

**APLIKASI MINIMARKET KING MART**

**PROYEK AKHIR 1**

Oleh :

**Nurul Hadi 3311211060**

**Abdul Rahman Saputra 3311211051**

**Durojak 3311211066**

**M.Arrozi 331121143**



**PROGRAM STUDI TEKNIK INFORMATIKA**

**POLITEKNIK NEGERI BATAM**

**BATAM**

**2013**

**DAFTAR ISI**

DAFTAR ISI.....	ii
BAB I PENDAHULUAN.....	1
1.1 LATAR BELAKANG.....	1
1.2 RUMUSAN MASALAH.....	1
1.3 BATASAN MASALAH.....	1
1.4 TUJUAN.....	2
1.5 SISTEMATIKA PENULISAN.....	2
BAB II LANDASAN TEORI.....	3
2.1 DEFINISI NETBEANS.....	3
2.2 DEFINISI JAVA.....	7
2.3 UML(Unified Modelling Language).....	7
2.4 POINTS OF SALES.....	8
BAB III ANALISIS DAN PERANCANGAN.....	9
3.1 DESKRIPSI UMUM.....	9
3.2 USE CASE DIAGRAM.....	10
3.3 SKENARIO USE CASE.....	11
3.4 SEQUENCE DIAGRAM.....	14
3.4.1 SEQUENCE DIAGRAM LOGIN.....	15
3.4.2 SEQUENCE DIAGRAM TRANSAKSI.....	16
3.4.3 SEQUENCE DIAGRAM PENGELOLAAN DATA BARANG.....	16
BAB IV PEMBAHASAAN.....	17
4.1 IMPLEMENTASI.....	17
4.1.1 IMPLEMENTASI LOGIN.....	19
4.1.2 IMPLEMENTASI TRANSAKSI.....	21
4.1.3 IMLEMENTASI UPDATE BARANG.....	23
BAB V KESIMPULAN DAN SARAN.....	
5.1 Kesimpulan.....	
5.2 Saran.....	

## **BAB I PENDAHULUAN**

## 1.1 LATAR BELAKANG

Saat ini mini market sudah tersebar luas dimana-mana, setiap minimarket sudah menggunakan aplikasi untuk mempercepat transaksi, tetapi tidak seperti mini market yang kami jumpai yaitu minimarket kingmart. minimarket dalam bertransaksi masih menggunakan secara manual, seperti menghitung total pembayaran menggunakan kalkulator, mendata pemasukan barang menggunakan buku. minimarket ini tidak tersedia aplikasi yang mendukung untuk mempercepat transaksi penjualan.

Semakin berkembangnya teknologi, komputer sudah menjadi bagian yang sangat penting dan bermanfaat dalam segala hal. salah satu manfaat dari komputer adalah bertransaksi menggunakan perangkat lunak atau sering disebut aplikasi. pentingnya akan sebuah aplikasi minimarket dan banyaknya minimarket yang berkembang saat ini memberi inspirasi terhadap penulis untuk membuat aplikasi minimarket menggunakan bahasa java. yang bertujuan untuk lebih mempermudah bertransaksi antara kasir dan pembeli.

## 2.2 RUMUSAN MASALAH

Berdasarkan latar belakang diatas penulis merumuskan masalah sebagai berikut :

- Bagaimana mempercepat transaksi penjualan di minimarket?
- Bagaimana mempermudah pengecekan stok barang?

## 2.3 BATASAN MASALAH

Berdasarkan rumusan masalah, penulis pembatasan masalah yang akan dibahas adalah:

- Aplikasi hanya bisa di jalankan di windows
- Aplikasi digunakan secara offline
- Aplikasi hanya digunakan 2 user
- Admin hanya dapat pengelolaan data barang
- Kasir hanya dapat bertransaksi

## **2.4 TUJUAN**

Tujuan penulis dalam membuat tugas proyek akhir ini adalah

- Mempercepat transaksi penjualan
- Mempermudah pengecekan stok barang

## **2.5 SISTEMATIKA PENULISAN**

Untuk memberika gambaran umum dari penulisan tugas proyek akhir 1 ini maka penulis membuat uraian singkat dari tiap-tiap BAB yang ada didalam penulisan tugas proyek akhir 1 ini

## **BAB I PENDAHULUAN**

Pada bab ini menerangkan pendahuluan yang berisi tentang latar belakang,masalah,tujuan penulisan,rumusan masalah,batasan masalah,dan sistematika penulisan.

## **BAB II LANDASAN TEORI**

Pada bab ini akan dibahas tentang pemrograman serta penjelasan yang terdapat pada software.

## **BAB III ANALISI DAN PERANCANGAN**

Pada bab ini akan di bahas mengenai cara pembuatan sebuah aplikasi minimarket dengan menggunakan bahasa pemrograman java yang di lengkapi dengan use case, class diagram dan sequence diagram.

## **BAB IV PEMBAHASAN**

Pada bab ini membahas tentang implementasi aplikasi mini market dilengkapi gambar,source code program.

## **BAB V KESIMPULAN DAN SARAN**

Pada bab ini membahas tentang kesimpulan dan saran

## **BAB II LANDASAN TEORI**

### **2.1 Netbeans**

Netbeans merupakan sebuah aplikasi *Integrated Development Environment (IDE)* yang berbasiskan Java dari Sun *Microsystems* yang berjalan di atas swing. Swing merupakan sebuah teknologi Java untuk pengembangan aplikasi desktop yang dapat berjalan pada berbagai macam platform seperti windows, linux, Mac OS X dan Solaris. Sebuah IDE merupakan lingkup pemrograman yang di integrasikan ke dalam suatu aplikasi perangkat lunak yang menyediakan Graphic User Interface (GUI), suatu kode editor atau text, suatu compiler dan suatu debugger.

Netbeans juga digunakan oleh sang programmer untuk menulis, meng-compile, mencari kesalahan dan menyebarkan program netbeans yang ditulis dalam bahasa pemrograman java namun selain itu dapat juga mendukung bahasa pemrograman lainnya dan program ini pun bebas untuk digunakan dan untuk membuat professional desktop, enterprise, web, and mobile applications dengan Java language, C/C++, dan bahkan dynamic languages seperti PHP, JavaScript, Groovy, dan Ruby.

NetBeans merupakan sebuah proyek kode terbuka yang sukses dengan pengguna yang sangat luas, komunitas yang terus tumbuh, dan memiliki hampir 100 mitra (dan terus bertambah!). Sun Microsystems mendirikan proyek kode terbuka NetBeans pada bulan Juni 2000 dan terus menjadi sponsor utama. Dan saat ini pun netbeans memiliki 2 produk yaitu Platform Netbeans dan Netbeans IDE. Platform Netbeans merupakan framework yang dapat digunakan kembali (reusable) untuk menyederhanakan pengembangan aplikasi deskto dan Platform NetBeans juga menawarkan layanan-layanan yang umum bagi aplikasi desktop, memungkinkan pengembang untuk fokus ke logika yang spesifik terhadap aplikasi.

**Fitur-fitur dari Platform Netbeans antara lain:**

- Manajemen antarmuka (misal: menu & toolbar)
- Manajemen pengaturan pengguna
- Manajemen penyimpanan (menyimpan dan membuka berbagai macam data)
- Manajemen jendela

- Wizard framework (mendukung dialog langkah demi langkah)

Netbeans IDE merupakan sebuah IDE open source yang ditulis sepenuhnya dengan bahasa pemrograman java menggunakan platform netbeans. NetBeans IDE mendukung pengembangan semua tipe aplikasi Java (J2SE, web, EJB, dan aplikasi mobile). Fitur lainnya adalah sistem proyek berbasis Ant, kontrol versi, dan refactoring.

Versi terbaru saat ini adalah NetBeans IDE 5.5.1 yang dirilis Mei 2007 mengembangkan fitur-fitur Java EE yang sudah ada (termasuk Java Persistence support, EJB-3 dan JAX-WS). Sementara paket tambahannya, NetBeans Enterprise Pack mendukung pengembangan aplikasi perusahaan Java EE 5, meliputi alat desain visual SOA, skema XML, web service dan pemodelan UML. NetBeans C/C++ Pack mendukung proyek C/C++.

Modularitas: Semua fungsi IDE disediakan oleh modul-modul. Tiap modul menyediakan fungsi yang didefinisikan dengan baik, seperti dukungan untuk bahasa pemrograman Java, editing, atau dukungan bagi CVS. NetBeans memuat semua modul yang diperlukan dalam pengembangan Java dalam sekali download, memungkinkan pengguna untuk mulai bekerja sesegera mungkin. Modul-modul juga memungkinkan NetBeans untuk bisa dikembangkan. Fitur-fitur baru, seperti dukungan untuk bahasa pemrograman lain, dapat ditambahkan dengan menginstal modul tambahan. Sebagai contoh, Sun Studio, Sun Java Studio Enterprise, dan Sun Java Studio Creator dari Sun Microsystems semuanya berbasis NetBeans IDE.

Fitur fitur yang terdapat dalam netbeans antara lain:

1. Smart Code Completion: untuk mengusulkan nama variabel dari suatu tipe, melengkapi keyword dan mengusulkan tipe parameter dari sebuah method.
2. Bookmarking: fitur yang digunakan untuk menandai baris yang suatu saat hendak kita modifikasi.
3. Go to commands: fitur yang digunakan untuk jump ke deklarasi variabel, source code atau file yang ada pada project yang sama.

4. Code generator: jika kita menggunakan fitur ini kita dapat meng-generate constructor, setter and getter method dan yang lainnya.

5. Error stripe: fitur yang akan menandai baris yang eror dengan memberi highlight merah.

#### Paket-Paket Tambahan NetBeans IDE

##### NetBeans Mobility Pack

NetBeans Mobility Pack adalah alat untuk mengembangkan aplikasi yang berjalan pada perangkat bergerak (mobile), umumnya telepon seluler, tetapi juga mencakup PDA, dan lain-lain.

##### NetBeans Profil

NetBeans Profiler adalah alat untuk mengoptimalkan aplikasi Java, membantu menemukan kebocoran memori dan mengoptimalkan kecepatan. NetBeans C/C++ pack NetBeans C/C++ Pack menambahkan dukungan terhadap pengembang C/C++ ke NetBeans IDE 5.5. Paket ini memperbolehkan pengembang menggunakan sekumpulan kompiler dan alat sendiri bersama dengan NetBeans IDE untuk membangun aplikasi native untuk MS Windows, Linux, dan Solaris. Paket ini membuat editor mengenali bahasa C/C++ dan menyediakan project template, browser kelas yang dinamis, dukungan pembuatan file dan fungsionalitas debugger. Para pengembang juga dapat mengembangkan paket tersebut dengan fungsionalitas tambahan mereka sendiri.

NetBeans Enterprise Pack memperluas dukungan terhadap pengembangan aplikasi perusahaan dan web service di NetBeans IDE 5.5. Enterprise Pack ini mengembangkan kemampuan untuk menulis, menguji, dan debug aplikasi dengan arsitektur berorientasi layanan (Service-Oriented Architecture) menggunakan XML, BPEL, dan Java web service. Paket ini menambahkan alat desain visual untuk pemodelan UML, skema XML, dan web service orchestration NetBeans Enterprise pack

NetBeans Versi NetBeans 6.0 mendatang akan mengijinkan pengembangan IDE menggunakan Ruby dan Jruby, sebagaimana Rails untuk dua



implementasi Ruby yang lain. Preview NetBeans Ruby Pack tersedia sejak rilis Milestone 7 NetBeans Ruby pack

### **Database yang didukung oleh Netbeans antara lain:**

1. JDBC merupakan spesifikasi standar dari JavaSoft API (Application Programming Interface) yang memungkinkan program Java untuk mengakses sistem database manajemen. JDBC API terdiri dari satu set interface dan kelas yang ditulis dalam bahasa pemrograman Java. JDBC API Menggunakan interface standar dan kelas, programmer dapat menulis aplikasi yang terhubung ke database, mengirimkan pertanyaan ditulis SQL (Structured Query Language), dan memproses hasilnya.
2. ORACLE merupakan sebuah aplikasi basisdata yang didukung oleh NetBeans, Oracle secara umum hampir sama dengan MySQL namun yang membedakan adalah oracle dapat digunakan dan dihubungkan dengan netbeans dan harus menggunakan drivers untuk menyimpan data-data yang telah dibuat. Oracle merupakan salah satu dari beberapa aplikasi basisdata yang sering digunakan untuk koneksi ke basisdata pada NetBeans karena relatif mudah dan cepat.
3. MySQL merupakan sebuah aplikasi basisdata yang didukung oleh NetBeans, MySQL memiliki Fitur Yang ada sudah lumayan lengkap, dari input, update, delete serta Search. 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.

### **Berikut sekilas mengenai beberapa sintaks SQL yang digunakan pada NetBeans:**

#### **1. Interface :**

- `java.sql.Statement` Interface yang digunakan untuk mengirim laporan statis SQL ke server database dan memperoleh hasilnya.
- `javax.sql.ConnectionEventListener` Menerima event atau kejadian bahwa obyek `PooledConnection` digeneralisasi.

## **2. Classes :**

- `java.sql.Date` Subclass dari `java.util.Date` digunakan untuk tipe data SQL DATE.
- `java.lang.DriverManager` Class yang digunakan untuk mengelola satu set JDBC drivers.

## **3. Exception Classes:**

- `java.sql.SQLException` Eksepsi yang menyediakan informasi tentang Database error
- `java.sql.SQLWarning` Eksepsi yang menyediakan informasi tentang peringatan Database.

## **4. DataBase Access**

(Microsoft Access) merupakan sebuah aplikasi basis data yang didukung oleh NetBeans, data dapat disimpan di dalam format Microsoft Access, Microsoft Jet Database Engine, Microsoft SQL Server, Oracle Database, atau semua kontainer basis data yang mendukung standar ODBC. Access juga mendukung teknik-teknik pemrograman berorientasi objek.

### **Kelebihan dan Kekurangan Netbeans**

#### **Kelebihan NetBeans GUI Builder :**

Salah satu yang menjadi kelebihan NetBeans GUI Builder adalah yang telah disebutkan diatas, yaitu GRATIS. Selain itu NetBeans GUI Builder sangat kompetebel dengan Swing karena memang langsung dikembangkan oleh Sun yang notabeneanya sebagai pengembang Swing.

#### **Kekurangan NetBeans GUI Builder :**

NetBeans hanya mensupport 1 pengembangan Java GUI, yaitu Swing, yang padahal ada Java GUI yang dikembangkan oleh eclipse yang bernama SWT dan JFace yang sudah cukup populer.

## **2.2 JAVA**

Java adalah bahasa pemrograman yang disusun oleh James Gosling yang dibantu oleh rekan-rekannya seperti Patrick Naughton, Chris Warth, Ed Rank, dan Mike Sheridan di suatu perusahaan perangkat lunak yang bernama Sun

Microsystems pada tahun 1991. Bahasa pemrograman ini mula-mula diinisialisasi dengan nama “Oak”, namun pada tahun 1995 diganti namanya menjadi “Java”.

Secara arsitektur, Java tidak berubah sedikit pun semenjak awal mula bahasa tersebut dirilis. Kompiler Java (yang disebut dengan **javac** atau *Java Compiler*) akan mentransformasikan kode-kode dalam bahasa Java ke dalam suatu *bytecode*. *Bytecode* adalah sekumpulan perintah hasil kompilasi yang kemudian dapat dieksekusi melalui sebuah mesin komputer abstrak, yang disebut dengan JVM (*Java Virtual Machine*). JVM juga sering dinamakan sebagai *interpreter*, karena sifatnya yang selalu menerjemahkan kode-kode yang tersimpan dalam *bytecode* dengan cara baris demi baris.

### 2.3 UML(*Unified Modelling Language*)

*Unified Modelling Language* (UML) adalah sebuah bahasa untuk menentukan, visualisasi, konstruksi, dan mendokumentasikan *artifacts* dari sistem *software*, untuk memodelkan bisnis, dan sistem *nonsoftware* lainnya. *Artifacts* adalah sepotong informasi yang digunakan atau dihasilkan dalam suatu proses rekayasa *software*. *Artifacts* dapat berupa model, deskripsi, atau *software*. Untuk membuat suatu model, UML memiliki diagram grafis yang diberi nama berdasarkan sudut pandang yang berbeda-beda terhadap sistem dalam proses analisa atau rekayasa. Diagram grafis tersebut antara lain :

1. Use case diagram
2. Class diagram
3. Behavior diagram :
  - a. Statechart diagram
  - b. Activity diagram
  - c. Interaction diagram :
4. Sequence diagram

5. Collaboration diagram
6. Implementation diagram :
  - a. Component diagram
  - b. Deployment diagram

### ***Use Case Diagram***

*Use case diagram* menjelaskan manfaat sistem jika dilihat menurut pandangan orang yang berada diluar sistem (*actor*). Diagram ini menunjukkan fungsionalitas suatu sistem atau kelas dan bagaimana sistem berinteraksi dengan dunia luar.

### ***Class Diagram***

*Class diagram* membantu dalam visualisasi struktur kelas-kelas dari suatu sistem dan merupakan tipe diagram yang paling banyak dipakai. *Class diagram* memperlihatkan hubungan antarkelas dan penjelasan detail tiap-tiap kelas didalam model desain dari suatu sistem . Selama proses analisis, *class diagram* memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem. Selama tahap desain, *class diagram* berperan dalam menangkap struktur dari semua kelas yang membentuk arsitektur sistem yang dibuat.

### ***Sequence Diagram***

*Sequence diagram* menjelaskan interaksi objek yang disusun dalam suatu urutan waktu. Diagram ini secara khusus berasosiasi dengan *use case*. *Sequence diagram* memperlihatkan tahap demi tahap apa yang seharusnya terjadi untuk menghasilkan sesuatu didalam *use case*.

## 2.4 Point Of Sale

Istilah "**Point Of Sale (POS)** dipakai untuk menyebut mesin kasir (Cash Register). Dalam perkembangannya, istilah **Point Of Sale** ini kurang tepat jika diasosiasikan dengan **Cash Register**, karena Cash Register sendiri sebenarnya

adalah kalkulator yang diberi laci otomatis dan pencetak struk (invoice/faktur).  
Selengkapnya Cash Register tidak punya nilai tambah lagi.

# BAB III

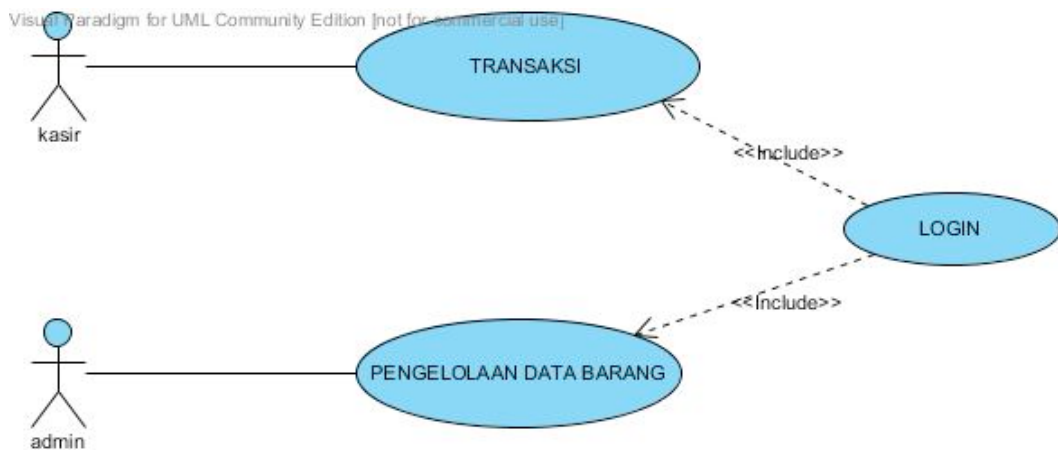
## ANALISIS DAN PERANCANGAN

### 3.1 DESKRIPSI UMUM

Aplikasi minimarket adalah suatu program kasir komputer yang digunakan untuk kegiatan usaha di bidang penjualan. Fasilitas aplikasi minimarket mencakup berbagai menu diantaranya adalah menu database barang dan transaksi kasir.

Aplikasi minimarket hanya bisa digunakan oleh 2 user yaitu admin dan kasir. aplikasi ini hanya dapat digunakan dalam 1 komputer

### 3.2 USE CASE DIAGRAM



### 3.3 SKENARIO USE CASE

Skenario use case

Nama use case : Login

Actor : kasir,admin

Tujuan : verifikasi aplikasi

Deskripsi : kasir,admin melakukan verifikasi login kedalam aplikasi untuk masuk ke dalam aplikasi dan aplikasi meminta nama user dan password user.

login

Pegawai	System
1.Klik Login	
2.Menginputkan user dan password	
	3.Eksekusi validasi use dan password pada database
	4.Jika tidak sesuai, tampil notif login gagal dan kembali ke 1
	5.Jika sesuai, tampil notif login berhasil dan lanjut ke 6
	6. Menampilkan halaman utama program
	7.berhasil

Nama use case : Transaksi

Actor : kasir

Tujuan : transaksi pembelian barang

Deskripsi : Aplikasi meminta kode barang,kasir akan menginputkan kode barang kedalam form transaksi,aplikasi meminta jumlah barang dan system akan menampilkan total harga yang harus di bayar

Kasir	System
	1.Meminta kode barang
2.kasir menginputkan kode barang	3.system check kode barang pada database
	4.status harga
	5.jika sesuai di tampilkan di layar monitor
	6.berhasil

Nama use case : Pengelolaan data barang

Actor : Admin

Tujuan : Pengelolaan data barang

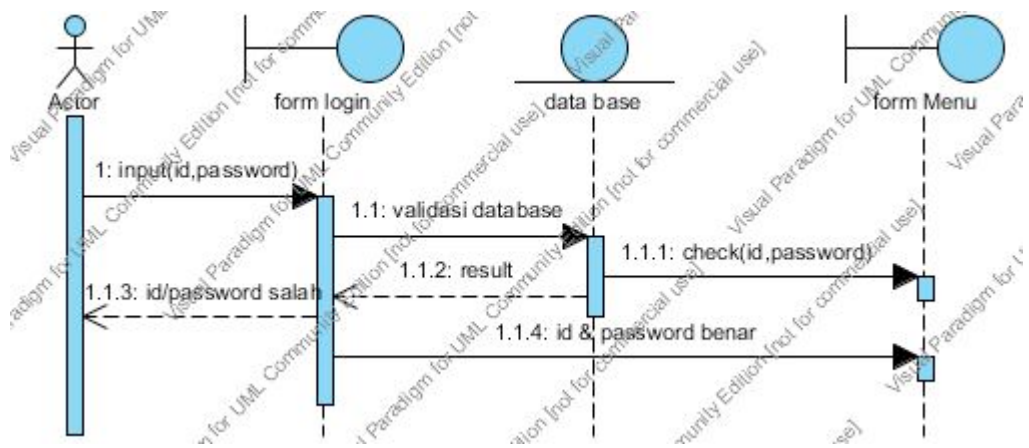
Deskripsi : Admin menginputkan kode,harga,jumlah barang,system akan menyimpan ke dalam database.

Admin	System
1.Admin menginputkan Kode barang,harga barang,jumlah barang.	
2.admin menekan tombol save	3.system akan menyimpan ke dalam database

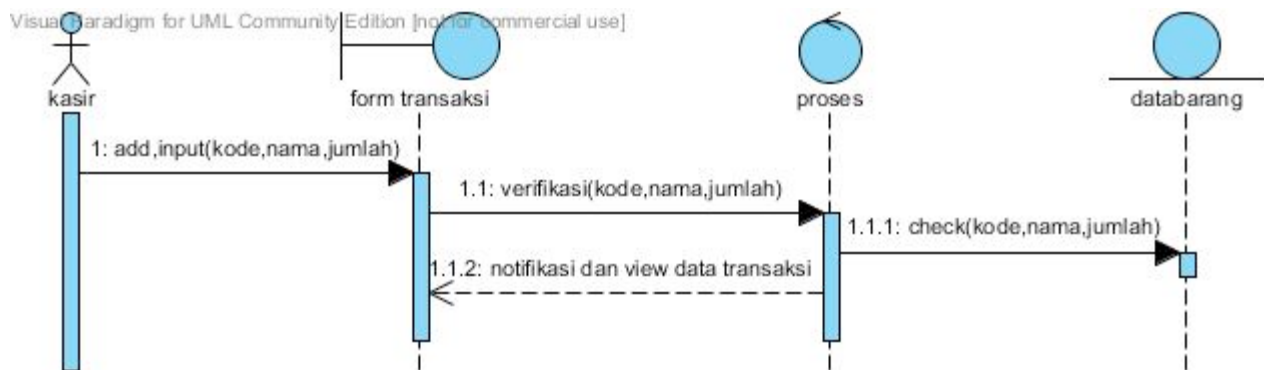
### 3.4 SEQUENCE DIAGRAM

#### 3.4.1 SEQUENCE DIAGRAM LOGIN

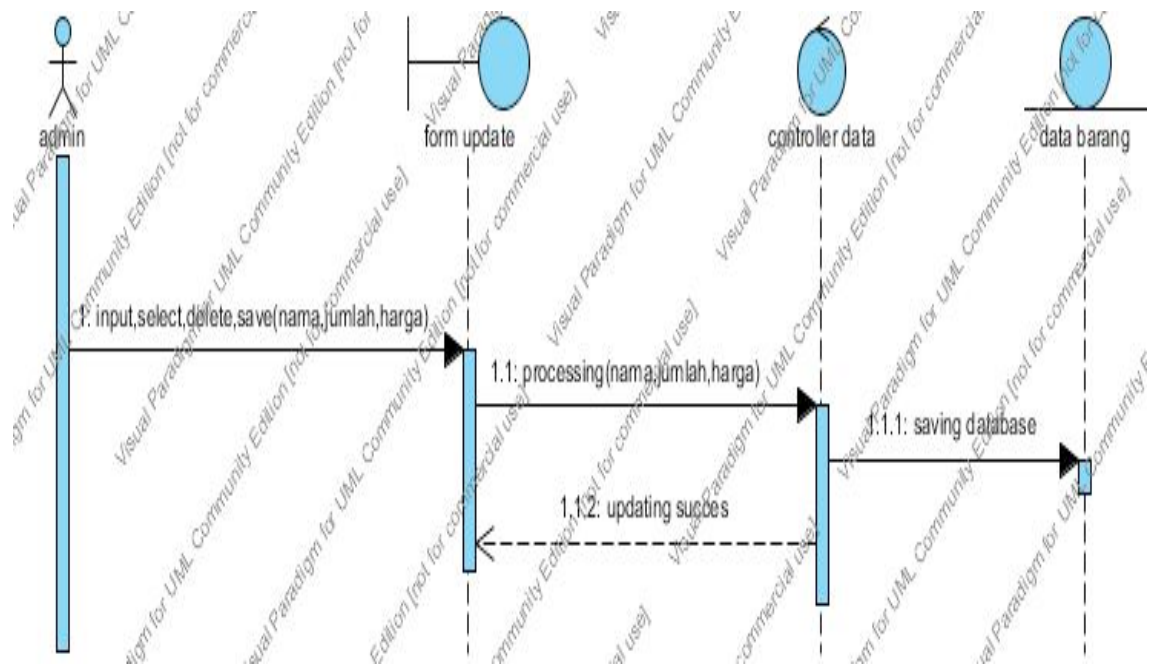




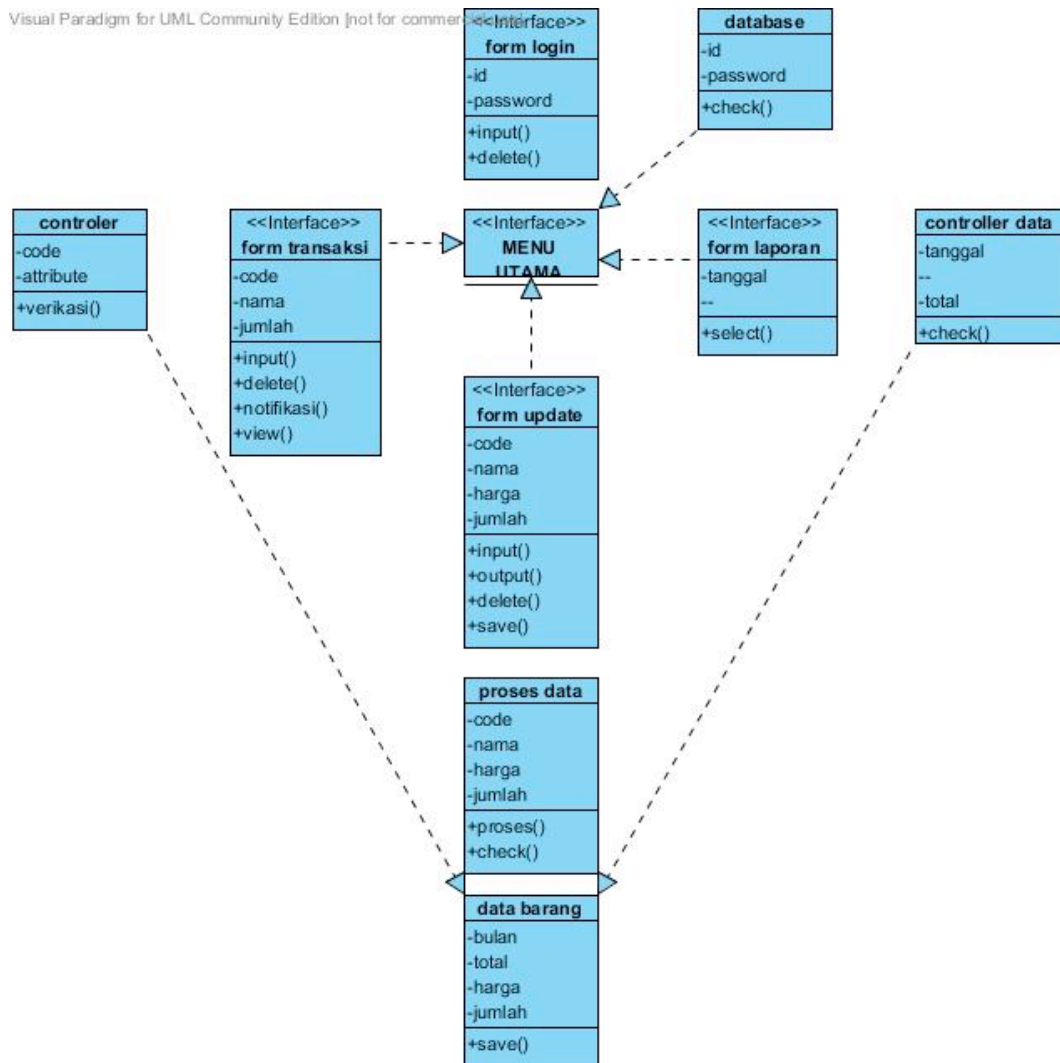
### 3.4.2 SEQUENCE DIAGRAM TRANSAKSI



### 3.4.3 SEQUENCE DIAGRAM PENGELOLAAN DATA BARANG



### 3.4.6 CLASS DIAGRAM



## BAB IV PEMBAHASAN

### 4.1 INTERFACE LOGIN



Pada interface login ini,system akan meminta username dan password untuk masuk ke menu utama.

Source code login

```
public class Login {  
    private String username;  
    private String password;  
  
    public String getPassword() {  
        return password;  
    }  
  
    public void setPassword(String password) {  
        this.password = password;  
    }  
  
    public String getUsername() {  
        return username;  
    }  
}
```

```
public void setUsername(String username) {  
    this.username = username;  
}  
}
```

## 4.2 IMPLEMENTASI PENGELOLAAN DATA BARANG

**Input Data Barang**

Nama

Harga Beli

Harga Jual

Jumlah

Nama	Beli	Jual	Jumlah
Aqua Botol 500ml	2.000	2.500	47
Aqua Gelas	400	500	47
clear	4.000	4.500	3
cocolatos	10.000	10.500	6
dua kelinci	4.250	4.500	16
emeron	2.000	2.500	14
kacang garuda	4.250	4.500	30
Kopi ABC	800	1.000	32
Kopi Kapal Api	800	1.000	52
luk	2.000	2.300	9
pentin	5.000	6.000	48
Pepsodent	2.500	3.000	50
polo mild	5.600	6.000	5
pop mie	2.500	3.000	11

Pada interface ini admin mengelola barang yang masuk dan keluar, barang yang di inputkan admin akan tersimpan dalam database system.

Source code input barang

```

@Override
public List<minimarket.pos.entity.Barang> selectAll() {
    List<minimarket.pos.entity.Barang> lb = new
ArrayList<minimarket.pos.entity.Barang>();
    Connection con = new DataConnection().buatKoneksi();
    PreparedStatement ps;
    try {
        ps = con.prepareStatement(qSelectAll);
        ResultSet rs = ps.executeQuery();
        while (rs.next()) {
            minimarket.pos.entity.Barang b = new windu.pos.entity.Barang();
            b.setId(rs.getInt(1));
            b.setNama(rs.getString(2));
            b.setHargaBeli(rs.getDouble(4));
            b.setHargaJual(rs.getDouble(5));
            b.setJumlah(rs.getInt(3));
            lb.add(b);
        }
    } catch (SQLException ex) {
        Logger.getLogger(Barang.class.getName()).log(Level.SEVERE, null, ex);
    }
    return lb;
}

```

```

@Override
public minimarket.pos.entity.Barang selectById(int id) {
    Connection con = new DataConnection().buatKoneksi();
    ResultSet rs = null;
    minimarket.pos.entity.Barang barang = new minimarket.pos.entity.Barang();
    try {
        PreparedStatement ps = con.prepareStatement(qSelectById);
    }
}

```

```

ps.setInt(1, id);
rs = ps.executeQuery();
rs.next();
String nama = rs.getString(2);
int jumlah = rs.getInt(3);
double hargaBeli = rs.getDouble(4);
double hargaJual = rs.getDouble(5);

barang.setId(id);
barang.setNama(nama);
barang.setJumlah(jumlah);
barang.setHargaBeli(hargaBeli);
barang.setHargaJual(hargaJual);
} catch (SQLException ex) {
    Logger.getLogger(Barang.class.getName()).log(Level.SEVERE, null, ex);
}
return barang;
}

```

@Override

```

public int add(minimarket.pos.entity.Barang barang) {
    Connection con = new DataConnection().buatKoneksi();
    ResultSet rs = null;
    int key = 0;
    try {
        PreparedStatement ps = con.prepareStatement(qAdd,
Statement.RETURN_GENERATED_KEYS);
        ps.setString(1, barang.getNama());
        ps.setInt(2, barang.getJumlah());
        ps.setDouble(3, barang.getHargaBeli());
        ps.setDouble(4, barang.getHargaJual());

```



```

        ps.executeUpdate();
        rs = ps.getGeneratedKeys();
        rs.next();
        key = rs.getInt(1);
    } catch (SQLException ex) {
        Logger.getLogger(Barang.class.getName()).log(Level.SEVERE, null, ex);
    }
    return key;
}

```

@Override

```

public void update(minimarket.pos.entity.Barang barang) {
    Connection con = new DataConnection().buatKoneksi();
    try {
        PreparedStatement ps = con.prepareStatement(qUpdate);
        ps.setString(1, barang.getNama());
        ps.setInt(2, barang.getJumlah());
        ps.setDouble(3, barang.getHargaBeli());
        ps.setDouble(4, barang.getHargaJual());
        ps.setInt(5, barang.getId());
        ps.executeUpdate();
    } catch (SQLException ex) {
        Logger.getLogger(Barang.class.getName()).log(Level.SEVERE, null, ex);
    }
}

```

```

public void updateByBatch(List<windu.pos.entity.Barang> listBarang){
    Connection con = new DataConnection().buatKoneksi();
    PreparedStatement ps;
    try {

```

```

ps = con.prepareStatement(qUpdate);
for (int i = 0; i < listBarang.size(); i++) {
    ps.setString(1, listBarang.get(i).getNama());
    ps.setInt(2, listBarang.get(i).getJumlah());
    ps.setDouble(3, listBarang.get(i).getHargaBeli());
    ps.setDouble(4, listBarang.get(i).getHargaJual());
    ps.setInt(5, listBarang.get(i).getId());
    ps.addBatch();
}
ps.executeBatch();
} catch (SQLException ex) {
    Logger.getLogger(Barang.class.getName()).log(Level.SEVERE, null, ex);
}
}

```

@Override

```

public void delete(int id) {
    throw new UnsupportedOperationException("Not supported yet.");
}

```

@Override

```

public List<minimarket.pos.entity.Barang> selectByPartNama(String nama) {
    List<minimarket.pos.entity.Barang> lb = new
ArrayList<minimarket.pos.entity.Barang>();
    Connection con = new DataConnection().buatKoneksi();
    PreparedStatement ps;
    try {
        ps = con.prepareStatement(qSelectPart);
        ps.setString(1, nama+"%");
        ResultSet rs = ps.executeQuery();
        while (rs.next()) {

```

```
        minimarket.pos.entity.Barang b = new minimarket.pos.entity.Barang();
        b.setId(rs.getInt(1));
        b.setNama(rs.getString(2));
        b.setHargaBeli(rs.getDouble(4));
        b.setHargaJual(rs.getDouble(5));
        b.setJumlah(rs.getInt(3));
        lb.add(b);
    }
} catch (SQLException ex) {
    Logger.getLogger(Barang.class.getName()).log(Level.SEVERE, null, ex);
}
return lb;
}
}
```

#### 4.3 INTERFACE TRANSAKSI

Rp 17.000		Rp 3.000		
Nama	Harga	Jumlah	Diskon	Total
Kopi Kapal Api	1.000	3		3.000
<small>Pilih Barang [F1] ; Pembayaran[F2]</small>		Total Pembelian	Pembayaran	Kembalian
		Rp 3.000	Rp 20.000	Rp 17.000

Pada interface ini kasir menginputkan nama barang, jumlah barang, sistem akan memproses transaksi kasir.

Source kode transaksi

/\*\*

```

*
* @author
*/
public class PenjualanBarang implements IPenjualanBarang{
    private String qSelect = "SELECT * FROM penjualanbrg";
    private String qAdd = "INSERT INTO penjualanbrg (IdPenjualan, IdBarang,
    Jumlah, Diskon) VALUES (?, ?, ?, ?)";

    @Override
    public List<penjualan.pos.entity.PenjualanBarang> selectAll() {
        throw new UnsupportedOperationException("Not supported yet.");
    }

    @Override
    public minimarket.pos.entity.PenjualanBarang selectById(int id) {
        throw new UnsupportedOperationException("Not supported yet.");
    }

    @Override
    public int add(minimarket.pos.entity.PenjualanBarang penjualanBarang) {
        Connection con = new DataConnection().buatKoneksi();
        ResultSet rs = null;
        int key = 0;
        try {
            PreparedStatement ps = con.prepareStatement(qAdd);
            ps.setInt(1, penjualanBarang.getPenjualan().getIdPenjualan());
            ps.setInt(2, penjualanBarang.getBarang().getId());
            ps.setInt(3, penjualanBarang.getJumlah());
            ps.setDouble(4, penjualanBarang.getDiskon());
            ps.executeUpdate();
            rs = ps.getGeneratedKeys();

```

```

        rs.next();
        key = rs.getInt(1);
    } catch (SQLException ex) {
        Logger.getLogger(PenjualanBarang.class.getName()).log(Level.SEVERE,
null, ex);
    }
    return key;
}

```

```

@Override
public void addByBatch(List<minimarket.pos.entity.PenjualanBarang>
listPenjBrg) {
    Connection con = new DataConnection().buatKoneksi();
    try {
        PreparedStatement ps = con.prepareStatement(qAdd);
        for (int i = 0; i < listPenjBrg.size(); i++) {
            ps.setInt(1, listPenjBrg.get(i).getPenjualan().getIdPenjualan());
            ps.setInt(2, listPenjBrg.get(i).getBarang().getId());
            ps.setInt(3, listPenjBrg.get(i).getJumlah());
            ps.setDouble(4, listPenjBrg.get(i).getDiskon());
            ps.addBatch();
        }
        ps.executeBatch();
    } catch (SQLException ex) {
        Logger.getLogger(PenjualanBarang.class.getName()).log(Level.SEVERE,
null, ex);
    }
}

```

```

@Override
public void update(minimarket.pos.entity.PenjualanBarang penjualanBarang) {

```

```
        throw new UnsupportedOperationException("Not supported yet.");
    }

    @Override
    public void delete(int id) {
        throw new UnsupportedOperationException("Not supported yet.");
    }
}
```

## BAB V KESIMPULAN DAN SARAN

### 5.1 Kesimpulan

- Aplikasi ini mempermudah proses transaksi kasir dengan pelanggan.
- Aplikasi ini mempermudah proses pengelolaan data barang minimarket.

### 5.2 Saran

- Aplikasi minimarket ini harus tersedia menu laporan
- Aplikasi minimarket ini harus dapat digunakan dalam berbagai OS
- Aplikasi minimarket ini harus memiliki system barcode



## DAFTAR PUSTAKA

### Internet

<https://en.wikipedia.org/wiki/netbeans>

<http://alqaa.wordpress.com/2011/10/14/16>

<http://ilmuti.org/2012/08/cara-mudah-belajar-menggunakan-net-beans-2/>

<http://www.slideshare.net/bayuvins/koneksi-database-mysql-dengan-netbeans>

<http://blog.politeknitelkom.ac.id/30210045/2012/12/08/cara-mudah-belajar-menggunakan-netbeans/>

<http://kitacomunity.blogspot.com/2010/05/sejarah-netbeans.html>

<http://dwiendahnihayati.blogspot.com/2013/02/sejarah-netbeans.html>

### Buku

Andi. 1995. Pengenalan java netbeans. Bandung: Gramedia

Hirin. A. M. database mysql. Jakarta: Penerbit Andi