**SKPL**-4003

SPESIFIKASI KEBUTUHAN PERANGKAT LUNAK

Yang Penting Diskon!

untuk:

Masyarakat Bandung dan sekitarnya

Kelompok 5

Dipersiapkan oleh:

Taufik Fathurahman (1301160790)

Brenda Irena (1301164641)

Husnul Khotimah Farid (1301162742)

Siti Fitria Yonalia (1301162744)

Yan Roberto (1301154715)

Program Studi S1 Teknik Informatika – Fakultas Informatika

Universitas Telkom

Jalan Telekomunikasi Terusan Buah Batu, Bandung

Indonesia

|  |  |  |  |
| --- | --- | --- | --- |
| Program Studi S1 Teknik Informatika  -  Fakultas Informatika | Nomor Dokumen | | Halaman |
| *SKPL-4003* | | 46 |
| Revisi |  | *Tgl:* |

# Daftar Perubahan

|  |  |
| --- | --- |
| Revisi | Deskripsi |
| A |  |
| B |  |
| C |  |
| D |  |
| E |  |
| F |  |
| G |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| INDEX | - | A | B | C | D | E | F | G |
| TGL |  |  |  |  |  |  |  |  |
| Ditulis oleh |  |  |  |  |  |  |  |  |
| Diperiksa oleh |  |  |  |  |  |  |  |  |
| Disetujui oleh |  |  |  |  |  |  |  |  |

# Daftar Halaman Perubahan

|  |  |  |  |
| --- | --- | --- | --- |
| Halaman | Revisi | Halaman | Revisi |
|  |  |  |  |

# Daftar Isi

[Daftar Perubahan 1](#_Toc527914662)

[Daftar Halaman Perubahan 2](#_Toc527914663)

[Daftar Isi 3](#_Toc527914664)

[1. Pendahuluan 6](#_Toc527914665)

[1.1 Tujuan Penulisan Dokumen 6](#_Toc527914666)

[1.2 Konvensi Dokumen 6](#_Toc527914667)

[1.3 Cakupan Produk 7](#_Toc527914668)

[1.4 Refernsi 8](#_Toc527914669)

[2. Overall Description 8](#_Toc527914670)

[2.1 Perspektif Produk 8](#_Toc527914671)

[2.2 Fungsi Produk 9](#_Toc527914672)

[2.3 Kelas dan Karakteristik Pengguna 10](#_Toc527914673)

[2.4 Lingkungan Operasi 11](#_Toc527914674)

[2.5 Batasan Perancangan dan Implementasi 11](#_Toc527914675)

[2.6 Dokumentasi Pengguna 12](#_Toc527914676)

[2.7 Asumsi dan Dependensi 12](#_Toc527914677)

[3. Requirements Antarmuka Eksternal 13](#_Toc527914678)

[3.1 Antarmuka Pengguna 13](#_Toc527914679)

[3.2 Antarmuka Perangkat Keras 16](#_Toc527914680)

[3.3 Antarmuka Perangkat Lunak 16](#_Toc527914681)

[3.4 Antarmuka Komunikasi 16](#_Toc527914682)

[4. Fitur Sistem (Use Cases) 18](#_Toc527914683)

[4.1 Use Case 1 18](#_Toc527914684)

[4.1.1 Nama Use Case: 18](#_Toc527914685)

[4.1.2 Tujuan: 19](#_Toc527914686)

[4.1.3 Input: 19](#_Toc527914687)

[4.1.4 Output: 19](#_Toc527914688)

[4.1.5 Skenario Utama: 19](#_Toc527914689)

[4.1.6 Prakondisi: 19](#_Toc527914690)

[4.1.7 Langkah-langkah: 19](#_Toc527914691)

[4.1.8 Pascakondisi: 19](#_Toc527914692)

[4.1.9 Skenario eksepsional 1: 20](#_Toc527914693)

[4.2 Use Case 2 20](#_Toc527914694)

[4.2.1 Nama Use Case: 20](#_Toc527914695)

[4.2.2 Tujuan: 20](#_Toc527914696)

[Input: 20](#_Toc527914697)

[4.2.3 Output: 20](#_Toc527914698)

[4.2.4 Skenario Utama: 20](#_Toc527914699)

[4.2.5 Prakondisi: 20](#_Toc527914700)

[4.2.6 Langkah-langkah: 21](#_Toc527914701)

[4.2.7 Pascakondisi: 21](#_Toc527914702)

[4.2.8 Skenario eksepsional 1: 21](#_Toc527914703)

[4.3 Use Case 3 21](#_Toc527914704)

[4.3.1 Nama Use Case: 21](#_Toc527914705)

[4.3.2 Tujuan: 21](#_Toc527914706)

[4.3.3 Input: 21](#_Toc527914707)

[4.3.4 Output: 22](#_Toc527914708)

[4.3.5 Skenario Utama: 22](#_Toc527914709)

[4.3.6 Prakondisi: 22](#_Toc527914710)

[4.3.7 Langkah-langkah: 22](#_Toc527914711)

[4.3.8 Pascakondisi: 22](#_Toc527914712)

[4.3.9 Skenario eksepsional 1: 23](#_Toc527914713)

[4.3.10 Skenario eksepsional 2: 23](#_Toc527914714)

[4.4 Use Case 4 23](#_Toc527914715)

[4.4.1 Nama Use Case: 23](#_Toc527914716)

[4.4.2 Tujuan: 23](#_Toc527914717)

[4.4.3 Input: 23](#_Toc527914718)

[4.4.4 Output: 23](#_Toc527914719)

[4.4.5 Skenario Utama: 24](#_Toc527914720)

[4.4.6 Prakondisi: 24](#_Toc527914721)

[4.4.7 Lankah-langkah: 24](#_Toc527914722)

[4.4.8 Pascakondisi: 24](#_Toc527914723)

[4.4.9 Skenario eksepsional 1: 25](#_Toc527914724)

[4.4.10 Skenario eksepsional 2: 25](#_Toc527914725)

[4.5 Use Case 5 25](#_Toc527914726)

[4.5.1 Nama Use Case: 25](#_Toc527914727)

[4.5.2 Tujuan: 25](#_Toc527914728)

[4.5.3 Input: 25](#_Toc527914729)

[4.5.4 Output: 25](#_Toc527914730)

[4.5.5 Skenario Utama: 25](#_Toc527914731)

[4.5.6 Prakondisi: 26](#_Toc527914732)

[4.5.7 Lankah-langkah: 26](#_Toc527914733)

[4.5.8 Pascakondisi: 26](#_Toc527914734)

[4.5.9 Skenario eksepsional 1: 26](#_Toc527914735)

[4.6 Use Case 6 26](#_Toc527914736)

[4.6.1 Nama Use Case: 26](#_Toc527914737)

[4.6.2 Tujuan: 27](#_Toc527914738)

[4.6.3 Input: 27](#_Toc527914739)

[4.6.4 Output: 27](#_Toc527914740)

[4.6.5 Skenario Utama: 27](#_Toc527914741)

[4.6.6 Prakondisi: 27](#_Toc527914742)

[4.6.7 Lankah-langkah: 28](#_Toc527914743)

[4.6.8 Pascakondisi: 28](#_Toc527914744)

[4.6.9 Skenario eksepsional 1: 28](#_Toc527914745)

[4.7 Use Case 7 29](#_Toc527914746)

[4.7.1 Nama Use Case: 29](#_Toc527914747)

[4.7.2 Tujuan: 29](#_Toc527914748)

[4.7.3 Input: 29](#_Toc527914749)

[4.7.4 Output: 29](#_Toc527914750)

[4.7.5 Skenario Utama: 29](#_Toc527914751)

[4.7.6 Prakondisi: 30](#_Toc527914752)

[4.7.7 Lankah-langkah: 30](#_Toc527914753)

[4.7.8 Pascakondisi: 30](#_Toc527914754)

[4.7.9 Skenario eksepsional 1: 30](#_Toc527914755)

[4.7.10 Skenario eksepsional 2: 31](#_Toc527914756)

[4.7.11 Skenario eksepsional 3: 31](#_Toc527914757)

[4.7.12 Skenario eksepsional 4: 31](#_Toc527914758)

[4.8 Use Case 8 31](#_Toc527914759)

[4.8.1 Nama Use Case: 31](#_Toc527914760)

[4.8.2 Tujuan: 31](#_Toc527914761)

[4.8.3 Input: 32](#_Toc527914762)

[4.8.4 Output: 32](#_Toc527914763)

[4.8.5 Skenario Utama: 32](#_Toc527914764)

[4.8.6 Prakondisi: 32](#_Toc527914765)

[4.8.7 Lankah-langkah: 32](#_Toc527914766)

[4.8.8 Pascakondisi: 32](#_Toc527914767)

[4.8.9 Skenario eksepsional 1: 33](#_Toc527914768)

[4.8.10 Skenario eksepsional 2: 33](#_Toc527914769)

[4.9 Use Case 9 33](#_Toc527914770)

[4.9.1 Nama Use Case: 33](#_Toc527914771)

[4.9.2 Tujuan: 33](#_Toc527914772)

[4.9.3 Input: 33](#_Toc527914773)

[4.9.4 Output: 33](#_Toc527914774)

[4.9.5 Skenario Utama: 33](#_Toc527914775)

[4.9.6 Prakondisi: 34](#_Toc527914776)

[4.9.7 Lankah-langkah: 34](#_Toc527914777)

[4.9.8 Pascakondisi: 35](#_Toc527914778)

[4.9.9 Skenario eksepsional 1: 35](#_Toc527914779)

[4.9.10 Skenario eksepsional 2: 35](#_Toc527914780)

[4.9.11 Skenario eksepsional 3: 35](#_Toc527914781)

[4.10 Use Case 10 35](#_Toc527914782)

[4.10.1 Nama Use Case: 35](#_Toc527914783)

[4.10.2 Tujuan: 35](#_Toc527914784)

[4.10.3 Input: 36](#_Toc527914785)

[4.10.4 Output: 36](#_Toc527914786)

[4.10.5 Skenario Utama: 36](#_Toc527914787)

[4.10.6 Prakondisi: 36](#_Toc527914788)

[4.10.7 Langkah-langkah: 37](#_Toc527914789)

[4.10.8 Pascakondisi: 37](#_Toc527914790)

[4.10.9 Skenario eksepsional 1: 37](#_Toc527914791)

[4.10.10 Skenario eksepsional 2: 37](#_Toc527914792)

[4.10.11 Skenario eksepsional 3: 38](#_Toc527914793)

[5. Requirements Nonfungsional Lainnya 38](#_Toc527914794)

[5.1 Requirements Performa 38](#_Toc527914795)

[5.2 Requirements Keselamatan 38](#_Toc527914796)

[5.3 Requirements Keamanan 39](#_Toc527914797)

[5.4 Atribut Kualitas Perangkat Lunak 39](#_Toc527914798)

[6. Requirements Lain 40](#_Toc527914799)

# Pendahuluan

## Tujuan Penulisan Dokumen

Dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL) merupakan dokumen acuan atau panduan baik bagi pengembang maupun pengguna perangkat lunak selama dalam pengembangan perangkat lunak. Dalam dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL) ini bertujuan untuk perancangan, memberikan penjelasan aplikasi yang akan dikembangkan, dan sebagai acuan teknis pengembangan perangkat lunak Aplikasi Yang Penting Diskon (AYEDIS) pada tahapan selanjutnya.

Bagi pihak pengembang aplikasi, dokumen SKPL ini akan digunakan sebagai acuan dalam setiap tahapan pengembangan agar sesuai dengan kebutuhan yang diinginkan oleh pengguna dan tujuan awal dari diciptakannya perangkat lunak ini. Sedangkan bagi pihak pengguna, SKPL ini akan digunakan untuk mendata semua spesifikasi kebutuhan dari perangkat lunak yang akan dikembangkan.

## Konvensi Dokumen

Konvensi dokumen dalam Spesifikasi Kebutuhan Perangkat Lunak (SKPL) ini ditulis dengan jenis huruf Times New Roman dengan ukuran huruf 12. Untuk mempermudah dan memperjelas requirement yang akan menjadi perhatian utama pada perangkat lunak akan **diberi cetak tebal**, dan yang tidak dicetak tebal adalah requirement turunannya. Diantaranya adalah sebagai berikut :

1. **User Costumer :** mendapatkan informasi yang paling akurat dan terbaru.
2. **Admin :** dapat mengorganisir setiap akun yang menggunakan aplikasi.
3. **User Seller :** dapat memberikan informasi toko dan produk yang dimiliki dengan informasi yang akurat dan terbaru.
4. Admin : dapat menambahkan, memperbaharui, dan menghapus kategori produk pada aplikasi.
5. Admin : dapat menambahkan, memperbaharui, dan menghapus produk pada aplikasi.
6. User Seller : dapat menambahkan, memperbaharui, dan menghapus produk pada aplikasi.
7. User Customer : dapat melihat kategori barang, list barang, dan memilih barang.

## Cakupan Produk

Perangkat lunak yang akan dibuat dalam SKPL ini adalah suatu produk perangkat lunak yang dapat memfasilitasi masyarakat agar dapat lebih mudah mencari informasi mengenai diskon yang terbaru dan terakurat sekaligus memberikan informasi mengenai toko penyedia barang secara menyeluruh. Pada aplikasi ini, selain menyediakan informasi, juga dapat sekaligus menjadi peta untuk menuju toko penyedia produk dengan menggunakan API google map. Aplikasi ini dapat digunakan kapan saja dan dimana saja melalui koneksi internet sehingga memudahkan penggunanya.

Dokumen SKPL ini akan menyasar beberapa jenis pembaca, antara lain :

1. Developer

Developer merupakan orang-orang yang terlibat secara langsung dalam pembuatan software ini, seperti programmer dan system analyst. Setiap cakupan yang berada pada SKPL ini diharapkan akan membantu developer dalam menjalankan setiap langkah-langkah teknisnya.

1. Project Manager

Merupakan orang-orang yang menjadi pemimpin dalam proyek yang sedang dikembangkan secara langsung. Diharapkan Project Manager dapat memahami konsep dari perangkat lunak dan ide utama dari pembuatan perangkat lunak yang ada pada dokumen SKPL ini.

1. Software Tester

Merupakan orang-orang yang ditunjuk secara acak untuk menguji perangkat lunak yang sedang dikembangkan. Pengujian akan berfokus pada performa dari aplikasi, diharapkan setelah proses pengujian akan meminimalisir kesalahan-kesalahan dalam perangkat lunak dan meningkatkan performa dari perangkat lunak.

1. User Customer

Merupakan orang-orang yang akan menggunakan perangkat lunak sebagai konsumen. Dalam dokumen ini diharapkan user customer dapat memahami spesifikasi dan fungsionalitas setiap fitur yang disediakan dalam perangkat lunak.

1. User Seller

Merupakan orang-orang atau lembaga yang akan menggunakan perangkat lunak sebagai produsen atau orang yang akan membuka toko di dalam perangkat lunak. Dalam dokumen SKPL ini diharapkan user seller dapat memahami spesifikasi dan fungsionalitas dari setiap fitur yang disediakan di dalam perangkat lunak.

## Refernsi

Dokumen-dokumen dan sumber-sumber yang digunakan sebagai referensi pada pembuatan SKPL ini adalah sebagai berikut :

1. Template SKPL Fakultas Informatika, Telkom University.
2. IEEE Std 830-1993, IEEE Recommended Parctice for Software Requirement Specifications.
3. Ahmad Permana. “Spesifikasi Kebutuhan Perangkat Lunak untuk Smart Hotel Pro”. 2007.

<https://komputasi.files.wordpress.com/2017/03/contoh-skpl-smart-hotel-pro.doc>.

# Overall Description

## Perspektif Produk

Proyek yang dikembangkan dalam dokumen ini akan menghasilkan sebuah produk yang dapat membantu proses pemberian informasi dari produsen kepada konsumen. Informasi pada perangkat lunak ini akan berfokus pada produk-produk yang sedang pada masa diskon untuk wilayah Bandung dan sekitarnya. Perangkat lunak yang akan dikembangkan akan diberi nama AYEDIS (Aplikasi Yang Penting Diskon).

AYEDIS akan mampu menolong para pencari barang diskon yang berada di wilayah Bandung dan sekitarnya dengan mudah, praktis, akurat, dan ter-update. Dan bagi para produsen yang ingin memasarkan, atau mengiklankan produk-produknya yang sedang dalam masa diskon dapat menjadikan perangkat lunak ini sebagai wadah promosi dan pemasaran. Para produsen atau akan kami sebut sebagai user seller pada perangkat lunak ini dapat melakukan pembuatan toko, penambahan, pembaharuan, dan penghapusan produk setiap saat dengan mudah dan cepat, sehingga keakuratan dan keterbaharuan dari informasi yang ada pada perangkat lunak selalu terjaga dengan baik.

Bagi pihak yang menjadi administrator pada perangkat lunak ini, diharapkan perangkat lunak ini akan memberikan kemudahan dalam mengorganisir setiap kegiatan yang dilakukan pada perangkat lunak.

Seller Interface

Admin Interface

Customer Interface

1. Menu Kategori Barang
2. Menu Kelola Produk
3. Menu Kelola Produk
4. Menu Lihat barang
5. Menu Kelola Toko
6. Menu Kelola Toko
7. Menu Kelola Kategori
8. Menu Kelola Akun

Gambar 1 Hubungan antar subsistem Aplikasi Yang Penting Diskon

## Fungsi Produk

Masalah yang sering dihadapi saat mencari atau mendapati informasi produk diskon sebelum ada perangkat lunak ini antara lain adalah :

1. Tanggal atau waktu dari produk yang mengalami diskon telah habis tanpa disadari oleh konsumen.
2. Stok dari barang diskon tiba-tiba habis ketika akan membeli produk.
3. Ketika mendapati informasi produk diskon namun mengalami kebingungan karena tidak mengetahui lokasi toko penyedia barang diskon.

Dengan hadirnya perangkat lunak AYEDIS (Aplikasi Yang Penting Diskon) akan memberi dampak sebagai berikut ini :

1. Dapat menampilkan menu pilihan kategori produk yang sedang diskon.
2. Dapat menampilkan *list* barang yang sedang diskon di aplikasi.
3. Dapat menampilkan informasi detail barang yang sedang diskon di aplikasi.
4. Dapat menampilkan informasi toko yang sedang diskon di aplikasi.
5. Menunjukkan dan mengarahkan ke tempat lokasi diskon dengan menggunakan API *Google Map.*
6. Ketika *user customer* memilih barang pada keranjang, maka ia telah memesan barang tersebut dan tidak akan jatuh ke tangan orang lain, namun dengan jangka waktu tertentu.

## Kelas dan Karakteristik Pengguna

Dalam perangkat lunak AYEDIS (Aplikasi Yang Penting Diskon) akan terdapat beberapa kelas pengguna, yang akan dibedakan atas beberapa faktor. Diantaranya akan dibedakan menjadi administrator atau admin, dan *user*. Untuk *user* akan dipecah menjadi dua jenis yaitu *user customer* dan *user seller.* Setiap kelas pengguna aplikasi akan memiliki hak akses yang berbeda-beda saat menggunakan layanan pada aplikasi AYEDIS.

1. Administrator (Admin)

Pengguna yang pertama adalah admin, dimana admin akan memiliki hak otorisasi lebih dibandingkan *user customer* dan *user seller*. Admin dapat melakukan perubahan isi pada perangkat lunak seperti *add/update/delete* pada toko, produk, dan kategori yang ada pada perangkat lunak. Admin juga memiliki hak untuk mengorganisir atau mengelola setiap akun yang terdaftar pada aplikasi, sehingga admin dapat menghapus atau melakukan *banned* pada akun yang bermasalah karena melakukan pelanggaran.

1. User Seller

Merupakan pengguna aplikasi yang akan berperan sebagai produsen atau pembuka toko pada aplikasi. *User seller* akan memiliki hak akses untuk melakukan *add/update/delete* pada toko dan produk yang dimiliki.

1. User Customer

Merupakan pengguna aplikasi yang akan berperan sebagai konsumen, dimana akan mendapatkan hak akses untuk memilih kategori produk, melihat *list* barang yang disediakan oleh aplikasi, memilih barang untuk dimasukkan kedalam keranjang, dan mendapatkan informasi mendetail tentang produk yang dipilih, dan toko yang menyediakan produk.

## Lingkungan Operasi

Berikut ini merupakan lingkup operasi dari aplikasi AYEDIS (Aplikasi Yang Penting Diskon) :

1. Hadware :

* Processor QuadCore, 1.2 GHz
* RAM 512 MB
* Internal storage free space 100 MB
* Smartphone dengan layar sentuh

1. Software :

* Sistem operasi minimum Jelly Bean
* DBMS Oracle 11g
* Bahasa pemrograman Java
* Sistem operasi windows server 2003

## Batasan Perancangan dan Implementasi

Batasan-batasan khusus yang akan ada pada aplikasi ini antara lain adalah sebagai berikut ini:

* *Database* yang akan digunakan adalah *Oracle.*
* Aplikasi hanya dapat dijalan pada *platform* *android* dengan sistem operasi minimum adalah *Jelly Bean.*
* Bahasa pemrograman yang akan digunakan adalah Java.
* Desain antarmuka harus menarik dan mudah dipahami oleh *user customer.*
* Aplikasi hanya dapat dinikmati atau digunakan apabila telah memiliki akun atau telah terdaftar pada aplikasi.
* *Device* *android* yang digunakan harus mendukung penggunaan layar sentuh.

## Dokumentasi Pengguna

Pada aplikasi AYEDIS (Aplikasi Yang Penting Diskon) akan memanfaatkan salah satu jenis *help* yaitu *on-line help* yang akan ada pada salah satu menu pada aplikasi AYEDIS. Dokumentasi pengguna ini akan berisikan panduan dasar menggunakan aplikasi AYEDIS, dan beberapa permasalahan umum yang sering terjadi serta bagaimana cara mengatasinya. Informasi yang ada akan terus diperbaharui berdasarkan pada perkembangan dan masukan dari pengguna aplikasi.

## Asumsi dan Dependensi

Aplikasi AYEDIS (Aplikasi Yang Penting Diskon) akan berjalan dengan baik apabila didukung oleh *software* dan *hardware* yang memenuhi syarat minimum, sepeti diantaranya :

1. Hadware :

* Processor QuadCore, 1.2 GHz
* RAM 512 MB
* Internal storage free space 100 MB
* Smartphone dengan layar sentuh

1. Software :

* Sistem operasi minimum Jelly Bean
* DBMS Oracle 11g
* Bahasa pemrograman Java
* Sistem operasi windows server 2003

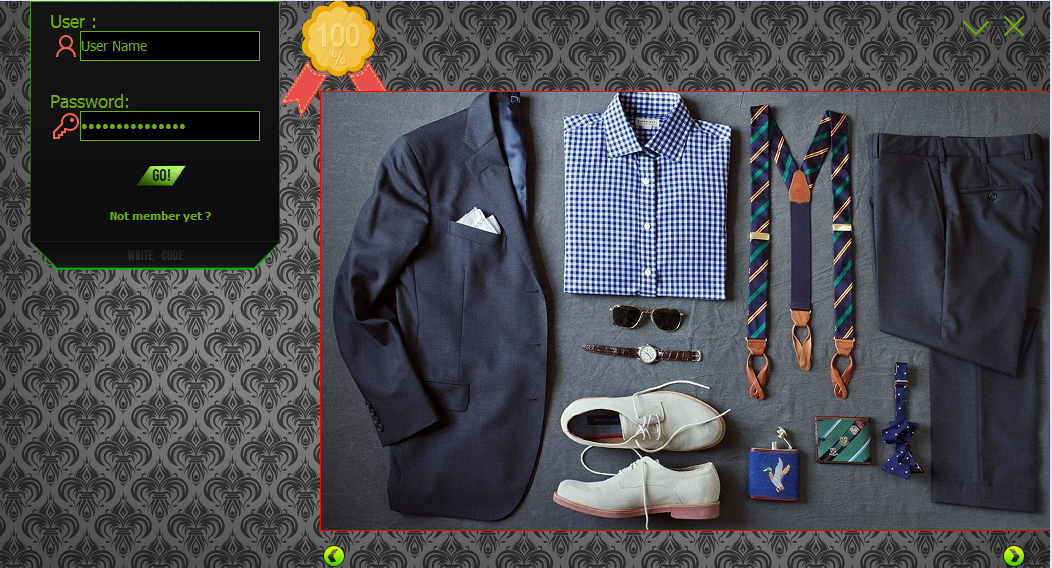
Selain faktor teknis seperti kebutuhan hardware dan software, faktor yang tak kalah penting adalah desain antarmuka yang interaktif, menarik, dan juga mudah dipahami juga akan sangat menentukan. Desain antarmuka akan dibuat lebih bernuansa modern dan anak muda, karena sektor atau konsumen utama yang akan disasar adalah anak muda atau anak kuliahan.

# Requirements Antarmuka Eksternal

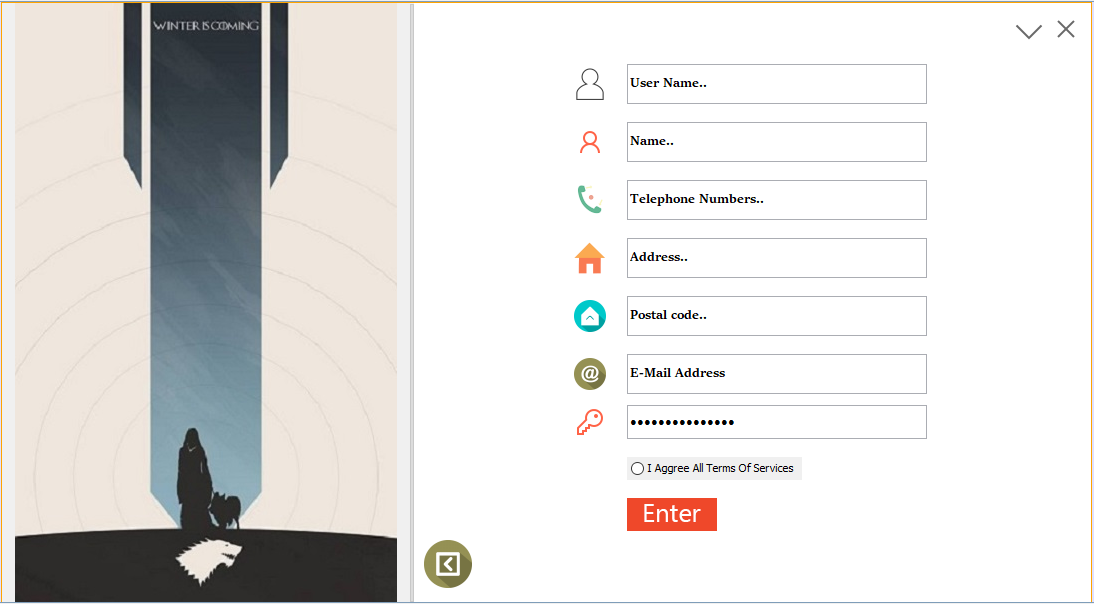
## Antarmuka Pengguna

Aplikasi AYEDIS (Aplikasi Yang Penting Diskon) merupakan aplikasi berbasis *android* yang memiliki tampilan antarmuka seperti pada umumnya aplikasi jual beli *online* JD.ID, Tokopedia, dan lainnya. Ketika pertama kali membuka aplikasi, maka akan dihadapkan pada halaman untuk melakukan *login* pada aplikasi. Kemudian ketika proses *login* dilakukan, maka sistem akan membawa ke halaman utama aplikasi sesuai dengan jenis *user*. Setiap jenis *user* akan memiliki fitur yang berbeda-beda yang diberikan oleh aplikasi. Pengguna aplikasi melakukan interaksi dengan aplikasi atau perangkat lunak melalui antarmuka yang telah diberikan dan apabila pengguna akan melakukan *input* dapat menggunkan perintah klik atau *touch* pada layar.

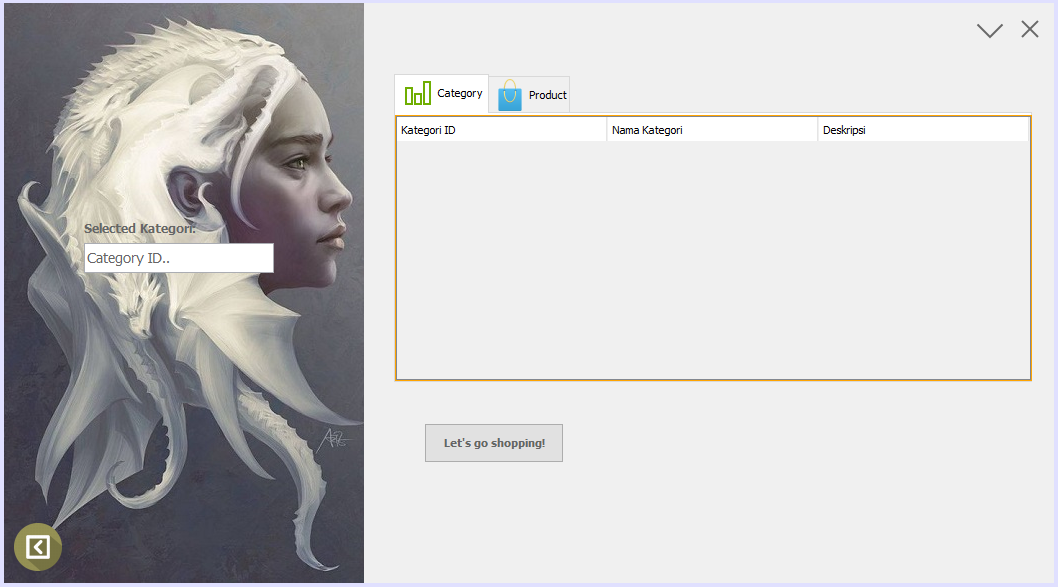
Tampilan antarmuka dari aplikasi AYEDIS dirancang seinteraktif, menarik, dan semudah mungkin untuk pengguna. Berikut ini adalah contoh dari beberapa halaman yang akan ada pada aplikasi :

1. Halaman Login

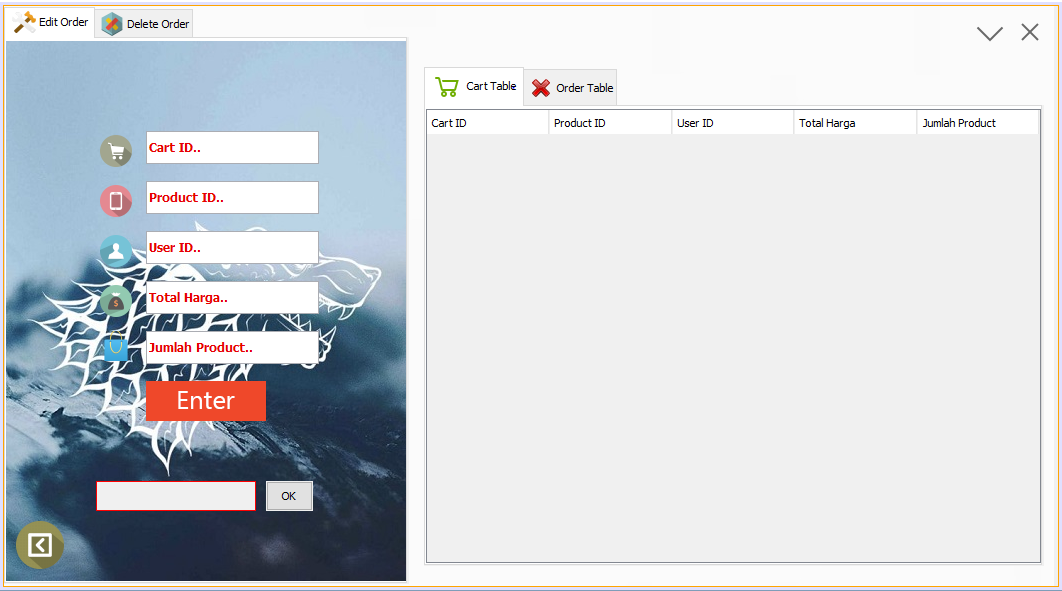
*Gambar 2 antarmuka pengguna*

1. Halaman Registrasi

*Gambar 3 antarmuka pengguna*

1. Halaman memilih kategori dan barang

*Gambar 4 antarmuka pengguna*

1. Halaman kelola keranjang untuk *customer*

*Gambar 5 antarmuka pengguna*

## Antarmuka Perangkat Keras

Berikut ini merupakan kebutuhan antarmuka perangkat keras yang akan digunakan:

* Processor QuadCore, 1.2 GHz
* RAM 512 MB
* Internal storage free space 100 MB
* Smartphone dengan layar sentuh

## Antarmuka Perangkat Lunak

Berikut ini merupakan kebutuhan perangkat lunak yang harus dipenuhi untuk aplikasi AYEDIS antara lain:

* Sistem operasi minimum Jelly Bean
* DBMS Oracle 11g
* Bahasa pemrograman Java
* Sistem operasi windows server 2003

## Antarmuka Komunikasi

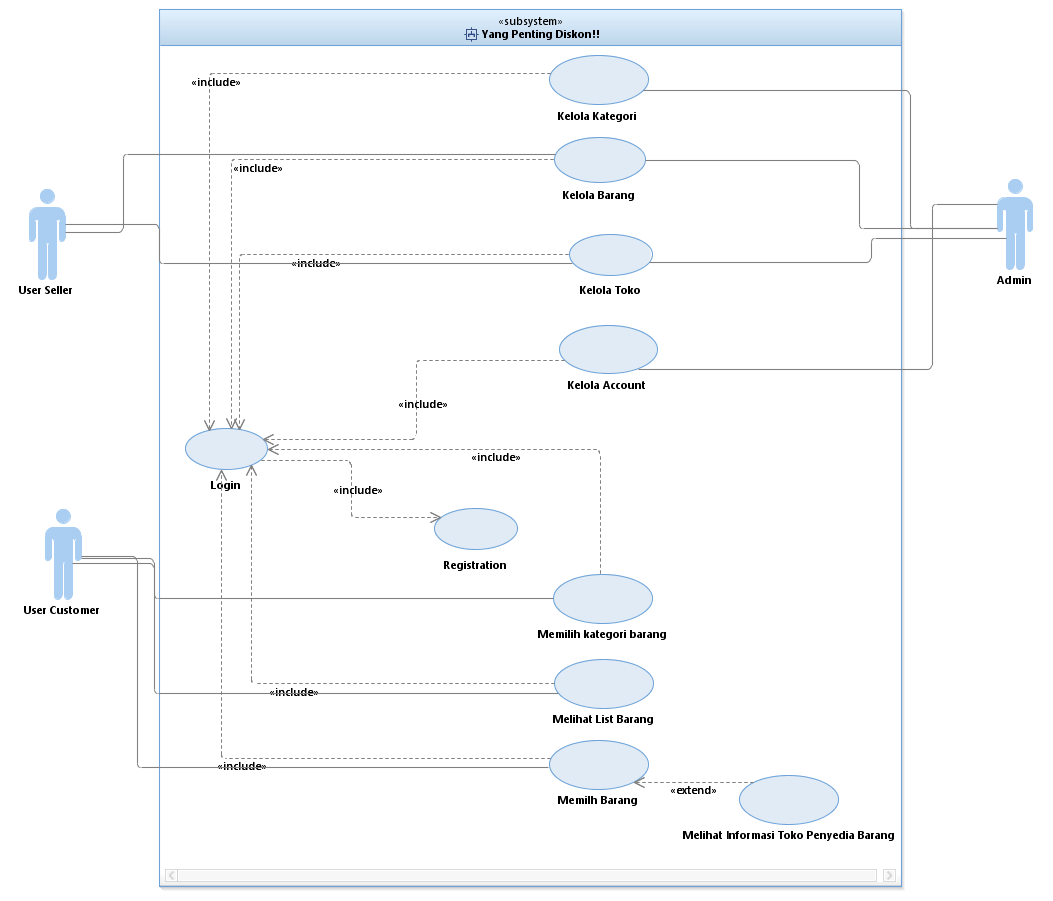
Untuk kebutuhan komunikasi utama sebernarnya tidak membutuhkan aplikasi pihak ketiga, atau aplikasi tambahan untuk dapat menggunakan aplikasi, komunikasi dapat terjalin melalui GUI pada aplikasi AYEDIS. Setiap device atau smartphone akan terhubung pada server dan saling berkomunikasi dengan server. Server dapat digunakan oleh beberapa user sekaligus dalam sebuah jaringan menggunakan koneksi internet.

Dalam melakukan pengamanan data, dilakukan dengan pemberian hak akses pada database atau grant hak akases. Setiap jenis user akan memiliki perbedaan hak akses, seperti admin yang dapat memodifikasi, berbagai hal yang ada pada aplikasi, dan pelanggan yang hanya bisa melakukan penambahan atau melihat saja.



*Gambar 6 antarmuka komunikasi*

# Fitur Sistem (Use Cases)



*Gambar 7 usecase diagram*

## Use Case 1

### Nama Use Case:

Memilih Kategori Barang.

### Tujuan:

Pelanggan dapat memilih kategori barang yang diinginkan.

### Input:

Klik menu kategori barang.

### Output:

*List* kategori barang ditampilkan oleh sistem.

### Skenario Utama:

Ketika *user customer* telah masuk kedalam aplikasi, *user customer* dapat memilih menu kategori barang pada menu utama aplikasi, lalu sistem akan menampilkan *list* kategori barang.

### Prakondisi:

Pengunjung yang telah terdaftar melakukan login pada aplikasi.

### Langkah-langkah:

#### Langkah 1: User customer melakukan proses login pada aplikasi.

#### .Langkah 2: Sistem menampilkan halaman utama aplikasi.

#### Langkah 3: User customer memilih menu kategori barang pada halaman utama.

#### Langkah 4: Sistem menampilkan pilihan kategori barang.

### Pascakondisi:

*List* kategori barang ditampilkan oleh sistem.

### Skenario eksepsional 1:

Ketika langkah 1 atau proses *login* terjadi kegagalan, sistem akan memberikan pemberitahuan *login* gagal, dan akan memberikan pilihan pada *user customer* untuk mencoba *login* kembali, memilih menu lupa *password*, dan melakukan registrasi.

## Use Case 2

### Nama Use Case:

Melihat *List* Barang.

### Tujuan:

*User customer* dapat melihat barang apa saja yang ditawarkan informasinya pada aplikasi.

### Input:

Klik menu pada *list* kategori barang.

### Output:

*List* barang berdasarkan kategori yang dipilih.

### Skenario Utama:

*User customer* yang telah melakukan *login* pada aplikasi dan telah memilih menu kategori barang pada halaman utama dapat melihat *list* barang dengan memilih salah satu kategori yang disediakan pada *list* kategori.

### Prakondisi:

*User customer* telah melakukan *login* pada aplikasi dan telah memilih kategori barang pada menu utama.

### Langkah-langkah:

#### Langkah 1: User customer melakukan login pada aplikasi.

#### .Langkah 2: Sistem menampilkan halaman utama dari aplikasi.

#### Langkah 3: User customer memilih menu kategori pada halaman utama.

#### Langkah 4: Sistem menampilkan pilihan kategori barang.

#### Langkah 5: User customer memilih salah satu kategori yang diinginkan.

#### Langkah 6: Sistem menampilkan list barang berdasarkan kategori yang telah dipilih.

### Pascakondisi:

*List* barang berdasarkan kategori yang dipilih ditampilakan kepada *user customer* oleh sistem.

### Skenario eksepsional 1:

Ketika langkah 1 atau proses *login* terjadi kegagalan, sistem akan memberikan pemberitahuan *login* gagal, dan akan memberikan pilihan pada *user customer* untuk mencoba lagi *login*, memilih menu lupa *password*, dan melakukan registrasi.

## Use Case 3

### Nama Use Case:

Memilih Barang.

### Tujuan:

*User customer* dapat memilih barang dari *list* barang yang tersedia.

### Input:

Klik pada produk yang ditampilkan pada *list* barang.

### Output:

Keranjang hasil pemilihan barang terisi.

### Skenario Utama:

*User customer* yang telah masuk kedalam aplikasi dan telah membuka salah satu fitur pada aplikasi yaitu melihat *list* barang dapat memilih salah satu atau lebih barang yang disediakan dan akan disimpan pada keranjang.

### Prakondisi:

*User customer* telah melakukan *login* pada aplikasi dan telah membuka menu sampai *list* barang.

### Langkah-langkah:

#### Langkah 1: User customer melakukan login pada aplikasi.

#### .Langkah 2: Sistem menampilkan halaman utama dari aplikasi.

#### Langkah 3: User customer memilih menu kategori pada halaman utama.

#### Langkah 4: Sistem menampilkan pilihan kategori barang.

#### Langkah 5: User customer memilih salah satu kategori yang diinginkan.

#### Langkah 6: Sistem menampilkan list barang berdasarkan kategori yang telah dipilih.

#### Langkah 7: User customer memilih barang yang disediakan oleh aplikasi dan akan masuk pada keranjang.

### Pascakondisi:

Keranjang akan terisi oleh barang-barang yang telah dipilih oleh *user customer*.

### Skenario eksepsional 1:

Ketika langkah 1 atau proses *login* terjadi kegagalan, sistem akan memberikan pemberitahuan *login* gagal, dan akan memberikan pilihan pada *user customer* untuk mencoba *login* lagi, memilih menu lupa *password*, dan melakukan registrasi.

### Skenario eksepsional 2:

Ketika barang yang dipilih telah habis, bisa berupa waktu dari diskon telah habis, dan stok barang telah habis, sistem akan memberikan notifikasi pada aplikasi kepada *user customer*.

## Use Case 4

### Nama Use Case:

Melihat Informasi Toko Penyedia Barang.

### Tujuan:

*User customer* dapat mengetahui informasi selengkap-lengkapnya dari toko penyedia barang, sehingga *user customer* memiliki informasi mulai dari detail produk sampai, detail toko penyedia seperti alamat yang harus dituju oleh *user customer*.

### Input:

Barang-barang yang telah dipilih oleh *user customer*.

### Output:

Informasi mendetail tentang toko yang barangnya dipilih oleh *user customer*.

### Skenario Utama:

*User customer* yang telah melakukan proses *login* pada aplikasi dan telah sampai pada tahap pemilihan barang, *user customer* akan mendapatkan informasi mendetail dari toko yang barangnya telah dipilih sebelumnya.

### Prakondisi:

*User customer* telah melakukan *login* dan telah sampai pada tahapan memilih barang.

### Langkah-langkah:

#### Langkah 1: User customer melakukan login pada aplikasi.

#### Langkah 2: Sistem menampilkan menu utama pada user customer.

#### Langkah 3: User customer memilih menu kategori pada halaman utama.

#### Langkah 4: Sistem menampilkan pilihan kategori barang.

#### Langkah 5: User customer memilih salah satu kategori yang diinginkan.

#### Langkah 6: Sistem menampilkan list barang berdasarkan kategori yang telah dipilih.

#### Langkah 7: User customer memilih barang atau produk yang disediakan oleh aplikasi dan memasukkannya kedalam keranjang.

#### Langkah 8: Sistem akan menampilkan informasi lengkap dari produk atau barang yang telah dipilih oleh user customer, informasi yang ditampilkan akan berisikan mulai dari detail produk sampai informasi detail dari toko yang menyediakan barang.

### Pascakondisi:

Sistem menampilkan informasi mendetail dari toko yang barangnya telah dipilih, mulai dari detail barang sampai detail toko seperti alamat dari toko.

### Skenario eksepsional 1:

Ketika langkah 1 atau proses *login* terjadi kegagalan, sistem akan memberikan pemberitahuan *login* gagal, dan akan memberikan pilihan pada *user customer* untuk mencoba lagi *login*, memilih menu lupa *password*, dan melakukan registrasi.

### Skenario eksepsional 2:

Ketika barang yang dipilih telah habis, bisa berupa waktu dari diskon telah habis, dan stok barang telah habis, sistem akan memberikan notifikasi pada aplikasi kepada *user customer*.

## Use Case 5

### Nama Use Case:

*Login*.

### Tujuan:

Setiap pengguna mulai dari *user customer*, *user seller*, dan admindapat masuk kedalam aplikasi dan menggunakannya.

### Input:

*Username* dan *password*.

### Output:

Notifikasi dari proses *login* berjalan suskses atau gagal.

### Skenario Utama:

*User customer, user seller*, dan admin membuka aplikasi lalu memasukkan *username* dan *password* pada kolom yang telah disediakan oleh aplikasi, bila berjalan sukses maka akan muncul notifikasi sukses dan akan menuju halaman utama dari aplikasi, dan bila gagal maka sistem akan memberikan opsi untuk melakukan *login* ulang, lupa *password*, dan registrasi.

### Prakondisi:

*User customer, user seller*, dan admin telah memasang aplikasi pada ponsel mereka yang berbasis *android*, dan membuka aplikasinya.

### Langgkah-langkah:

#### Langkah 1: User customer, user seller, dan admin membuka aplikasi.

#### Langkah 2: User customer, user aeller, dan admin melakukan input pada kolom username dan password.

#### Langkah 3: Sistem akan membawa user customer, user seller, dan admin ke halaman utama aplikasi.

### Pascakondisi:

Adanya notifikasi dari proses *login* apakah berjalan sukses atau gagal, bila sukses maka sistem akan membawa *user customer, user seller,* danadmin ke halaman utama dari aplikasi, dan bila tidak maka *user uustomer, user seller,* danadmin akan diberikan pilihan untuk melakukan *login* ulang, lupa *password,* danregistrasi.

### Skenario eksepsional 1:

Ketika proses *login* terjadi kegagalan, sistem akan memberikan pemberitahuan *login* gagal, dan akan memberikan pilihan pada *user customer* untuk mencoba lagi login, memilih menu lupa *password*, dan melakukan registrasi.

## Use Case 6

### Nama Use Case:

Registrasi

### Tujuan:

Calon pengguna aplikasi yang akan menjadi *user customer* atau *user seller*, dapat menikmati fitur yang ditawarkan.

### Input:

*Username, password*, nama depan, nama belakang, alamat, no telpon, *email*, dan alamat.

### Output:

Notifikasi dari proses registrasi berjalan suskses atau gagal.

### Skenario Utama:

Calon *user customer* dan *user seller* telah memasang aplikasi ini pada ponsel berbasis *android*, lalu mereka membuka aplikasi ini. Ketika mereka membuka aplikasi ini, maka sistem akan memberikan opsi untuk melakukan login atau melakukan registrasiatau *signup*, maka calon *user customer* dan *user seller* harus memilih registrasi. Setelah menu registrasi, maka akan ada *form* yang harus dilengkapi, setelah *form* dilengkapi maka sistem akan memberikan notifikasi proses berjalan sukses atau gagal.

### Prakondisi:

Calon *user customer*, dan *user seller* telah memasang aplikasi pada ponsel mereka yang berbasis *android*, dan membuka aplikasinya.

### Lankah-langkah:

#### Langkah 1: Calon user customer, dan user seller membuka aplikasi.

#### Langkah 2: Calon user customer, dan user seller memilih menu registrasi pada halaman awal.

#### Langkah 3: Sistem akan membawa calon user customer dan user seller pada halaman registrasi yang berisikan form yang harus dilengkapi oleh calon user customer dan user seller.

#### Langkah 4: Calon user customer dan user seller mengisi secara lengkap dan valid setiap field dari form yang diberikan.

#### Langkah 5: Sistem akan memberikan notifikasi apakah proses registrasi berjalan dengan sukses atau tidak, bila berjalan sukses maka calon user customer dan user seller telah menjadi user customer dan user seller, dan dapat menikmati setiap fitur yang ditawarkan.

### Pascakondisi:

Sistem memberikan notifikasi proses registrasi berjalan dengan sukses atau tidak.

### Skenario eksepsional 1:

Ketika proses registrasi terjadi kegagalan maka sistem akan memberikan notifikasi kegagalan, dan akan menunjuk *field* mana saja yang terjadi kesalahan *input* oleh calon *user customer* atau *user seller*. Kesalahan input dapat bermacam-macam, seperti alamat *email* yang tidak *valid*, alamat *email* yang telah terdaftar sebelumnya, dan kesalahan dalam format *input* yang benar. Sistem akan terus memberikan notifikasi gagal sampai setiap *field* pada *form* terisi dengan *valid*.

## Use Case 7

### Nama Use Case:

Kelola Toko.

### Tujuan:

Admindan *user seller* dapat menambahkan, menghapus, dan memperbaharui toko pada aplikasi, toko pada aplikasi akan bermanfaat agar *user customer* dapat melihat barang yang ditawarkan oleh *user seller*, dan *user seller* dapat memberikan informasi barang apa saja yang ingin dibagikan informasinya pada aplikasi. Pada aplikasi ini kenapa admin berperan juga melakukan *add/update/delete*, karena admin dapat berperan bila terjadi pelanggaran ketentuan yang dilakukan oleh *user seller*, maka admin dapat menghapus toko pada aplikasi, dan apabila *user seller* memiliki kesulitan dalam menambahkan dan memperbaharui toko, admin dapat memberikan bantuan.

### Input:

Nama toko, alamat toko, nomor telpon, *email*, dan keterangan singkat toko.

### Output:

Notifikasi *proses add, update,* atau *delete* sukses atau gagal.

### Skenario Utama:

*User seller* danadmin melakukan login pada aplikasi, lalu memilih menu kelola toko. Setelah proses memilih menu, sistem akan memberikan opsi untuk melakukan penambahan, hapus, atau *update* toko. Ketika salah satu dari menu dipilih dan dilakukan eksekusi, maka sistem akan memberikan notifikasi apakah proses berjalan dengan sukses atau tidak. Bila tidak berjalan dengan sukses maka sistem akan memberikan arahan pada *field* mana yang terjadi kesalahan *input*.

### Prakondisi:

*User seller* dan admin telah melakukan proses *login* pada aplikasi dan telah masuk ke halaman utama aplikasi.

### Langkah-langkah:

#### Langkah 1: User seller atau admin melakukan login pada aplikasi.

#### Langkah 2: User seller atau admin memilih menu kelola toko pada menu utama.

#### Langkah 3: Sistem memberikan opsi untuk melakukan add, update, atau delete.

#### Langkah 4: User seller atau admin memilih salah satu opsi.

#### Langkah 5: Sistem memberikan form bila yang dipilih adalah opsi update atau add, dan memberikan tampilan toko yang dimiliki oleh user seller untuk dilakukan proses penghapusan bila yang dipilih adalah opsi delete.

#### Langkah 6: Sistem memberikan notifikasi proses add, update, atau delete berjalan suskses atau gagal.

#### Langkah 7: Sistem menyimpan setiap perubahan yang terjadi baik itu berupa penambahan, pembaharuan, atau penghapusan pada database.

### Pascakondisi:

Sistem memberikan notifikasi proses berjalan dengan sukses atau tidak, dan menyimpan perubahan yang dilakukan baik itu penambahan, pembaharuan, atau penghapusan pada *database*.

### Skenario eksepsional 1:

Ketika proses *login* terjadi kegagalan, sistem akan memberikan pemberitahuan *login* gagal, dan akan memberikan pilihan pada *user customer* untuk mencoba lagi *login*, memilih menu lupa *password*, dan melakukan registrasi.

### Skenario eksepsional 2:

Bila terjadi kesalahan dalam proses *add* toko, maka sistem akan memberikan notifikasi kegagalan dan menunjuk *field* yang terjadi kesalahan *input* untuk dibenarkan. Sistem akan terus menunjuk *field* yang salah sampai *input* yang diberikan *valid*.

### Skenario eksepsional 3:

Bila terjadi kesalahan dalam proses *delete* toko, maka sistem akan memberikan notifikasi kegagalan lalu memberikan informasi kegagalan terletak dimana. Kegagalan dapat terjadi karena *user seller* masih belum memiliki toko, dan bisa juga karena kesalahan *input* data yang dilakukan.

### Skenario eksepsional 4:

Bila terjadi kesalahan dalam proses *update* toko, maka sistem akan memberikan notifikasi kegagalan lalu memberikan informasi letak kegagalan terjadi dimana. Kegagalan dapat terjadi karena *user* *seller* belum memiliki toko pada aplikasi, dan dapat pula terjadi karena kesalahan *input* data yang tidak *valid*. Sistem akan terus menunjuk pada kesalahan sampai dilakukan prosedur atau *input* data yang *valid*.

## Use Case 8

### Nama Use Case:

Kelola Akun.

### Tujuan:

Admin dapat melihat semua data dari *users* yang melakukan pendaftaran pada aplikasi, sehingga admin dapat melakukan tindakan penghapusan atau *banned* pada akun *user* yang melakukan pelanggaran. FiturKelola akun ini berfungsi agar keamanan dan kenyamanan pada aplikasi terjaga.

### Input:

Klik pada menu “Kelola akun”.

### Output:

*List* semua *users* yang telah melakukan registrasi dan keterangan aktivitas yang dilakukan.

### Skenario Utama:

Setiap *user* yang melakukan pendaftaran pada aplikasi akan disimpan datanya pada *database*. Data *users* yang berada pada *database* diakses oleh admin dengan memilih fitur “Kelola Akun “ pada menu utama aplikasi, hal ini bertujuan untuk pengawasan sehingga ketika terjadi pelanggaran, admin dapat bertindak.

### Prakondisi:

Terdapat data *users* yang telah terdaftar pada aplikasi pada *database*.

### Lankah-langkah:

#### Langkah 1: Admin melakukan proses login pada aplikasi.

#### Langkah 2: Admin memilih fitur “Kelola Akun” pada halaman utama.

#### Langkah 3: Sistem memberikan list akun yang telah terdaftar pada aplikasi dan keterangan kegiatan apa saja yang dilakukan oleh users.

#### Langkah 4: Admin melakukan tindakan pada setiap akun yang terdaftar.

### Pascakondisi:

Admin telah melakukan tindakan pada akun yang bermasalah, dapat berupa penghapusan akun.

### Skenario eksepsional 1:

Ketika proses *login* terjadi kegagalan, sistem akan memberikan pemberitahuan *login* gagal, dan akan memberikan pilihan pada *user customer* untuk mencoba lagi *login*, memilih menu lupa *password*, dan melakukan registrasi.

### Skenario eksepsional 2:

Belum ada data akun yang tersimpan pada *database*, maka sistem akan memberikan notifikasi *database* akun masih kosong pada admin.

## Use Case 9

### Nama Use Case:

Kelola Produk.

### Tujuan:

*User seller* dan admin dapat menambah, memperbaharui, dan menghapus produk, sehingga informasi yang diberikan pada *user customer* adalah selalu yang terbaru dan terakurat.

### Input:

Nama produk, id produk, tanggal kadaluarsa, besar diskon, nama kategori, id toko dan ulasan singkat produk.

### Output:

Notifikasi telah terjadi perubahan pada *database*.

### Skenario Utama:

Admin dan setiap *user seller* yang telah login kedalam aplikasi akan membuka menu “Kelola Produk”, setiap produkyang mereka sediakan informasinya pada aplikasi dapat diperbaharui, ditambah, dan dihapus, sesuai dengan keadaan dan kebutuhan dari *user seller*. Ketika salah satu proses dieksekusi atau dilakukan, maka sistem akan menyimpan perubahan pada *database*, dan pada aplikasi.

### Prakondisi:

Admin dan *user seller* telah masuk dan *login* kedalam aplikasi, dan akan melakukan perubahan baik itu penambahan, pembaharuan, atau penghapusan.

### Langkah-langkah:

#### Langkah 1: Admin dan user seller melakukan proses login pada aplikasi.

#### Langkah 2: Admin memilih fitur “Kelola Produk” pada halaman utama.

#### Langkah 3: Sistem memberikan list semua toko apabila yang melakukan admin, dan apabila yang melakukan adalah user seller maka sistem hanya akan menampilkan toko yang dimiliki oleh user seller yang bersangkutan.

#### Langkah 4: Admin dan user seller memilih toko yang produk didalamnya akan dilakukan penambahan, pembaharuan, atau penghapusan.

#### Langkah 5: Sistem memberikan list barang yang ada pada toko dan diberikan opsi untuk melakukan pembaharuan atau penghapusan. Bila yang akan dilakukan adalah penambahan produk maka yang dipilih adalah kolom “Tambah Produk”.

#### Langkah 6: Bila yang dipilih adalah “Tambah Produk”, sistem akan membawa user seller dan admin kehalaman untuk melakukan penambahan produk.

#### Langkah 7: Admin dan user seller mengisikan form tentang produk baru yang akan ditambahkan pada toko, mulai dari id produk sampai ulasan singkat produk.

#### Langkah 8: Sistem menyimpan setiap perubahan baik itu penambahan, pembaharuan, dan penghapusan pada database dan telah terjadi perubahan pada aplikasi.

### Pascakondisi:

*Database* telah menyimpan setiap perubahan yang dilakukan oleh admin dan *user seller*, baik itu penambahan, pembaharuan, atau penghapusan. Dan telah terjadi perubahan pada menu produk di aplikasi.

### Skenario eksepsional 1:

Ketika proses *login* terjadi kegagalan, sistem akan memberikan pemberitahuan *login* gagal, dan akan memberikan pilihan pada *user customer* untuk mencoba lagi *login*, memilih menu lupa *password*, dan melakukan registrasi.

### Skenario eksepsional 2:

Apabila *user seller* belum memiliki toko maka tidak dapat dilakukan kegiatan penambahan, pembaharuan, dan penghapusan produk, dan sistem akan memberikan notifikasi dan memberikan opsi untuk membuat toko kepada *user seller*.

### Skenario eksepsional 3:

Apabila toko masih kosong, atau belum terdapat produk, maka sistem akan memberikan notifikasi toko kosong, dan hanya dapat dilakukan penambahan produk saja pada toko.

## Use Case 10

### Nama Use Case:

Kelola Kategori.

### Tujuan:

Admin dapat melakukan penambahan, pembaharuan, dan penghapusan pada kategori yang ada pada aplikasi, ini bermanfaat ketika terjadi perkembangan yang sangat pesat pada bidang ekonomi, terutama ekonomi yang berbasis digital, dimana perubahan dan masuknya hal-hal baru yang sangat cepat. Sehingga diharapkan aplikasi dapat selalu beradaptasi dan memberikan layanan yang terbaru dan terakurat dapat *user customer*.

### Input:

ID kategori, nama kategori, dan ulasan singkat kategori.

### Output:

Terdapat perubahan pada menu kategori, bisa berupa penambahan, pembaharuan, atau penghapusan kategori.

### Skenario Utama:

Admin yang telah masuk kedalam aplikasi dan melakukan proses *login* kemudian memilih menu “Kelola Kategori” pada halaman utama aplikasi. Dihalaman tersebut admin dapat melakukan penambahan, pembaharuan, atau penghapusan sesuai dengan kebutuhan dari aplikasi.

### Prakondisi:

Admin dan *user seller* telah masuk dan *login* kedalam aplikasi. Untuk menambahkan kategori baru, harus yang belum ada pada aplikasi, dan apabila ingin melakukan pembaharuan atau penghapusan, maka harus pada kategori yang sudah ada pada aplikasi.

### Langkah-langkah:

#### Langkah 1: Admin dan user seller melakukan proses login pada aplikasi.

#### Langkah 2: Admin memilih fitur “Kelola Kategori” pada halaman utama.

#### Langkah 3: Sistem memberikan list semua kategori yang ada pada aplikasi untuk dilakukan penghapusan atau pembaharuan, dan terdapat juga opsi untuk menambahkan kategori baru pada aplikasi.

#### Langkah 4: Admin melakukan salah satu perubahan, dapat berupa penambahan, pembaharuan, atau penghapusan pada kategori yang ada.

#### Langkah 5: Sistem menyimpan setiap perubahan yang dilakukan oleh admin, baik itu perubahan berupa penambahan, pembaharuan, atau penghapusan pada database dan aplikasi.

### Pascakondisi:

*Database* telah menyimpan setiap perubahan yang dilakukan oleh admin, baik itu penambahan, pembaharuan, atau penghapusan. Dan telah terjadi perubahan pada menu kategori di aplikasi

### Skenario eksepsional 1:

Ketika proses *login* terjadi kegagalan, sistem akan memberikan pemberitahuan *login* gagal, dan akan memberikan pilihan pada *user customer* untuk mencoba lagi *login*, memilih menu lupa *password*, dan melakukan registrasi.

### Skenario eksepsional 2:

Apabila terjadi saat proses penambahan kategori terjadi kesalahan, maka sistem akan memberikan notifikasi pada admin. Bila kesalahan terletak pada pembuatan kategori yang sudah ada pada aplikasi, maka sistem akan memberikan saran untuk mengganti kategori baru atau keluar dari menu tambah kategori, namun apabila kesalahan terletak pada format *input* yang dilakukan, maka sistem akan menunjuk pada *field* yang *input-*nya tidak *valid*.

### Skenario eksepsional 3:

Apabila saat proses penghapusan dan pembaharuan dilakukan terjadi, maka sistem akan memberikan notifikasi juga. Bila kesalahan karena *input* yang tidak *valid* maka sistem akan menunjuk pada bagian *field* yang tidak *valid*, namun apabila kategori yang ingin dilakukan penghapusan atau pembaharuan tidak ada, maka sistem hanya akan memberikan opsi untuk menambah kategori baru.

# Requirements Nonfungsional Lainnya

## Requirements Performa

Aplikasi AYEDIS (Aplikasi Yang Penting Diskon) ini akan berjalan dalam performa terbaiknya jika dijalankan dalam spesifikasi *hardware* di atas spesifikasi minimal yang diberikan. Versi sistem operasi yang digunakan juga akan sangat berpengaruh terhadap kinerja dari sistem. Selain itu, tindakan perawatan terhadap sistem, seperti *back-up* data pun akan mempengaruhi kecepatan akses data oleh sistem. Dalam performa ketika menggunakan basis data akan dibuat semaksimal mungkin dengan cara melakukan perhitungan estimasi pengunjung yang mengakses aplikasi dalam satu waktu tertentu. Selain perhitungan estimasi pengunjung, *query* yang dilakukan akan dibuat semaksimal dan seefisien mungkin, sehingga performa pengeksekusian *query* akan cepat.

## Requirements Keselamatan

Aplikasi AYEDIS (Aplikasi Yang Penting Diskon) akan lebih baik apabila dilakukan *back-up* data secara berkala untuk menghindari kemungkinan *server down* yang dapat mengakibatkan hilangnya beberapa atau bahkan keseluruhan data yang tersimpan dalam *database* di *server*. Pihak *developer* tidak bertanggung jawab atas kerusakan atau kehilangan data yang disebabkan oleh kelalaian dari pihak *user* (admin).

## Requirements Keamanan

Aplikasi AYEDIS (Aplikasi Yang Penting Diskon) juga perlu memperhatikan keamanan. Untuk menjaga keamanan data yang terdapat pada server, maka hanya ada satu akun *user* yang berhak mengakses komputer *server* secara langsung, yaitu akun administrator (admin) jadi tidak ada satu orangpun yang dapat memanipulasi data selain admin. Admin juga berhak melakukan *banned* atau penghapusan pada akunatau data yang melakukan pelanggaran, sehingga keamanan dan kenyamanan aplikasi tetap terjaga. Sementara itu, untuk menghindari penggunaan sistem oleh pihak yang tidak memiliki hak, digunakan sistem *password* dimana masing-masing *user* diharuskan untuk mengisi *username* dan *password* terlebih dahulu untuk dapat memasuki aplikasi AYEDIS.

Untuk keamanan *data base* yang diakses oleh *user,* baik itu *user seller* dan *user customer* akan dilakukan pengamanan dengan hak *grant* akses atau penggunaan atotorifikasi hal apa saja yang dapat diakses dan digunakan oleh *user* yang menggunakan aplikasi. Hal ini dilakukan untuk meminimalisir pencurian data ketika data dipanggil dari *data base* menggunakan *query* pada aplikasi.

## Atribut Kualitas Perangkat Lunak

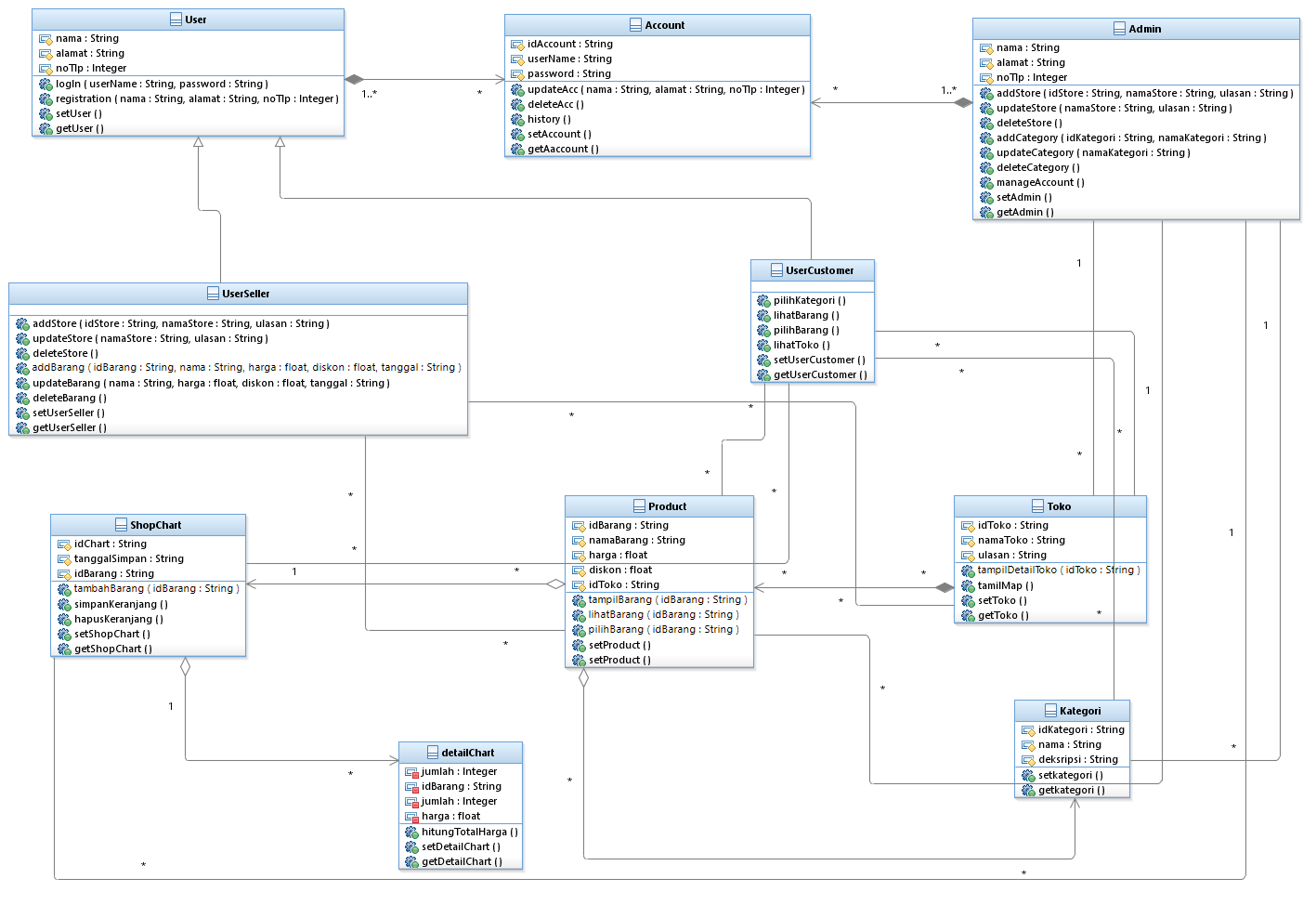
Kelebihan Aplikasi AYEDIS (Aplikasi Yang Penting Diskon) diantaranya yaitu fleksibilitas yang memungkinkan aplikasi ini dapat digunakan dimana saja dan kapan saja hanya dengan menggunakan *smartphone* yang terhubung dengan koneksi internet. Aplikasi ini juga mudah di-*install* dan dapat mudah digunakan karena disertai fitur *online help* untuk membantu *user* dalam penggunaan aplikasi. Keamanan aplikasi juga terjaga dengan adanya sistem *password* dan *server* hanya dipegang oleh satu akun administrator*.* Selain itu, aplikasi ini juga tidak hanya memberi informasi alamat toko, melainkan akan menunjukkan dan mengarahkan ke tempat lokasi dengan API *Google map* sehingga memudahkan penggunanya.

# Requirements Lain

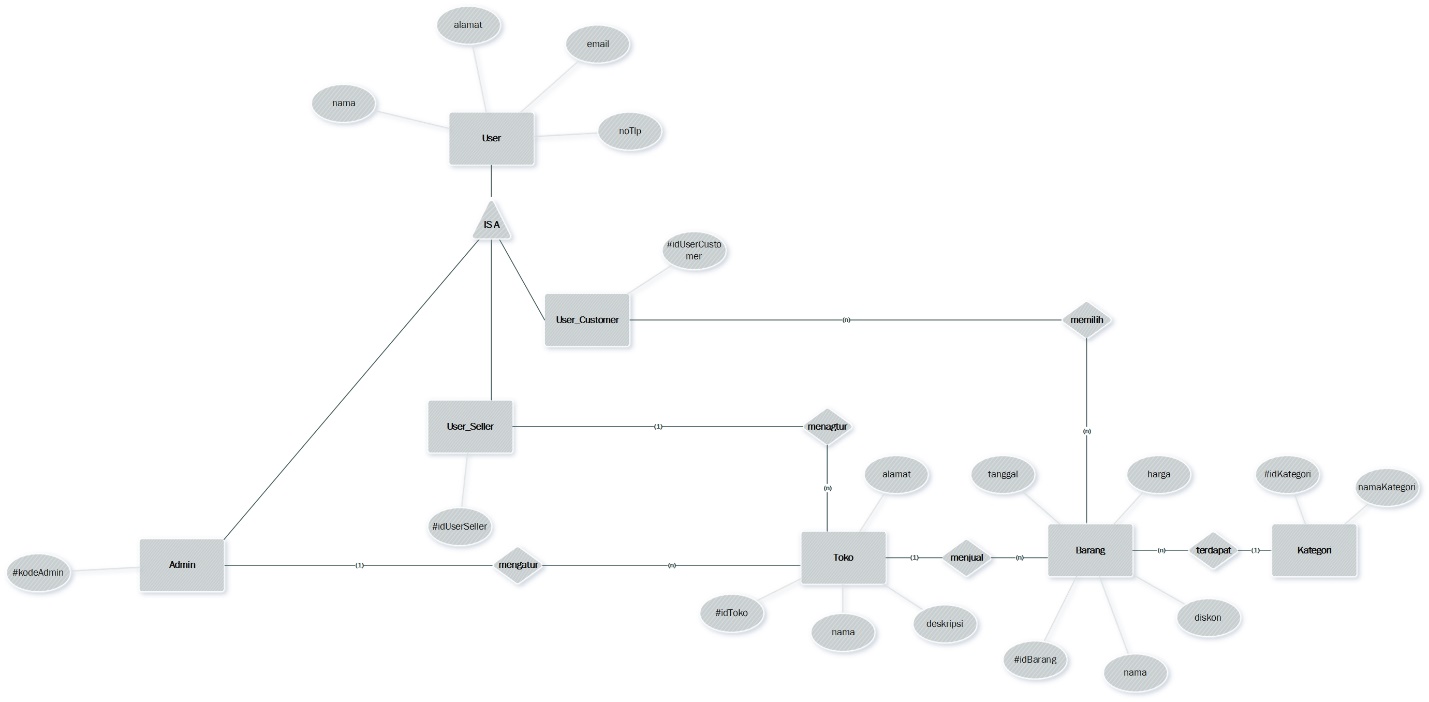
Bersama *software* ini tidak disertakan semua *software-software* pihak ketiga yang menjadi persyaratan untuk berjalannya Aplikasi AYEDIS (Aplikasi Yang Penting Diskon) ini dengan baik, seperti sistem operasi, *Oracle* sebagai DBMS, dan sebagainya. Oleh karena itu, disamping semua kebutuhan yang sudah dipaparkan, hendaknya untuk tidak menggunakan *software* bajakan.

|  |  |  |
| --- | --- | --- |
| Lampiran A: Daftar Kata-Kata Sukar | | |
| NO | Kata-Kata Sukar | Penjelasan |
| 1 | SKPL | Spesifikasi Kebutuhan Perangkat Lunak, atau dalam bahasa Inggrisnya sering juga disebut sebagai *Software Requirements Spesification (SRS),* dan merupakan spesifikasi dari perangkat lunak yang akan dikembangkan. |
| 2 | Use Case Diagram | Gambaran grafis dari beberapa atau semua *actor, use-case,* dan interaksi antara komponen-komponen tersebut yang memperkenalkan satu sistem yang dibangun. |
| 3 | Use Case Scenario | Narasi untuk menjelaskan *use case* yang ada sehingga *customer* lebih mudah memahami fungsi dari *use case* yang ada. |
| 4 | API | (*Application Programming Interface)* adalah sekumpulan instruksi program dan protokol yang digunakan untuk membangun aplikasi perangkat lunak. |
| 5 | Smartphone | Telepon genggam yang mempunyai kemampuan tingkat tinggi, kadang-kadang dengan fungsi yang menyerupai komputer. |
| 6 | GUI | *(Graphical User Interface),* adalah antarmuka pada sistem operasi atau komputer yang menggunakan menu grafis agar mempermudah para pengguna-nya untuk berinteraksi dengan komputer atau sistem operasi. |
| 7 | Android | Sistem operasi berbasis *Linux* yang dirancang untuk perangkat bergerak layar sentuh seperti telepon pintar dan komputer tablet. |
| 8 | Jellybean | Salah satu jenis sistem operasi pada Android. |
| 9 | Oracle | Salah satu produk yang bisa digunakan untuk mengolah data atau mengumpulkan data. |
| 10 | Quadcore | Salah satu jenis prosesor. |
| 11 | RAM | Suatu memori tempat penyimpanan data sementara. |
| 12 | DBMS | Sistem pengorganisasian dan sistem pengolahan *database* pada komputer. |
| 13 | Akun | Catatan rekening, akun. |
| 14 | User Customer | Pengguna yang mendapatkan informasi. |
| 15 | User Seller | Pengguna yang mempromosikan produk (toko). |
| 16 | Admin | Pengguna yang mengelola sistem. |
| 17 | Back up | Metode menyimpan data ditempat terpisah untuk suatu alasan. |
| 18 | Database | Basis data. |
| 19 | Server down | Server yang mengalami kerusakan atau terganggu. |
| 20 | Login | Masuk ke dalam suatu sistem dengan syarat tertentu. |
| 21 | Password | Kata kunci yang digunakan untuk keamanan. |
| 22 | Add | Menambahkan. |
| 23 | Update | Memperbaharui. |
| 24 | Delete | Menghapus. |

Lampiran B: Analysis Models

1. Class Diagram

Gambar 8 class diagram

1. ERD

Gambar 9 ERD

1. Kamus Data

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NO | Nama Data | Atribut | Tipe Data | PK/FK |
| 1 | User Seller | * idUserSeller * nama * alamat * email * noTlp | * varchar(50) * varchar(50) * varchar(100) * varchar(50) * numeric | PK |
| 2 | User Customer | * idUserCustomer * nama * alamat * email * noTlp | * varchar(50) * varchar(50) * varchar(100) * varchar(50) * numeric | PK |
| 3 | Admin | * kodeAdmin * nama * alamat * email * noTlp | * varchar(50) * varchar(50) * varchar(100) * varchar(50) * numeric | PK |
| 4 | Barang | * idBarang * idToko * idKategori * idDetail * nama * harga * diskon * tanggal | * varchar(50) * varchar(50) * varchar(50) * varchar(50) * varchar(50) * real * real * date | PK  FK  FK  FK |
| 5 | Kategori | * idKategori * namaKategori | * varchar(50) * varchar(50) | PK |
| 6 | Toko | * idToko * kodeAdmin * idUserSeller * nama * deskripsi * alamat | * varchar(50) * varchar(50) * varchar(50) * varchar(50) * varchar(200) * varchar(100) | PK  FK  FK |
| 7 | Shop Chart | * idShopChart * idUserCustomer * tglSimpan * totalHarga | * varchar(50) * varchar(50) * varchar(50) * real | PK  FK |
| 8 | Chart Detail | * idDetail * idShopChart * tanggalBarang * diskon * harga | * varchar(50) * varchar(50) * date * real * real | PK  FK |