt.event.MouseEvent evt) {

xx = evt.getX();

xy = evt.getY();

}

private void jPanel2MouseDragged(java.awt.event.MouseEvent evt) {

int x = evt.getXOnScreen();

int y = evt.getYOnScreen();

this.setLocation(x-xx,y-xy);

}

private void jLabel1MouseClicked(java.awt.event.MouseEvent evt) {

for (double i = 1.0; i >=0.0;i = i - 0.2)

{

String val = i+ "";

float f = Float.valueOf(val);

this.setOpacity(f);

try{

Thread.sleep(50);

}

catch(Exception e){}

}

System.exit(0);

}

private void formWindowOpened(java.awt.event.WindowEvent evt) {

for (double i = 0.0; i <=1.0;i = i + 0.2)

{

String val = i+ "";

float f = Float.valueOf(val);

this.setOpacity(f);

try{

Thread.sleep(50);

}

catch(Exception e){}

}

}

private void jLabel3MouseClicked(java.awt.event.MouseEvent evt) {

previousImage();

}

private void jPanel4MouseEntered(java.awt.event.MouseEvent evt) {

jPanel4.setBackground(new Color(71,120,197));

}

private void jPanel4MouseExited(java.awt.event.MouseEvent evt) {

jPanel4.setBackground(new Color(255,66,66));

}

private void jLabel1MouseEntered(java.awt.event.MouseEvent evt) {

jPanel4.setBackground(new Color(71,120,197));

}

private void jLabel1MouseExited(java.awt.event.MouseEvent evt) {

jPanel4.setBackground(new Color(255,66,66));

}

private void jLabel32MouseEntered(java.awt.event.MouseEvent evt) {

jLabel32.setText("Logout");

}

private void jLabel32MouseExited(java.awt.event.MouseEvent evt) {

jLabel32.setText(username);

}

private void jLabel32MouseClicked(java.awt.event.MouseEvent evt) {

new View.LoginSignUp().setVisible(true);

this.dispose();

}

private void jLabel29MouseClicked(java.awt.event.MouseEvent evt) {

new View.LoginSignUp().setVisible(true);

this.dispose();

}

private void formWindowActivated(java.awt.event.WindowEvent evt) {

Thread p = new Thread(){

public void run(){

for(;;){

try{

if(jTable3.getSelectedRow() > -1){

button4.setVisible(true);

}else{

button4.setVisible(false);

}

if(jTable2.getSelectedRow() > -1){

if(jTable2.getValueAt(jTable2.getSelectedRow(), 2).toString().equals("0")){

button2.setVisible(true);

button6.setVisible(true);

button6.setLabel("Telah Menjemput? Klik Disini");

button5.setVisible(false);

}else if(jTable2.getValueAt(jTable2.getSelectedRow(), 2).toString().equals("1")){

button5.setVisible(true);

button2.setVisible(true);

button6.setVisible(true);

button6.setLabel("Telah Dicuci? Klik Disini");

}else if(jTable2.getValueAt(jTable2.getSelectedRow(), 2).toString().equals("2")){

button5.setVisible(true);

button2.setVisible(true);

button6.setVisible(false);

}

}else{

button5.setVisible(false);

button2.setVisible(false);

button6.setVisible(false);

}

sleep(100);

}catch(InterruptedException ex){

Logger.getLogger(View.HomePenyedia.class.getName()).log(Level.SEVERE, null, ex);

}

}

}

};

p.start();

jLabel32.setText(username);

}

private void jLabel33MouseClicked(java.awt.event.MouseEvent evt) {

PenyediaHome.setTabelDeskripsi();

}

private void jTable2MouseClicked(java.awt.event.MouseEvent evt) {

}

private void button2ActionPerformed(java.awt.event.ActionEvent evt) {

PenyediaHome.getDataUserViaJOptionPane(jTable2.getValueAt(jTable2.getSelectedRow(), 1).toString());

}

private void jLabel34MouseClicked(java.awt.event.MouseEvent evt) {

PenyediaHome.setTabelLaporanPembayaran();

}

private void button6ActionPerformed(java.awt.event.ActionEvent evt) {

if(jTable2.getValueAt(jTable2.getSelectedRow(), 2).toString().equals("0")){

PenyediaHome.updateStatusLaundry(jTable2.getValueAt(jTable2.getSelectedRow(), 0).toString(), "1");

}else if(jTable2.getValueAt(jTable2.getSelectedRow(), 2).toString().equals("1")){

PenyediaHome.updateStatusLaundry(jTable2.getValueAt(jTable2.getSelectedRow(), 0).toString(), "2");

}

PenyediaHome.setTabelDeskripsi();

}

private void button3ActionPerformed(java.awt.event.ActionEvent evt) {

PenyediaHome.openEditProfile();

}

private void jLabel35MouseClicked(java.awt.event.MouseEvent evt) {

PenyediaHome.setTabelJenisLaundry();

}

private void button5ActionPerformed(java.awt.event.ActionEvent evt) {

PenyediaHome.openLaporanUser(jTable2.getValueAt(jTable2.getSelectedRow(), 0).toString());

}

private void button4ActionPerformed(java.awt.event.ActionEvent evt) {

PenyediaHome.deleteDataJenisLaundry(jTable3.getValueAt(jTable3.getSelectedRow(), 0).toString());

PenyediaHome.setTabelJenisLaundry();

}

private void button7ActionPerformed(java.awt.event.ActionEvent evt) {

PenyediaHome.addDataJenisLaundry();

PenyediaHome.setTabelJenisLaundry();

}

private void setColor(JPanel pane, JPanel indicator, JPanel showing)

{

pane.setBackground(new Color(41,57,80));

indicator.setOpaque(true);

showing.setEnabled(true);

showing.setVisible(true);

showing.setOpaque(true);

JPanel[] panels = {btn\_1,btn\_2,btn\_3,btn\_4,btn\_5};

JPanel[] indicators = {ind\_1,ind\_2,ind\_3,ind\_4,ind\_5};

JPanel[] shows = {show\_1,show\_2,show\_3,show\_4,show\_5};

for(int i=0;i<panels.length;i++){

if(panels[i] != pane){

panels[i].setBackground(new Color(23,35,51));

}

}

for(int i=0;i<indicators.length;i++){

if(indicators[i] != indicator){

indicators[i].setOpaque(false);

}

}

for(int i=0;i<shows.length;i++){

if(shows[i] != showing){

shows[i].setEnabled(false);

shows[i].setVisible(false);

shows[i].setOpaque(false);

}

}

}

int potition = -1;

private void previousImage(){

try{

AnimationClass anim = new AnimationClass();

if(imgHome.getY() == 0 && potition == -1){

anim.jLabelYUp(imgHome.getY(), -200, 2, 1, imgHome);

potition = 1;

}else if(potition == 1){

anim.jLabelYDown(imgHome.getY(), 0, 2, 1, imgHome);

potition = -2;

}else if(potition == -2){

anim.jLabelYDown(imgHome.getY(), 200, 2, 1, imgHome);

potition = -1;

}else{

anim.jLabelYUp(imgHome.getY(), 0, 2, 1, imgHome);

}

anim.wait(10,1);

}catch(Exception e){}

}

//Label

public JLabel setUser(){

return jLabel32;

}

public JLabel setUsername(){

return jLabel25;

}

public JLabel setEmail(){

return jLabel26;

}

public JLabel setHandphone(){

return jLabel27;

}

public JLabel setAlamat(){

return jLabel24;

}

public JLabel setTotal(){

return jLabel31;

}

//Tabel

public JTable tabelLaporanPembayaran(){

return jTable1;

}

public JTable tabelJenisLaundry(){

return jTable3;

}

public JTable tabelDeskripsi(){

return jTable2;

}

//Form Jenis Laundry

public JTextField nameBarang(){

return jTextField1;

}

public JComboBox nameHargaPer(){

return jComboBox1;

}

public JTextField nameHarga(){

return jTextField2;

}

1. LaporanUser.java

package View;

import java.awt.Dimension;

import java.awt.Toolkit;

import Controller.Laporan;

import javax.swing.JTextArea;

public class LaporanUser extends javax.swing.JFrame {

private Laporan Laporan;

public LaporanUser() {

initComponents();

Dimension layar = Toolkit.getDefaultToolkit().getScreenSize();

this.setLocation(layar.width / 2 - this.getSize().width / 2, layar.height / 2 - this.getSize().height / 2);

this.Laporan = new Laporan(this);

}

private void formWindowClosed(java.awt.event.WindowEvent evt) {

for (double i = 1.0; i >=1.0;i = i - 0.2)

{

String val = i+ "";

float f = Float.valueOf(val);

this.setOpacity(f);

try{

Thread.sleep(50);

}

catch(Exception e){}

}

new View.Home().setVisible(true);

}

private void formWindowOpened(java.awt.event.WindowEvent evt) {

}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

Laporan.sumbitForm();

}

public JTextArea deskripsi(){

return jTextArea1;

}

1. LoginSignUp.java

package View;

import AppPackage.AnimationClass;

import java.awt.Color;

import java.awt.Dimension;

import java.awt.Toolkit;

import Controller.Main;

import javax.swing.JOptionPane;

public class LoginSignUp extends javax.swing.JFrame {

int xx,xy;

private Main Main;

public LoginSignUp() {

initComponents();

Dimension layar = Toolkit.getDefaultToolkit().getScreenSize();

this.setLocation(layar.width / 2 - this.getSize().width / 2, layar.height / 2 - this.getSize().height / 2);

Main = new Main(this);

}

private void jPanel2MousePressed(java.awt.event.MouseEvent evt) {

xx = evt.getX();

xy = evt.getY();

}

private void jPanel2MouseDragged(java.awt.event.MouseEvent evt) {

int x = evt.getXOnScreen();

int y = evt.getYOnScreen();

this.setLocation(x-xx,y-xy);

}

private void jLabel1MouseClicked(java.awt.event.MouseEvent evt) {

for (double i = 1.0; i >=0.0;i = i - 0.2)

{

String val = i+ "";

float f = Float.valueOf(val);

this.setOpacity(f);

try{

Thread.sleep(50);

}

catch(Exception e){}

}

System.exit(0);

}

private void formWindowOpened(java.awt.event.WindowEvent evt) {

for (double i = 0.0; i <=1.0;i = i + 0.2)

{

String val = i+ "";

float f = Float.valueOf(val);

this.setOpacity(f);

try{

Thread.sleep(50);

}

catch(Exception e){}

}

}

private void jPanel4MouseEntered(java.awt.event.MouseEvent evt) {

jPanel4.setBackground(new Color(71,120,197));

}

private void jPanel4MouseExited(java.awt.event.MouseEvent evt) {

jPanel4.setBackground(new Color(255,66,66));

}

private void jLabel1MouseEntered(java.awt.event.MouseEvent evt) {

jPanel4.setBackground(new Color(71,120,197));

}

private void jLabel1MouseExited(java.awt.event.MouseEvent evt) {

jPanel4.setBackground(new Color(255,66,66));

}

private void jLabel14MouseClicked(java.awt.event.MouseEvent evt) {

AnimationClass anim = new AnimationClass();

anim.jLabelXLeft(jLabel10.getX(), jLabel10.getX()-380, 1, 1, jLabel10);

anim.jLabelXLeft(jLabel11.getX(), jLabel11.getX()-380, 1, 1, jLabel11);

anim.jLabelXLeft(jLabel12.getX(), jLabel12.getX()-380, 1, 1, jLabel12);

anim.jLabelXLeft(jLabel14.getX(), jLabel14.getX()-380, 1, 1, jLabel14);

anim.jButtonXLeft(jButton2.getX(), jButton2.getX()-380, 1, 1, jButton2);

anim.jTextFieldXLeft(jTextField4.getX(), jTextField4.getX()-380, 1, 1, jTextField4);

anim.jPasswordFieldXLeft(jPasswordField3.getX(), jPasswordField3.getX()-380, 1, 1, jPasswordField3);

anim.jLabelXLeft(jLabel2.getX(), 0, 1, 1, jLabel2);

anim.jLabelXLeft(jLabel3.getX(), jLabel3.getX()-380, 2, 1, jLabel3);

anim.jLabelXLeft(jLabel4.getX(), jLabel4.getX()-380, 2, 1, jLabel4);

anim.jLabelXLeft(jLabel5.getX(), jLabel5.getX()-380, 2, 1, jLabel5);

anim.jLabelXLeft(jLabel6.getX(), jLabel6.getX()-380, 2, 1, jLabel6);

anim.jLabelXLeft(jLabel7.getX(), jLabel7.getX()-380, 2, 1, jLabel7);

anim.jLabelXLeft(jLabel8.getX(), jLabel8.getX()-380, 2, 1, jLabel8);

anim.jLabelXLeft(jLabel9.getX(), jLabel9.getX()-380, 2, 1, jLabel9);

anim.jLabelXLeft(jLabel13.getX(), jLabel13.getX()-380, 2, 1, jLabel13);

anim.jTextFieldXLeft(jTextField1.getX(), jTextField1.getX()-380, 2, 1, jTextField1);

anim.jTextFieldXLeft(jTextField2.getX(), jTextField2.getX()-380, 2, 1, jTextField2);

anim.jTextFieldXLeft(jTextField3.getX(), jTextField3.getX()-380, 2, 1, jTextField3);

anim.jPasswordFieldXLeft(jPasswordField1.getX(), jPasswordField1.getX()-380, 2, 1, jPasswordField1);

anim.jPasswordFieldXLeft(jPasswordField2.getX(), jPasswordField2.getX()-380, 2, 1, jPasswordField2);

anim.jButtonXLeft(jButton1.getX(), jButton1.getX()-380, 2, 1, jButton1);

anim.jTextAreaXLeft(jScrollPane1.getX(), jScrollPane1.getX()-380, 2, 1, jScrollPane1);

}

private void jLabel13MouseClicked(java.awt.event.MouseEvent evt) {

AnimationClass anim = new AnimationClass();

anim.jLabelXRight(jLabel10.getX(), jLabel10.getX()+380, 2, 1, jLabel10);

anim.jLabelXRight(jLabel11.getX(), jLabel11.getX()+380, 2, 1, jLabel11);

anim.jLabelXRight(jLabel12.getX(), jLabel12.getX()+380, 2, 1, jLabel12);

anim.jLabelXRight(jLabel14.getX(), jLabel14.getX()+380, 2, 1, jLabel14);

anim.jButtonXRight(jButton2.getX(), jButton2.getX()+380, 2, 1, jButton2);

anim.jTextFieldXRight(jTextField4.getX(), jTextField4.getX()+380, 2, 1, jTextField4);

anim.jPasswordFieldXRight(jPasswordField3.getX(), jPasswordField3.getX()+380, 2, 1, jPasswordField3);

anim.jLabelXRight(jLabel2.getX(), jLabel2.getX()+380, 1, 1, jLabel2);

anim.jLabelXRight(jLabel3.getX(), jLabel3.getX()+380, 1, 1, jLabel3);

anim.jLabelXRight(jLabel4.getX(), jLabel4.getX()+380, 1, 1, jLabel4);

anim.jLabelXRight(jLabel5.getX(), jLabel5.getX()+380, 1, 1, jLabel5);

anim.jLabelXRight(jLabel6.getX(), jLabel6.getX()+380, 1, 1, jLabel6);

anim.jLabelXRight(jLabel7.getX(), jLabel7.getX()+380, 1, 1, jLabel7);

anim.jLabelXRight(jLabel8.getX(), jLabel8.getX()+380, 1, 1, jLabel8);

anim.jLabelXRight(jLabel9.getX(), jLabel9.getX()+380, 1, 1, jLabel9);

anim.jLabelXRight(jLabel13.getX(), jLabel13.getX()+380, 2, 1, jLabel13);

anim.jTextFieldXRight(jTextField1.getX(), jTextField1.getX()+380, 1, 1, jTextField1);

anim.jTextFieldXRight(jTextField2.getX(), jTextField2.getX()+380, 1, 1, jTextField2);

anim.jTextFieldXRight(jTextField3.getX(), jTextField3.getX()+380, 1, 1, jTextField3);

anim.jPasswordFieldXRight(jPasswordField1.getX(), jPasswordField1.getX()+380, 1, 1, jPasswordField1);

anim.jPasswordFieldXRight(jPasswordField2.getX(), jPasswordField2.getX()+380, 1, 1, jPasswordField2);

anim.jButtonXRight(jButton1.getX(), jButton1.getX()+380, 1, 1, jButton1);

anim.jTextAreaXLeft(jScrollPane1.getX(), jScrollPane1.getX()+380, 1, 1, jScrollPane1);

}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {

Main.Login(jTextField4.getText(), jPasswordField3.getText());

}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

try{

if(jPasswordField1.getText().equals(jPasswordField2.getText())){

Main.SignUp(

jTextField1.getText().toString()

, jTextField2.getText().toString()

, jTextField3.getText().toString()

, jPasswordField1.getText().toString()

, jTextArea1.getText().toString());

}else{

JOptionPane.showMessageDialog(null, "Password tidak sama");

}

}catch(Exception e){

JOptionPane.showMessageDialog(null, "Periksa Inputan Anda!");

}

}

1. Pembayaran.java

package View;

import java.awt.Dimension;

import java.awt.Toolkit;

import Controller.PembayaranJasa;

import javax.swing.JComboBox;

public class Pembayaran extends javax.swing.JFrame {

private PembayaranJasa PembayaranJasa;

public Pembayaran() {

initComponents();

Dimension layar = Toolkit.getDefaultToolkit().getScreenSize();

this.setLocation(layar.width / 2 - this.getSize().width / 2, layar.height / 2 - this.getSize().height / 2);

this.PembayaranJasa = new PembayaranJasa(this);

}

private void formWindowClosed(java.awt.event.WindowEvent evt) {

for (double i = 1.0; i >=1.0;i = i - 0.2)

{

String val = i+ "";

float f = Float.valueOf(val);

this.setOpacity(f);

try{

Thread.sleep(50);

}

catch(Exception e){}

}

new View.Home().setVisible(true);

}

private void formWindowOpened(java.awt.event.WindowEvent evt) {

}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

PembayaranJasa.cekTotalPembayaran();

}

public JComboBox metode\_pembayaran(){

return jComboBox1;

}

1. PesanJasaLaundry.java

package View;

import java.awt.Dimension;

import java.awt.Toolkit;

import Controller.PesanJasa;

import static java.lang.Thread.sleep;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.JComboBox;

import javax.swing.JLabel;

import javax.swing.JOptionPane;

import javax.swing.JTextArea;

import javax.swing.JTextField;

public class PesanJasaLaundry extends javax.swing.JFrame {

private PesanJasa PesanJasa;

public PesanJasaLaundry() {

initComponents();

Dimension layar = Toolkit.getDefaultToolkit().getScreenSize();

this.setLocation(layar.width / 2 - this.getSize().width / 2, layar.height / 2 - this.getSize().height / 2);

PesanJasa = new PesanJasa(this);

}

private void formWindowClosed(java.awt.event.WindowEvent evt) {

for (double i = 1.0; i >=1.0;i = i - 0.2)

{

String val = i+ "";

float f = Float.valueOf(val);

this.setOpacity(f);

try{

Thread.sleep(50);

}

catch(Exception e){}

}

new View.Home().setVisible(true);

}

private void formWindowOpened(java.awt.event.WindowEvent evt) {

Thread p = new Thread(){

public void run(){

for(;;){

try{

PesanJasa.selectedItem();

sleep(100);

}catch(InterruptedException ex){

Logger.getLogger(View.Home.class.getName()).log(Level.SEVERE, null, ex);

}

}

}

};

p.start();

}

private void jLabel7MouseClicked(java.awt.event.MouseEvent evt) {

JOptionPane.showMessageDialog(null, "Untuk 'Nama Pe-Laundry' dan 'No Handphone' bisa anda ubah di profile.");

}

private void jComboBox1ItemStateChanged(java.awt.event.ItemEvent evt) {

//

}

private void jComboBox1MouseClicked(java.awt.event.MouseEvent evt) {

//

}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

PesanJasa.sumbitForm(PesanJasa.modeltab1.getValueAt(jComboBox1.getSelectedIndex(), 2).toString(), jTextArea1.getText());

}

public JTextField textField1(){

return jTextField1;

}

public JComboBox comboBox1(){

return jComboBox1;

}

public JTextArea textArea1(){

return jTextArea1;

}

public JTextField textField2(){

return jTextField2;

}

public JLabel label6(){

return jLabel6;

}

1. RatingPenyedia.java

package View;

import java.awt.Dimension;

import java.awt.Toolkit;

import Controller.RatePenyedia;

import javax.swing.JComboBox;

public class RatingPenyedia extends javax.swing.JFrame {

private RatePenyedia RatePenyedia;

public RatingPenyedia() {

initComponents();

Dimension layar = Toolkit.getDefaultToolkit().getScreenSize();

this.setLocation(layar.width / 2 - this.getSize().width / 2, layar.height / 2 - this.getSize().height / 2);

this.RatePenyedia = new RatePenyedia(this);

}

private void formWindowClosed(java.awt.event.WindowEvent evt) {

for (double i = 1.0; i >=1.0;i = i - 0.2)

{

String val = i+ "";

float f = Float.valueOf(val);

this.setOpacity(f);

try{

Thread.sleep(50);

}

catch(Exception e){}

}

new View.Home().setVisible(true);

}

private void formWindowOpened(java.awt.event.WindowEvent evt) {

}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

RatePenyedia.sumbRating();

this.dispose();

}

public JComboBox rating(){

return jComboBox1;

}

1. TotalPembayaranOlehUser.java

package View;

import java.awt.Dimension;

import java.awt.Toolkit;

import Controller.TotalPembayaran;

import javax.swing.JTextField;

public class TotalPembayaranOlehUser extends javax.swing.JFrame {

private TotalPembayaran TotalPembayaran;

public TotalPembayaranOlehUser() {

initComponents();

Dimension layar = Toolkit.getDefaultToolkit().getScreenSize();

this.setLocation(layar.width / 2 - this.getSize().width / 2, layar.height / 2 - this.getSize().height / 2);

this.TotalPembayaran = new TotalPembayaran(this);

}

private void formWindowClosed(java.awt.event.WindowEvent evt) {

for (double i = 1.0; i >=1.0;i = i - 0.2)

{

String val = i+ "";

float f = Float.valueOf(val);

this.setOpacity(f);

try{

Thread.sleep(50);

}

catch(Exception e){}

}

new View.HomePenyedia().setVisible(true);

}

private void formWindowOpened(java.awt.event.WindowEvent evt) {

}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

TotalPembayaran.sumbitForm();

}

public JTextField nameTotal(){

return jTextField1;

}

1. UserProfile.java

package View;

import java.awt.Dimension;

import java.awt.Toolkit;

import Controller.ChangeToPenyedia;

import Controller.EditProfile;

import javax.swing.JButton;

import javax.swing.JLabel;

import javax.swing.JOptionPane;

import javax.swing.JTextArea;

import javax.swing.JTextField;

public class UserProfile extends javax.swing.JFrame {

private EditProfile EditProfile;

private ChangeToPenyedia ChangeToPenyedia;

public String mode = "";

public UserProfile() {

initComponents();

Dimension layar = Toolkit.getDefaultToolkit().getScreenSize();

this.setLocation(layar.width / 2 - this.getSize().width / 2, layar.height / 2 - this.getSize().height / 2);

if(Controller.PenyediaHome.menuTabbed == "modePenyedia"){

this.mode = "modePenyedia";

}else{

this.mode = "";

}

if(Controller.UserHome.menuTabbed == "changeToPenyedia"){

this.ChangeToPenyedia = new ChangeToPenyedia(this);

}else{

this.EditProfile = new EditProfile(this);

}

}

private void formWindowClosed(java.awt.event.WindowEvent evt) {

for (double i = 1.0; i >=1.0;i = i - 0.2)

{

String val = i+ "";

float f = Float.valueOf(val);

this.setOpacity(f);

try{

Thread.sleep(50);

}

catch(Exception e){}

}

JOptionPane.showMessageDialog(null, "Harap login kembali!");

mode = "";

new View.LoginSignUp().setVisible(true);

}

private void formWindowOpened(java.awt.event.WindowEvent evt) {

}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

if(Controller.UserHome.menuTabbed == "changeToPenyedia"){

ChangeToPenyedia.sumbAction(jTextField1.getText(), jTextField2.getText(), jTextField3.getText(), jTextArea1.getText(), jTextField4.getText(), jTextField5.getText());

}else{

EditProfile.sumbAction(jTextField1.getText(), jTextField2.getText(), jTextField3.getText(), jTextArea1.getText(), jTextField4.getText(), jTextField5.getText());

}

}

public JLabel judul(){

return jLabel1;

}

public JLabel usernameNamaPenyedia(){

return jLabel4;

}

public JButton action(){

return jButton1;

}

public JTextField username(){

return jTextField1;

}

public JTextField email(){

return jTextField2;

}

public JTextField handphone(){

return jTextField3;

}

public JTextArea alamat(){

return jTextArea1;

}

1. Controllers
2. AdminHome.java

package Controller;

import View.\*;

import Model.\*;

import java.sql.ResultSet;

import java.sql.SQLException;

import javax.swing.JOptionPane;

import javax.swing.table.DefaultTableModel;

import javax.swing.table.TableColumnModel;

public class AdminHome {

public static String idPenyedia = "";

public static String idLaundry = "";

private final HomeAdmin HomeAdmin;

private final User User;

private final Laundry Laundry;

private final LaporanUserM LaporanUserM;

public static String username, email, handphone, alamat;

public AdminHome(HomeAdmin HomeAdmin){

this.HomeAdmin = HomeAdmin;

this.User = new User();

this.Laundry = new Laundry();

this.LaporanUserM = new LaporanUserM();

setProfile();

setTabelPengguna(0, "");

setTabelLaporan();

}

public void setProfile(){

HomeAdmin.setUser().setText(username);

HomeAdmin.username = username;

}

public void setTabelPengguna(Integer CBstatus, String cari){

HomeAdmin.modeltab1 = new DefaultTableModel();

HomeAdmin.tabelPengguna().setModel(HomeAdmin.modeltab1);

HomeAdmin.modeltab1.addColumn("id\_user");

HomeAdmin.modeltab1.addColumn("Username");

HomeAdmin.modeltab1.addColumn("Email");

HomeAdmin.modeltab1.addColumn("Handphone");

HomeAdmin.modeltab1.addColumn("Alamat");

HomeAdmin.modeltab1.addColumn("Status");

HomeAdmin.modeltab1.setNumRows(0);

TableColumnModel tcm = HomeAdmin.tabelPengguna().getColumnModel();

//hide Id

tcm.getColumn(0).setWidth(0);

tcm.getColumn(0).setMinWidth(0);

tcm.getColumn(0).setMaxWidth(0);

try{

ResultSet home = User.getPengguna(cari);

while(home.next()){

String status = "";

if(home.getString("status").equals("-1")){

status = "'Blocked'";

}else if(home.getString("status").equals("0")){

status = "'Pengguna'";

}else if(home.getString("status").equals("1")){

status = "'Penyedia'";

}

if(CBstatus == 1 && home.getString("status").equals("0")){

HomeAdmin.modeltab1.addRow(new Object[]{

home.getString("id\_user"),home.getString("username"),home.getString("email"),home.getString("handphone"),home.getString("alamat"),status

});

}else if(CBstatus == 2 && home.getString("status").equals("1")){

HomeAdmin.modeltab1.addRow(new Object[]{

home.getString("id\_user"),home.getString("username"),home.getString("email"),home.getString("handphone"),home.getString("alamat"),status

});

}else if(CBstatus == 3 && home.getString("status").equals("-1")){

HomeAdmin.modeltab1.addRow(new Object[]{

home.getString("id\_user"),home.getString("username"),home.getString("email"),home.getString("handphone"),home.getString("alamat"),status

});

}else if(CBstatus == 0){

HomeAdmin.modeltab1.addRow(new Object[]{

home.getString("id\_user"),home.getString("username"),home.getString("email"),home.getString("handphone"),home.getString("alamat"),status

});

}

}

}catch(SQLException e){System.err.println(e.getMessage());}

}

public void setStatus(String id\_user, String status){

User.setStatus(id\_user, status);

}

public void setTabelLaporan(){

HomeAdmin.modeltab2 = new DefaultTableModel();

HomeAdmin.tabelLaporan().setModel(HomeAdmin.modeltab2);

HomeAdmin.modeltab2.addColumn("id\_laporanuser");

HomeAdmin.modeltab2.addColumn("id\_penyedia");

HomeAdmin.modeltab2.addColumn("id\_user");

HomeAdmin.modeltab2.addColumn("Deskripsi Laporan");

HomeAdmin.modeltab2.setNumRows(0);

TableColumnModel tcm = HomeAdmin.tabelLaporan().getColumnModel();

//hide Id

tcm.getColumn(0).setWidth(0);

tcm.getColumn(0).setMinWidth(0);

tcm.getColumn(0).setMaxWidth(0);

//

tcm.getColumn(1).setWidth(0);

tcm.getColumn(1).setMinWidth(0);

tcm.getColumn(1).setMaxWidth(0);

//

tcm.getColumn(2).setWidth(0);

tcm.getColumn(2).setMinWidth(0);

tcm.getColumn(2).setMaxWidth(0);

try{

ResultSet home = LaporanUserM.getLaporan();

while(home.next()){

HomeAdmin.modeltab2.addRow(new Object[]{

home.getInt(1), home.getInt(2), home.getInt(3), home.getString("deskripsi\_laporan")

});

}

}catch(SQLException e){System.err.println(e.getMessage());}

}

public void getDataUserViaJOptionPane(String id\_user){

if(Integer.parseInt(id\_user) < 1){

JOptionPane.showMessageDialog(null, "Tampaknya Pengguna telah mengkosongkan laporan ini");

}

try{

ResultSet home = User.detailPenyedia(id\_user);

while(home.next()){

if(home.getString("status") == "0"){

JOptionPane.showMessageDialog(null, "Nama Pengguna: "+home.getString("username")+

"\nEmail: "+home.getString("email")+

"\nHandphone: "+home.getString("handphone")+

"\nAlamat: "+home.getString("alamat")

);

}else{

JOptionPane.showMessageDialog(null, "Nama Laundry: "+home.getString("username")+

"\nEmail: "+home.getString("email")+

"\nHandphone: "+home.getString("handphone")+

"\nAlamat: "+home.getString("alamat")

);

}

}

}catch(SQLException e){System.err.println(e.getMessage());}

}

public void deleteLaporan(String id\_laporan){

LaporanUserM.fullDeleteNotif(id\_laporan);

}

}

1. ChangeToPenyedia.java

package Controller;

import View.\*;

import Model.\*;

import java.util.regex.Matcher;

import java.util.regex.Pattern;

import javax.swing.JOptionPane;

public class ChangeToPenyedia {

private View.UserProfile UserProfile;

private User User;

public ChangeToPenyedia(UserProfile UserProfile){

this.UserProfile = UserProfile;

this.User = new User();

setProfileLama();

}

public void sumbAction(String username, String email, String handphone, String alamat, String password, String confirmPassword){

Pattern emailP = Pattern.compile("^[A-Za-z0-9+\_.-]+@(.+)$");

Matcher emailM = emailP.matcher(email);

Pattern handphoneP = Pattern.compile("^[0-9]+$");

Matcher handphoneM = handphoneP.matcher(handphone);

if(username.isEmpty()){

JOptionPane.showMessageDialog(null, "Username Kosong");

}else if(email.isEmpty()){

JOptionPane.showMessageDialog(null, "Email Kosong");

}else if(!emailM.matches()){

JOptionPane.showMessageDialog(null, "Masukkan Email dengan benar");

}else if(handphone.isEmpty()){

JOptionPane.showMessageDialog(null, "No Handphone Kosong");

}else if(!handphoneM.matches()){

JOptionPane.showMessageDialog(null, "Masukkan No Handphone dengan benar");

}else if(alamat.isEmpty()){

JOptionPane.showMessageDialog(null, "Alamat Kosong");

}else if(!password.equals(confirmPassword)){

JOptionPane.showMessageDialog(null, "Password Tidak Sama");

}else{

User.cekLaundry(Controller.UserHome.email, Controller.UserHome.handphone);

if(User.statusLaundry == 0){

User.editProfile(Controller.UserHome.email, Controller.UserHome.handphone, username, email, handphone, password, alamat, "1");

UserProfile.dispose();

}else if(User.statusLaundry == 1){

JOptionPane.showMessageDialog(null, "Kosongkan Deskripsi untuk gabung menjadi Penyedia");

UserProfile.dispose();

}

}

}

public void setProfileLama(){

UserProfile.judul().setText("Lamar Sebagai Penyedia");

UserProfile.usernameNamaPenyedia().setText("Nama Penyedia");

UserProfile.action().setText("Lamar");

UserProfile.username().setText(Controller.UserHome.username);

UserProfile.email().setText(Controller.UserHome.email);

UserProfile.handphone().setText(Controller.UserHome.handphone);

UserProfile.alamat().setText(Controller.UserHome.alamat);

}

}

1. DetailPenyediaC.java

package Controller;

import View.\*;

import Model.\*;

import java.sql.ResultSet;

import javax.swing.JOptionPane;

public class DetailPenyediaC {

private DetailPenyedia DetailPenyedia;

private User User;

public static String idPenyedia = Controller.UserHome.idPenyedia;

public DetailPenyediaC(DetailPenyedia DetailPenyedia){

this.DetailPenyedia = DetailPenyedia;

this.User = new User();

getDetailPenyedia(idPenyedia);

}

public void getDetailPenyedia(String id\_penyedia){

try{

ResultSet rs = User.detailPenyedia(id\_penyedia);

if(rs.next()){

DetailPenyedia.username().setText(rs.getString("username").toString());

DetailPenyedia.handphone().setText(rs.getString("handphone").toString());

DetailPenyedia.email().setText(rs.getString("email").toString());

DetailPenyedia.alamat().setText(rs.getString("alamat").toString());

}

}catch(Exception e){

JOptionPane.showMessageDialog(null, "Maaf ada masalah dalam pengambilan Data Penyedia!");

}

}

}

1. EditProfile.java

package Controller;

import View.\*;

import Model.\*;

import java.util.regex.Matcher;

import java.util.regex.Pattern;

import javax.swing.JOptionPane;

public class EditProfile {

private View.UserProfile UserProfile;

private User User;

public EditProfile(UserProfile UserProfile){

this.UserProfile = UserProfile;

this.User = new User();

setProfileLama();

}

public void sumbAction(String username, String email, String handphone, String alamat, String password, String confirmPassword){

Pattern emailP = Pattern.compile("^[A-Za-z0-9+\_.-]+@(.+)$");

Matcher emailM = emailP.matcher(email);

Pattern handphoneP = Pattern.compile("^[0-9]+$");

Matcher handphoneM = handphoneP.matcher(handphone);

if(username.isEmpty()){

JOptionPane.showMessageDialog(null, "Username Kosong");

}else if(email.isEmpty()){

JOptionPane.showMessageDialog(null, "Email Kosong");

}else if(!emailM.matches()){

JOptionPane.showMessageDialog(null, "Masukkan Email dengan benar");

}else if(handphone.isEmpty()){

JOptionPane.showMessageDialog(null, "No Handphone Kosong");

}else if(!handphoneM.matches()){

JOptionPane.showMessageDialog(null, "Masukkan No Handphone dengan benar");

}else if(alamat.isEmpty()){

JOptionPane.showMessageDialog(null, "Alamat Kosong");

}else if(!password.equals(confirmPassword)){

JOptionPane.showMessageDialog(null, "Password Tidak Sama");

}else{

if(Controller.PenyediaHome.email == null && UserProfile.mode == ""){

User.editProfile(Controller.UserHome.email, Controller.UserHome.handphone, username, email, handphone, password, alamat, "0");

}else{

User.editProfile(Controller.PenyediaHome.email, Controller.PenyediaHome.handphone, username, email, handphone, password, alamat, "1");

}

UserProfile.dispose();

}

}

public void setProfileLama(){

UserProfile.judul().setText("Edit Profile");

UserProfile.usernameNamaPenyedia().setText("Username");

UserProfile.action().setText("Update");

if(UserProfile.mode == ""){

UserProfile.username().setText(Controller.UserHome.username);

UserProfile.email().setText(Controller.UserHome.email);

UserProfile.handphone().setText(Controller.UserHome.handphone);

UserProfile.alamat().setText(Controller.UserHome.alamat);

}else{

UserProfile.username().setText(Controller.PenyediaHome.username);

UserProfile.email().setText(Controller.PenyediaHome.email);

UserProfile.handphone().setText(Controller.PenyediaHome.handphone);

UserProfile.alamat().setText(Controller.PenyediaHome.alamat);

}

}

}

1. Laporan.java

package Controller;

import View.\*;

import Model.\*;

import javax.swing.JOptionPane;

public class Laporan {

private View.LaporanUser LaporanUser;

private LaporanUserM LaporanUserM;

public static String idLaundry = Controller.UserHome.idLaundry;

public Laporan(LaporanUser LaporanUser){

this.LaporanUser = LaporanUser;

this.LaporanUserM = new LaporanUserM();

}

public void sumbitForm(){

if(LaporanUser.deskripsi().getText().isEmpty()){

JOptionPane.showMessageDialog(null, "Masukkan deskripsi terlebih dahulu!");

}else{

LaporanUserM.reportPenyedia(idLaundry, LaporanUser.deskripsi().getText());

}

LaporanUser.dispose();

}

}

1. Main.java

package Controller;

import View.\*;

import Model.\*;

import java.util.regex.\*;

import javax.swing.JOptionPane;

public class Main {

private LoginSignUp LoginSignUp;

private Home Home;

private User User;

public static String emailUserLogin;

public Main(LoginSignUp LoginSignUp){

this.LoginSignUp = LoginSignUp;

this.User = new User();

}

public static void main(String[] args) {

try{

new View.LoginSignUp().setVisible(true);

}

catch(Exception e){}

}

public void SignUp(String username, String email, String handphone, String password, String alamat){

Pattern emailP = Pattern.compile("^[A-Za-z0-9+\_.-]+@(.+)$");

Matcher emailM = emailP.matcher(email);

Pattern handphoneP = Pattern.compile("^[0-9]+$");

Matcher handphoneM = handphoneP.matcher(handphone);

if(username.isEmpty()){

JOptionPane.showMessageDialog(null, "Username Kosong");

}else if(email.isEmpty()){

JOptionPane.showMessageDialog(null, "Email Kosong");

}else if(!emailM.matches()){

JOptionPane.showMessageDialog(null, "Masukkan Email dengan benar");

}else if(handphone.isEmpty()){

JOptionPane.showMessageDialog(null, "No Handphone Kosong");

}else if(!handphoneM.matches()){

JOptionPane.showMessageDialog(null, "Masukkan No Handphone dengan benar");

}else if(alamat.isEmpty()){

JOptionPane.showMessageDialog(null, "Alamat Kosong");

}else if(password.isEmpty()){

JOptionPane.showMessageDialog(null, "Password Kosong");

}else{

User.signUp(username, email, handphone, password, alamat);

if(User.loginStatus == 0){

new View.Home().setVisible(true);

LoginSignUp.dispose();

}else{

System.out.println("Gagal menampilkan halaman User karna ada masalah");

}

}

}

public void Login(String email, String password){

if(email.isEmpty()){

JOptionPane.showMessageDialog(null, "Email Kosong");

}else if(password.isEmpty()){

JOptionPane.showMessageDialog(null, "Password Kosong");

}else{

User.Login(email, password);

if(User.loginStatus == 2){

new View.HomeAdmin().setVisible(true);

LoginSignUp.dispose();

}else if(User.loginStatus == 1){

new View.HomePenyedia().setVisible(true);

LoginSignUp.dispose();

}else if(User.loginStatus == 0){

new View.Home().setVisible(true);

LoginSignUp.dispose();

}else if(User.loginStatus == -1){

JOptionPane.showMessageDialog(null, "Blocked");

LoginSignUp.dispose();

}

}

}

}

1. PembayaranJasa.java

package Controller;

import View.\*;

import Model.\*;

import java.sql.ResultSet;

import javax.swing.JOptionPane;

public class PembayaranJasa {

private Pembayaran Pembayaran;

private Transaksi Transaksi;

public static String idLaundry = Controller.UserHome.idLaundry;

public PembayaranJasa(Pembayaran Pembayaran){

this.Pembayaran = Pembayaran;

this.Transaksi = new Transaksi();

}

public void cekTotalPembayaran(){

try{

ResultSet rs = Transaksi.totalPembayaran(idLaundry);

if(rs.next()){

if(rs.getString("total\_pembayaran") == null){

JOptionPane.showMessageDialog(null, "Harap tunggu, tampaknya penyedia belum memasukkan harga.");

Pembayaran.dispose();

}else{

String metode = "";

if(Pembayaran.metode\_pembayaran().getSelectedIndex() == 0){

metode = "dana";

}else if(Pembayaran.metode\_pembayaran().getSelectedIndex() == 1){

metode = "gopay";

}else if(Pembayaran.metode\_pembayaran().getSelectedIndex() == 2){

metode = "ovo";

}

bayar(rs.getString("total\_pembayaran"), metode);

}

}

}catch(Exception e){

}

}

public void bayar(String total\_pembayaran, String metode\_pembayaran){

Transaksi.tambahkanPembayaran(idLaundry, metode\_pembayaran, total\_pembayaran);

Pembayaran.dispose();

}

}

1. PenyediaHome.java

package Controller;

import View.\*;

import Model.\*;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.logging.Level;

import java.util.logging.Logger;

import java.util.regex.Matcher;

import java.util.regex.Pattern;

import javax.swing.JOptionPane;

import javax.swing.table.DefaultTableModel;

import javax.swing.table.TableColumnModel;

public class PenyediaHome {

public static String menuTabbed = "";

public static String idLaundry = "";

private final HomePenyedia HomePenyedia;

private final User User;

private final Laundry Laundry;

private final JenisLaundry JenisLaundry;

private final Transaksi Transaksi;

public static String username, email, handphone, alamat;

public PenyediaHome(HomePenyedia HomePenyedia){

this.HomePenyedia = HomePenyedia;

this.User = new User();

this.Laundry = new Laundry();

this.JenisLaundry = new JenisLaundry();

this.Transaksi = new Transaksi();

setProfile();

setTabelLaporanPembayaran();

setTabelJenisLaundry();

setTabelDeskripsi();

}

public void setProfile(){

HomePenyedia.setUser().setText(username);

HomePenyedia.setUsername().setText(username);

HomePenyedia.setEmail().setText(email);

HomePenyedia.setHandphone().setText(handphone);

HomePenyedia.setAlamat().setText(alamat);

HomePenyedia.username = username;

}

public void setTabelLaporanPembayaran(){

HomePenyedia.modeltab1 = new DefaultTableModel();

HomePenyedia.tabelLaporanPembayaran().setModel(HomePenyedia.modeltab1);

HomePenyedia.modeltab1.addColumn("Metode Pembayaran");

HomePenyedia.modeltab1.addColumn("Pembayaran");

HomePenyedia.modeltab1.setNumRows(0);

try{

ResultSet home = Transaksi.getDataLaporanPembayaran();

ResultSet user = User.getIdUser(email, handphone);

Integer totalSemua = 0;

user.next();

while(home.next()){

if(home.getString("id\_penyedia").equals(user.getString("id\_user"))){

HomePenyedia.modeltab1.addRow(new Object[]{

home.getString("metode\_pembayaran"),"Rp."+home.getString("total\_pembayaran")

});

totalSemua += home.getInt("total\_pembayaran");

}

}

HomePenyedia.setTotal().setText("Total Pemasukan: Rp."+totalSemua.toString());

}catch(SQLException e){System.err.println(e.getMessage());}

}

public void setTabelJenisLaundry(){

HomePenyedia.modeltab3 = new DefaultTableModel();

HomePenyedia.tabelJenisLaundry().setModel(HomePenyedia.modeltab3);

HomePenyedia.modeltab3.addColumn("id\_jenislaundry");

HomePenyedia.modeltab3.addColumn("Nama Barang");

HomePenyedia.modeltab3.addColumn("Harga Per");

HomePenyedia.modeltab3.addColumn("Harga");

HomePenyedia.modeltab3.setNumRows(0);

TableColumnModel tcm = HomePenyedia.tabelJenisLaundry().getColumnModel();

//hide Id

tcm.getColumn(0).setWidth(0);

tcm.getColumn(0).setMinWidth(0);

tcm.getColumn(0).setMaxWidth(0);

try{

ResultSet home = JenisLaundry.getDataJenisLaundry();

ResultSet user = User.getIdUser(email, handphone);

user.next();

while(home.next()){

if(home.getString("id\_user").equals(user.getString("id\_user"))){

if(home.getString("harga\_peritem") == null){

HomePenyedia.modeltab3.addRow(new Object[]{

home.getString("id\_jenislaundry"),home.getString("barang"),"KiloGram","Rp."+home.getString("harga\_perkg")

});

}else{

HomePenyedia.modeltab3.addRow(new Object[]{

home.getString("id\_jenislaundry"),home.getString("barang"),"Item","Rp."+home.getString("harga\_peritem")

});

}

}

}

}catch(SQLException e){System.err.println(e.getMessage());}

}

public void deleteDataJenisLaundry(String id\_jenislaundry){

JenisLaundry.fullDeleteJenisLaundy(id\_jenislaundry);

}

public void addDataJenisLaundry(){

String barang = HomePenyedia.nameBarang().getText();

String harga = HomePenyedia.nameHarga().getText();

Pattern HargaP = Pattern.compile("^[0-9]+$");

Matcher hargaM = HargaP.matcher(harga);

if(barang == null){

JOptionPane.showMessageDialog(null, "Nama Barang tidak boleh kosong!");

}else if(harga == null){

JOptionPane.showMessageDialog(null, "Harga tidak boleh kosong!");

}else if(!hargaM.matches()){

JOptionPane.showMessageDialog(null, "Masukkan Harga dengan benar!");

}else{

try {

ResultSet user = User.getIdUser(email, handphone);

user.next();

if(HomePenyedia.nameHargaPer().getSelectedIndex() == 0){

JenisLaundry.addJenisLaundry(user.getString("id\_user"), barang, harga, "NULL");

}else{

JenisLaundry.addJenisLaundry(user.getString("id\_user"), barang, "NULL", harga);

}

}catch(SQLException e){System.err.println(e.getMessage());}

}

HomePenyedia.nameBarang().setText("");

HomePenyedia.nameHargaPer().setSelectedIndex(0);

HomePenyedia.nameHarga().setText("");

}

public void setTabelDeskripsi(){

HomePenyedia.modeltab2 = new DefaultTableModel();

HomePenyedia.tabelDeskripsi().setModel(HomePenyedia.modeltab2);

HomePenyedia.modeltab2.addColumn("id\_laundry");

HomePenyedia.modeltab2.addColumn("id\_user");

HomePenyedia.modeltab2.addColumn("status");

HomePenyedia.modeltab2.addColumn("Judul");

HomePenyedia.modeltab2.addColumn("Alamat");

HomePenyedia.modeltab2.setNumRows(0);

TableColumnModel tcm = HomePenyedia.tabelDeskripsi().getColumnModel();

//hide Id

tcm.getColumn(0).setWidth(0);

tcm.getColumn(0).setMinWidth(0);

tcm.getColumn(0).setMaxWidth(0);

//hide Id

tcm.getColumn(1).setWidth(0);

tcm.getColumn(1).setMinWidth(0);

tcm.getColumn(1).setMaxWidth(0);

//hide Status

tcm.getColumn(2).setWidth(0);

tcm.getColumn(2).setMinWidth(0);

tcm.getColumn(2).setMaxWidth(0);

try{

ResultSet user = User.getIdUser(email, handphone);

user.next();

ResultSet home = Laundry.getDataLaundry(user.getString("id\_user"));

while(home.next()){

String judulTabel = "";

String alamatTabel = "";

if(home.getString("status").equals("-1")){

judulTabel = "Terjadi Kesalahan!";

alamatTabel = "";

}else if(home.getString("status").equals("0")){

judulTabel = "Jemput Pakaian Pelangganmu Sekarang, Dia Sudah Menunggumu!";

alamatTabel = home.getString("alamat");

}else if(home.getString("status").equals("1")){

judulTabel = "Selesaikan Cucian Kamu, Pelangganmu Sudah Menunggu Lho!";

alamatTabel = home.getString("alamat");

}else if(home.getString("status").equals("2")){

judulTabel = "Masukkan Jumlah yang Harus Dibayarkan dan Tunggu Pelangganmu Membayar!";

alamatTabel = home.getString("alamat");

}

if(!home.getString("status").equals("3")){

HomePenyedia.modeltab2.addRow(new Object[]{

home.getString("id\_laundry"),home.getString("id\_user"),home.getString("status"),judulTabel,alamatTabel

});

}

}

}catch(SQLException e){System.err.println(e.getMessage());}

}

public void getDataUserViaJOptionPane(String id\_user){

try{

ResultSet home = User.detailPenyedia(id\_user);

while(home.next()){

JOptionPane.showMessageDialog(null, "Nama Pengguna: "+home.getString("username")+

"\nEmail: "+home.getString("email")+

"\nHandphone: "+home.getString("handphone")+

"\nAlamat: "+home.getString("alamat")

);

}

}catch(SQLException e){System.err.println(e.getMessage());}

}

public void updateStatusLaundry(String id\_laundy, String status){

try{

Laundry.setStatus(id\_laundy, status);

}catch(Exception e){}

}

public void openLaporanUser(String id\_laundry){

for (double i = 1.0; i >=0.0;i = i - 0.2)

{

String val = i+ "";

float f = Float.valueOf(val);

HomePenyedia.setOpacity(f);

try{

Thread.sleep(50);

}

catch(InterruptedException e){}

}

HomePenyedia.dispose();

idLaundry = id\_laundry;

new View.TotalPembayaranOlehUser().setVisible(true);

}

public void openEditProfile(){

for (double i = 1.0; i >=0.0;i = i - 0.2)

{

String val = i+ "";

float f = Float.valueOf(val);

HomePenyedia.setOpacity(f);

try{

Thread.sleep(50);

}

catch(InterruptedException e){}

}

menuTabbed = "modePenyedia";

HomePenyedia.dispose();

new View.UserProfile().setVisible(true);

menuTabbed = "";

}

}

1. PesanJasa.java

package Controller;

import View.\*;

import Model.\*;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.JOptionPane;

import javax.swing.table.DefaultTableModel;

public class PesanJasa {

public static String idPenyedia = Controller.UserHome.idPenyedia;

public static String namaPelaundry = Controller.UserHome.username;

public static String handphonePelaundry = Controller.UserHome.handphone;

public static String alamatPelaundry = Controller.UserHome.alamat;

public DefaultTableModel modeltab1;

private final PesanJasaLaundry PesanJasaLaundry;

private final User User;

private final JenisLaundry JenisLaundry;

private final Laundry Laundry;

public PesanJasa(PesanJasaLaundry PesanJasaLaundry){

this.PesanJasaLaundry = PesanJasaLaundry;

this.User = new User();

this.JenisLaundry = new JenisLaundry();

this.Laundry = new Laundry();

setDefaultForm();

}

public void setDefaultForm(){

PesanJasaLaundry.textField1().setText(namaPelaundry);

PesanJasaLaundry.textField1().setEditable(false);

PesanJasaLaundry.textField2().setText(handphonePelaundry);

PesanJasaLaundry.textField2().setEditable(false);

PesanJasaLaundry.textArea1().setText(alamatPelaundry);

PesanJasaLaundry.comboBox1().removeAllItems();

ResultSet rs = JenisLaundry.getJenisLaundry(idPenyedia);

modeltab1 = new DefaultTableModel();

modeltab1.addColumn("harga\_perkg");

modeltab1.addColumn("harga\_peritem");

modeltab1.addColumn("id\_jenislaundry");

modeltab1.setNumRows(0);

try {

while(rs.next()){

PesanJasaLaundry.comboBox1().addItem(rs.getString("barang"));

if(rs.getString("harga\_perkg") == null){

modeltab1.addRow(new Object[]{

null,rs.getString("harga\_peritem"),rs.getString("id\_jenislaundry")

});

}else{

modeltab1.addRow(new Object[]{

rs.getString("harga\_perkg"),null,rs.getString("id\_jenislaundry")

});

}

}

if(modeltab1.getValueAt(0, 1) == null){

PesanJasaLaundry.label6().setText("Rp."+modeltab1.getValueAt(0, 0).toString()+" per KG");

}else{

PesanJasaLaundry.label6().setText("Rp."+modeltab1.getValueAt(0, 1).toString()+" per Item");

}

} catch (SQLException ex) {

Logger.getLogger(PesanJasa.class.getName()).log(Level.SEVERE, null, ex);

}

}

public void selectedItem(){

if(modeltab1.getValueAt(PesanJasaLaundry.comboBox1().getSelectedIndex(), 1) == null){

PesanJasaLaundry.label6().setText("Rp."+modeltab1.getValueAt(PesanJasaLaundry.comboBox1().getSelectedIndex(), 0).toString()+" per KG");

}else{

PesanJasaLaundry.label6().setText("Rp."+modeltab1.getValueAt(PesanJasaLaundry.comboBox1().getSelectedIndex(), 1).toString()+" per Item");

}

}

public void sumbitForm(String id\_jenislaundry, String alamat){

Laundry.submit(PesanJasa.namaPelaundry, id\_jenislaundry, PesanJasa.handphonePelaundry, alamat, PesanJasa.idPenyedia);

PesanJasaLaundry.dispose();

}

}

1. RatePenyedia.java

package Controller;

import View.\*;

import Model.\*;

public class RatePenyedia {

public static String idPenyedia = Controller.UserHome.idPenyedia;

public static String namaPelaundry = Controller.UserHome.username;

public static String handphonePelaundry = Controller.UserHome.handphone;

private View.RatingPenyedia RatingPenyedia;

private Rating Rating;

private Laundry Laundry;

public RatePenyedia(RatingPenyedia RatingPenyedia){

this.RatingPenyedia = RatingPenyedia;

this.Rating = new Rating();

this.Laundry = new Laundry();

}

public void sumbRating(){

String bintang = "0";

if(RatingPenyedia.rating().getSelectedIndex() == 0){

bintang = "1";

}else if(RatingPenyedia.rating().getSelectedIndex() == 1){

bintang = "2";

}else if(RatingPenyedia.rating().getSelectedIndex() == 2){

bintang = "3";

}else if(RatingPenyedia.rating().getSelectedIndex() == 3){

bintang = "4";

}else if(RatingPenyedia.rating().getSelectedIndex() == 4){

bintang = "5";

}

Rating.ratingPenyedia(bintang, idPenyedia, namaPelaundry, handphonePelaundry);

RatingPenyedia.dispose();

}

}

1. TotalPembayaran.java

package Controller;

import View.\*;

import Model.\*;

import java.util.regex.Matcher;

import java.util.regex.Pattern;

import javax.swing.JOptionPane;

public class TotalPembayaran {

private View.TotalPembayaranOlehUser TotalPembayaranOlehUser;

private Laundry Laundry;

public static String idLaundry = Controller.PenyediaHome.idLaundry;

public TotalPembayaran(TotalPembayaranOlehUser TotalPembayaranOlehUser){

this.TotalPembayaranOlehUser = TotalPembayaranOlehUser;

this.Laundry = new Laundry();

}

public void sumbitForm(){

Pattern totalP = Pattern.compile("^[0-9]+$");

Matcher totalM = totalP.matcher(TotalPembayaranOlehUser.nameTotal().getText());

if(TotalPembayaranOlehUser.nameTotal().getText().isEmpty()){

JOptionPane.showMessageDialog(null, "Masukkan Total Pembayaran terlebih dahulu!");

}else if(!totalM.matches()){

JOptionPane.showMessageDialog(null, "Masukkan Total Harga dengan benar!");

}else{

Laundry.setTotalPembayaran(idLaundry, TotalPembayaranOlehUser.nameTotal().getText());

TotalPembayaranOlehUser.dispose();

}

}

}

1. UserHome.java

package Controller;

import View.\*;

import Model.\*;

import javax.swing.JOptionPane;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.ArrayList;

import javax.swing.table.DefaultTableModel;

import javax.swing.table.TableColumnModel;

public class UserHome {

public static String idPenyedia = "";

public static String idLaundry = "";

public static String menuTabbed = "";

private final Home Home;

private final User User;

private final JenisLaundry JenisLaundry;

private final Rating Rating;

private final Laundry Laundry;

private final LaporanUserM LaporanUserM;

public static String username, email, handphone, alamat;

public UserHome(Home Home){

this.Home = Home;

this.User = new User();

this.JenisLaundry = new JenisLaundry();

this.Rating = new Rating();

this.Laundry = new Laundry();

this.LaporanUserM = new LaporanUserM();

setProfile();

setTabelPenyedia(0,"");

setTabelDeskripsi();

}

public void setProfile(){

Home.setUser().setText(username);

Home.setUsername().setText(username);

Home.setEmail().setText(email);

Home.setHandphone().setText(handphone);

Home.setAlamat().setText(alamat);

Home.username = username;

}

public void setTabelPenyedia(Integer rating, String cari){

Home.modeltab1 = new DefaultTableModel();

Home.tabelPenyedia().setModel(Home.modeltab1);

Home.modeltab1.addColumn("id");

Home.modeltab1.addColumn("Nama Laundry");

Home.modeltab1.addColumn("Alamat");

Home.modeltab1.addColumn("Jenis Barang Yang Dilaundry");

Home.modeltab1.addColumn("Rating");

Home.modeltab1.setNumRows(0);

TableColumnModel tcm = Home.tabelPenyedia().getColumnModel();

//hide Id

tcm.getColumn(0).setWidth(0);

tcm.getColumn(0).setMinWidth(0);

tcm.getColumn(0).setMaxWidth(0);

//setRating

tcm.getColumn(4).setWidth(50);

tcm.getColumn(4).setMinWidth(50);

tcm.getColumn(4).setMaxWidth(50);

try{

ResultSet home = User.getUserPenyedia(email,cari);

while(home.next()){

ResultSet jl = JenisLaundry.getJenisLaundry(home.getString("id\_user"));

ArrayList<String> ary = new ArrayList<String>();

while(jl.next()){

ary.add(jl.getString("barang"));

}

ResultSet r = Rating.getAverageRatingPenyedia(home.getString("id\_user"));

r.next();

if(r.getFloat(1) >= rating){

Home.modeltab1.addRow(new Object[]{

home.getString("id\_user"),home.getString("username"),home.getString("alamat"),ary.toString(),r.getFloat(1)

});

}

}

}catch(SQLException e){System.err.println(e.getMessage());}

}

public void openPesanJasaForm(String id\_penyedia){

for (double i = 1.0; i >=0.0;i = i - 0.2)

{

String val = i+ "";

float f = Float.valueOf(val);

Home.setOpacity(f);

try{

Thread.sleep(50);

}

catch(InterruptedException e){}

}

Home.dispose();

idPenyedia = id\_penyedia;

new View.PesanJasaLaundry().setVisible(true);

}

public void setTabelDeskripsi(){

Home.modeltab2 = new DefaultTableModel();

Home.tabelDeskripsi().setModel(Home.modeltab2);

Home.modeltab2.addColumn("ID Laundry");

Home.modeltab2.addColumn("ID Penyedia");

Home.modeltab2.addColumn("Judul Notifikasi");

Home.modeltab2.addColumn("Deskripsi");

Home.modeltab2.addColumn("STATUS");

Home.modeltab2.setNumRows(0);

TableColumnModel tcm = Home.tabelDeskripsi().getColumnModel();

//hide Id

tcm.getColumn(0).setWidth(0);

tcm.getColumn(0).setMinWidth(0);

tcm.getColumn(0).setMaxWidth(0);

tcm.getColumn(1).setWidth(0);

tcm.getColumn(1).setMinWidth(0);

tcm.getColumn(1).setMaxWidth(0);

tcm.getColumn(4).setWidth(0);

tcm.getColumn(4).setMinWidth(0);

tcm.getColumn(4).setMaxWidth(0);

try{

ResultSet rs = Laundry.getLaundry(username, handphone);

while(rs.next()){

if(rs.getInt(4) == 0){

Home.modeltab2.addRow(new Object[]{

rs.getInt(1),rs.getInt(2),

"Karyawan sedang menjemput pakaian Anda",

"Harap tunggu karyawan di alamat yang telah Anda masukkan.",

rs.getInt(4)

});

}else if(rs.getInt(4) == 1){

Home.modeltab2.addRow(new Object[]{

rs.getInt(1),rs.getInt(2),

"Pakaian Anda sedang dilaundry",

"Harap tunggu laundry Anda.",

rs.getInt(4)

});

}else if(rs.getInt(4) == 2){

Home.modeltab2.addRow(new Object[]{

rs.getInt(1),rs.getInt(2),

"Karyawan sedang mengantarkan pakaian Anda",

"Harap tunggu karywan di alamat yang telah Anda masukkan. Siapkan pembayaran Anda sebelum karyawan datang.",

rs.getInt(4)

});

}else if(rs.getInt(4) == 3){

Home.modeltab2.addRow(new Object[]{

rs.getInt(1),rs.getInt(2),

"Rating penyedia??",

"Bantu penyedia menjadi top laundry didekat sini??",

rs.getInt(4)

});

}else if(rs.getInt(4) == -1){

Home.modeltab2.addRow(new Object[]{

rs.getInt(1),rs.getInt(2),

"Anda telah melaporkan "+rs.getString("username"),

"laporan Anda akan diproses secepatnya",

rs.getInt(4)

});

}

}

}catch(Exception e){System.out.println(e);}

}

public void openDetailPenyedia(String id\_penyedia){

for (double i = 1.0; i >=0.0;i = i - 0.2)

{

String val = i+ "";

float f = Float.valueOf(val);

Home.setOpacity(f);

try{

Thread.sleep(50);

}

catch(InterruptedException e){}

}

Home.dispose();

idPenyedia = id\_penyedia;

new View.DetailPenyedia().setVisible(true);

}

public void openLaporanUser(String id\_laundry){

for (double i = 1.0; i >=0.0;i = i - 0.2)

{

String val = i+ "";

float f = Float.valueOf(val);

Home.setOpacity(f);

try{

Thread.sleep(50);

}

catch(InterruptedException e){}

}

Home.dispose();

idLaundry = id\_laundry;

new View.LaporanUser().setVisible(true);

}

public void setStatusNotifikasi(String id\_laundry, String status){

Laundry.setStatus(id\_laundry, status);

}

public void deleteNotif(String id\_laundry){

LaporanUserM.deleteNotif(id\_laundry);

}

public void openRatePenyedia(String id\_laundry, String id\_penyedia){

for (double i = 1.0; i >=0.0;i = i - 0.2)

{

String val = i+ "";

float f = Float.valueOf(val);

Home.setOpacity(f);

try{

Thread.sleep(50);

}

catch(InterruptedException e){}

}

Home.dispose();

idLaundry = id\_laundry;

idPenyedia = id\_penyedia;

new View.RatingPenyedia().setVisible(true);

}

public void openPembayaran(String id\_laundry){

for (double i = 1.0; i >=0.0;i = i - 0.2)

{

String val = i+ "";

float f = Float.valueOf(val);

Home.setOpacity(f);

try{

Thread.sleep(50);

}

catch(InterruptedException e){}

}

Home.dispose();

idLaundry = id\_laundry;

new View.Pembayaran().setVisible(true);

}

public void openEditProfile(){

for (double i = 1.0; i >=0.0;i = i - 0.2)

{

String val = i+ "";

float f = Float.valueOf(val);

Home.setOpacity(f);

try{

Thread.sleep(50);

}

catch(InterruptedException e){}

}

Home.dispose();

menuTabbed = "editProfile";

new View.UserProfile().setVisible(true);

}

public void openChangeToPenyedia(){

for (double i = 1.0; i >=0.0;i = i - 0.2)

{

String val = i+ "";

float f = Float.valueOf(val);

Home.setOpacity(f);

try{

Thread.sleep(50);

}

catch(InterruptedException e){}

}

Home.dispose();

menuTabbed = "changeToPenyedia";

new View.UserProfile().setVisible(true);

}

}