

## Sistem Persediaan Barang Pada UKM Deviande (*Meat Shop*) Bekasi

Hendra Jaya Putra<sup>1</sup>, Anna Mukhayaroh<sup>2</sup>

STMIK Nusa Mandiri

xiathim@gmail.com<sup>1</sup>, anna.auh@nusamandiri.ac.id<sup>2</sup>

**Abstrak** - Sistem persediaan barang yang masih menggunakan cara manual yaitu dengan input data pada buku manual serta pada Microsoft office excel dan informasi kadaluarsa barang masih harus melihat secara aktual sehingga banyak sekali *lost dead stock* produk. Dengan adanya permasalahan ini maka penulis bermaksud membuat konsep perancangan sistem persediaan barang, sehingga dari laporan data ketersediaan barang dan data kadaluarsa barang dapat dikontrol melalui aplikasi. Metode pengembangan pada aplikasi menggunakan konsep *waterfall* (air terjun) dengan konsep pemrograman berbasis object. Aplikasi tersebut menggunakan Bahasa pemrograman PHP dengan menggunakan basis data MySQL.

*Keywords: Sistem, persediaan barang, Waterfall*

**Abstract** - *Inventory systems still use the manual method that is by inputting data on manual books as well as on Microsoft office excel and expired information still have to see the actual goods so that a lot of lost dead stock products. With this problem, the authors intend to make the concept of inventory system design, so that the availability of goods data reports and expired data can be controlled through the application. The method of application development uses the waterfall concept with the concept of object-based programming. The application uses the PHP programming language by using a MySQL database.*

*Keywords: System, inventory, Waterfall*

### I. PENDAHULUAN

Salah satu faktor yang penting dalam suatu usaha adalah persediaan barang, dimana barang - barang yang tersedia akan dijual oleh pelaku usaha kepada pelanggan. Laporan persediaan barang sangatlah penting dalam menjalankan suatu usaha, karena dapat dijadikan salah satu acuan perkembangan usaha tersebut. Kesuksesan perusahaan dalam mempertahankan bisnisnya tidak terlepas dari peran sebuah perusahaan dalam mengelola persediaan barang sehingga dapat memenuhi permintaan pelanggan secara maksimal (Jayanti, 2015).

Banyak pelaku usaha yang beranggapan bahwa memanfaatkan teknologi komputer dalam kegiatan stok barang kurang penting, karena kegiatan tersebut dapat dikerjakan secara manual. Sistem persediaan barang yang dilakukan menggunakan cara manual, dimana pencatatan data masih menggunakan kertas menyebabkan kurang terjamin keakuratan data, kemungkinan terjadi kesalahan pencatatan dan perhitungan atas transaksi yang terjadi, serta mengalami kesulitan dalam mencari data yang dibutuhkan dan mengontrol stok barang (Sikumbang, 2016).

Deviande *Meat Shop*, Bekasi adalah sebuah pemasok daging ke beberapa toko dan tempat makan di daerah Jabodetabek. Dalam menjalankan usahanya, Deviande *Meat Shop*, Bekasi mengelola stok daging secara manual. Masa kadaluarsa yang tidak dicek secara berkala sehingga beresiko mengalami kerugian

karena terbuangnya beberapa stok daging yang sudah kadaluarsa.

Selain itu, pelacakan dan pemeriksaan persediaan barang yang lama menyebabkan pelanggan tidak dapat melakukan pemesanan secara langsung ketika menghubungi Deviande *Meat Shop*. Oleh itu perlu dibangunnya sebuah sistem informasi yang dapat membantu proses pengelolaan data persediaan barang yang mencakup pemasukan dan/atau pengeluaran barang serta hingga informasi mengenai masa kadaluarsa barang. Dengan adanya sistem ini pembuatan laporan dapat menghasilkan laporan dengan tepat waktu dan akurat sesuai kebutuhan.

### II. METODOLOGI PENELITIAN

Metode yang digunakan pada pengembangan perangkat lunak ini adalah model air terjun (*Waterfall Model*). Menurut Indriani & Sudarmadi (2015) model *waterfall* terbagi menjadi 5 tahap yaitu:

#### 1. Analisa Kebutuhan Sistem

Proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak agar dapat dipahami user.

#### 2. Desain

Desain adalah proses multi langkah yang fokus pada desain program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengodean.

3. Pembuatan Kode Program  
*Desain* harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat.
4. Pengujian  
Pengujian focus pada perangkat lunak secara dari segi logik dan fungsional, memastikan bahwa semua bagian sudah diuji untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan keinginan.
5. Pendukung  
Tidak menutup kemungkinan sebuah perangkat lunak yang telah dikirimkan kepada user terjadi perubahan. Perubahan tersebut karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru.

Teknik pemrograman yang digunakan di dalam pengembangan perangkat lunak yaitu OOP (*Object Oriented Programming*). Menurut (Abdulloh, 2018:7) “OOP (*Object Oriented Programming*) merupakan teknik pemrograman dengan menggunakan konsep objek. Tujuan dari OOP adalah untuk memudahkan programmer dalam pembuatan program dengan menggunakan konsep objek yang ada dalam kehidupan sehari-hari.”

Beberapa keuntungan membuat program dengan konsep OOP sebagai berikut:

- a Mudah dikelola karena dapat dengan mudah menemukan sumber kesalahan untuk diperbaiki.
- b Setiap objek dapat ditambah kemampuannya tanpa mengganggu objek lain.
- c Setiap objek dapat dipakai dalam proyek yang lain tanpa perlu banyak penyesuaian.

Adapun tools system yang digunakan pada penelitian ini, sebagai berikut :

#### 1. UML (*Unified Modeling Language*)

Menurut (Mulyani, 2016b:35) “UML adalah sebuah teknik pengembangan sistem yang menggunakan bahasa grafis sebagai alat untuk pendokumentasian dan melakukan spesifikasi pada sistem.”

Berikut adalah penjelasan tentang beberapa diagram UML yang akan digunakan dalam perancangan sistem yang dibangun:

##### a *Activity Diagram*

Menurut Sukamto & Shalahuddin dalam (Rizki dkk, 2019) menyatakan bahwa “Diagram aktivitas atau *activity* diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak”.

##### b *Use Case Diagram*

*Use Case* merupakan diagram pertama dalam kelompok UML. *Use Case* menunjukkan urutan tindakan ketika user berinteraksi dengan sistem. Diagram ini didefinisikan sebagai kemampuan dari sistem untuk memberikan user sebuah hasil karena *Use Case* mengendalikan seluruh proses pengembangan perangkat lunak mulai dari desain sampai implementasi secara sistematis dalam komputer (James, 2016:58).

##### c *Class Diagram*

*Class Diagram* digunakan sebagai gambaran objek apa sajakah yang dibutuhkan oleh sistem informasi perusahaan dan menjelaskan pula hubungan antar kelas yang telah ditentukan (Jayanti, 2015).

##### d *Sequence Diagram*

*Sequence Diagram* menggambarkan perilaku aktor dalam sistem dari sudut pandang user. *Sequence Diagram* digunakan untuk menggambarkan interaksi diantara objek tetapi bagian dalam dari sebuah objek tidak ditampilkan (Rumpe, 2016:192).

##### e *Component Diagram*

*Component Diagram* digunakan untuk menggambarkan struktur fisik dari sebuah implementasi macam-macam komponen dalam sistem. Sebuah *Component Diagram* secara khusus digunakan untuk mencapai beberapa hal (Rajib, 2018:345) yaitu :Mengatur *source code* agar dapat membangun keluaran (*releases*) yang dapat dieksekusi;Menggambarkan ketergantungan diantara komponen.

##### f *Deployment Diagram*

*Deployment Diagram* menunjukkan hal yang berhubungan dengan sebuah sistem. Dengan kata lain, *Deployment Diagram* menunjukkan bagaimana sebuah *software system* secara fisik saling berhubungan dengan *hardware*. Diagram ini dapat menggambarkan bagaimana komponen perangkat lunak yang berbeda-beda didistribusikan diantara komponen perangkat keras yang berbeda-beda pada sebuah sistem (Rajib, 2018:345).

##### g *Package Diagram*

*Package Diagram* dapat digunakan untuk mengelompokkan beberapa kelas yang berhubungan. Diagram ini dapat menunjukkan perbedaan kelompok kelas (*package*) dan ketergantungan antar kelas (Rajib, 2018:344).

#### 2. ERD

*Entity Relationship Model (ERM)* adalah sebuah model data yang mendeskripsikan relasi (1:1, 1:M, dan M:N) diantara entitas-entitas sebuah diagram. Diagram yang menggambarkan sebuah hubungan antar entitas, atribut dan relasi dinamakan *Entity Relationship Diagram (ERD)* (Coronel & Morris, 2019:45). Menurut Ladjamudin dalam (Permana, 2018) “*Entity Relation Diagram*

(ERD) adalah suatu model jaringan yang menggunakan susunan data yang disimpan dalam sistem secara abstrak.”

Notasi-notasi simbolik yang digunakan dalam ERD menurut (Muslihudin & Oktafianto, 2016:84-84), sebagai berikut :

a. Entitas

Entitas adalah segala sesuatu yang dapat digambarkan oleh data yang dapat dibedakan dari sesuatu yang lain.

b. Atribut

Atribut merupakan pendeskripsian karakteristik dari entitas.

c. Relasi atau Hubungan

Relasi menunjukkan adanya hubungan antara sejumlah entitas.

d. Penghubung Antara Himpunan Relasi

Penghubung antara himpunan relasi dinyatakan dalam bentuk garis.

3. LRS

Menurut Hasugian & Shidiq dalam (Rizki dkk, 2019) memberikan batasan bahwa “*Logical Record Structured (LRS)* adalah sebuah model sistem yang digambarkan dengan sebuah diagram-ER akan mengikuti pola atau aturan permodelan tertentu dalam kaitannya dengan konvensi ke LRS”.

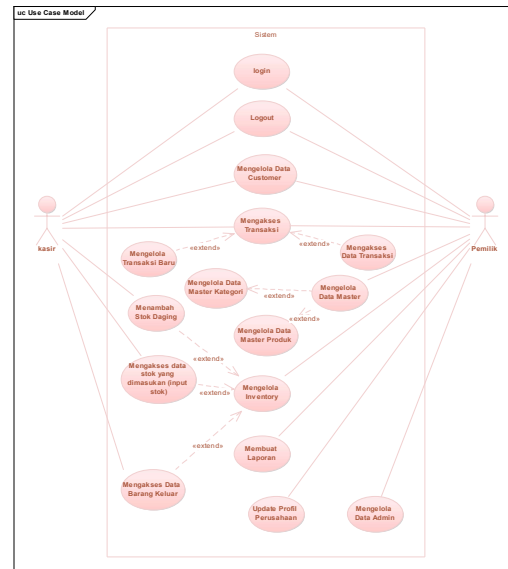
### III. HASIL DAN PEMBAHASAN

#### 3.1 Analisa Kebutuhan Software

Dari analisa prosedur sistem berjalan dapat diperoleh informasi yaitu terdapat dua aktor yang melakukan proses bisnis tersebut yaitu kasir dan juga pemilik toko. Dimana setiap aktor memiliki tanggungjawabnya masing-masing. Dalam pembangunan sistem informasi ini akan menyesuaikan dengan prosedur sistem berjalan. Sehingga sistem ini diharapkan optimal dan sesuai dengan kebutuhan yang ada. Aktor kasir dalam tahap prosesur sistem berjalan, kasir melakukan kegiatan melayani pembeli secara langsung, melayani pembayaran, dan sebagai pegawai toko yang melakukan proses pengemasan daging untuk customer. Aktor pemilik toko dimana sebagai pemilik toko, maka dia dapat mengakses keseluruhan sistem tersebut. Aktor pemilik toko, dapat mengelola data master seperti mengelola data produk daging, mengelola data kasir, melakukan penginputan data *inventory*, dan membuat laporan yang bisa dibuat sesuai dengan periode yang diinginkan.

#### 3.2 Use Case Diagram

Berikut merupakan *usecase diagram* dari sistem persediaan barang.



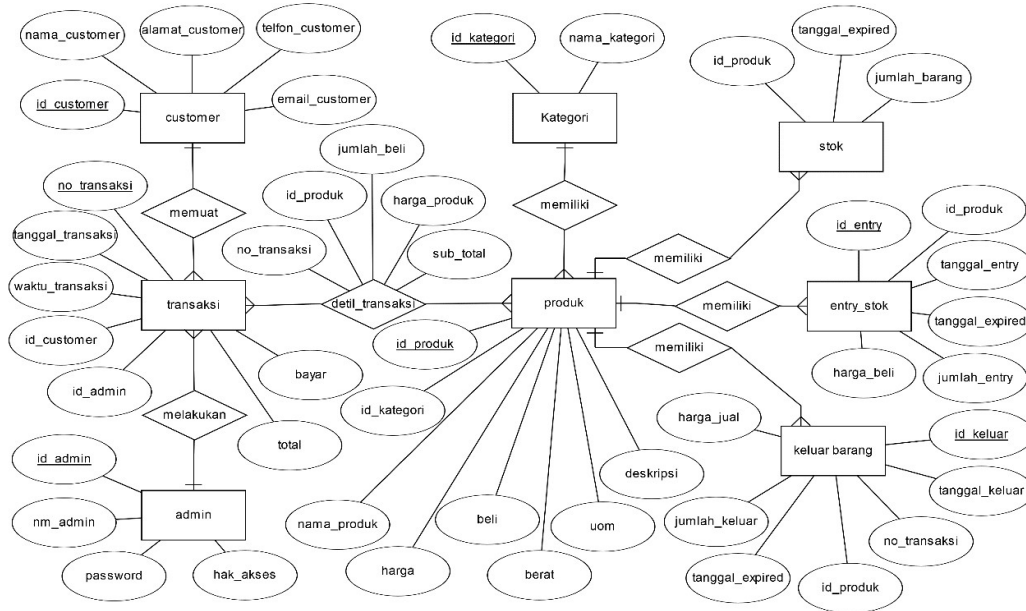
Sumber : Hasil Penelitian (2019)

**Gambar 1 Use Case Diagram Sistem Usulan**

1. Analisa kebutuhan aktor ( kasir ) akan sistem
  - A1 Kasir melakukan login kasir
  - A2 Kasir membuat data transaksi baru
  - A3 Kasir melihat data transaksi
  - A4 Kasir mengelola data customer
  - A5 Pemilik mengelola stock
  - A6 Pemilik mengelola data input stock
  - A7 Pemilik mengelola barang keluar
  - A8 Kasir melakukan logout
  
2. Analisa kebutuhan aktor (pemilik toko) akan sistem
  - B1 Pemilik melakukan login pemilik
  - B2 Pemilik mengelola data master kategori
  - B3 Pemilik mengelola data master produk daging
  - B4 Pemilik membuat data transaksi baru
  - B5 Pemilik melihat data transaksi
  - B6 Pemilik mengelola stock
  - B7 Pemilik mengelola data input stock
  - B8 Pemilik mengelola barang keluar
  - B9 Pemilik membuat laporan
  - B10 Pemilik mengupdate profil perusahaan
  - B11 Pemilik mengelola data customer
  - B12 Pemilik mengelola data admin
  - B13 Pemilik melakukan logout



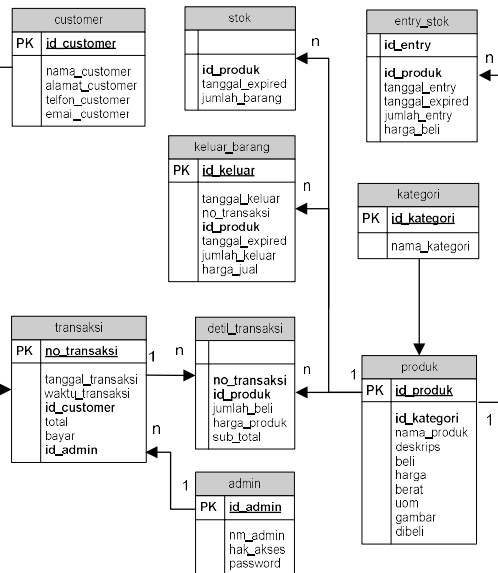
3.4 ERD



Sumber : Hasil Penelitian (2019)

Gambar 5  
ERD

3.5 LRS

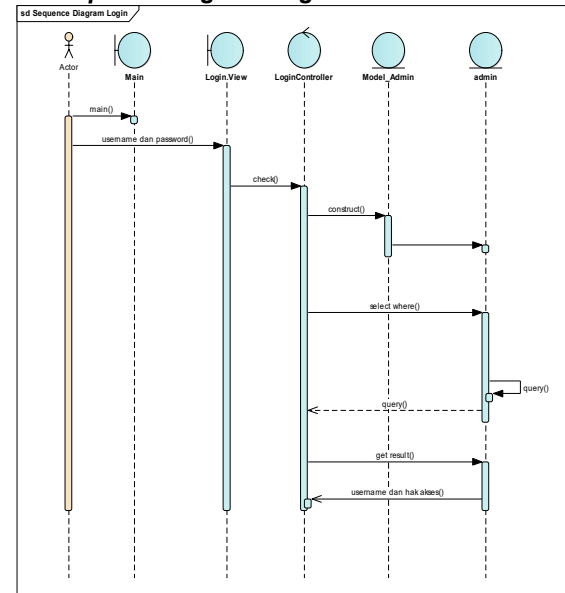


Sumber : Hasil Penelitian (2019)

Gambar 6  
LRS

3.6 Sequence Diagram

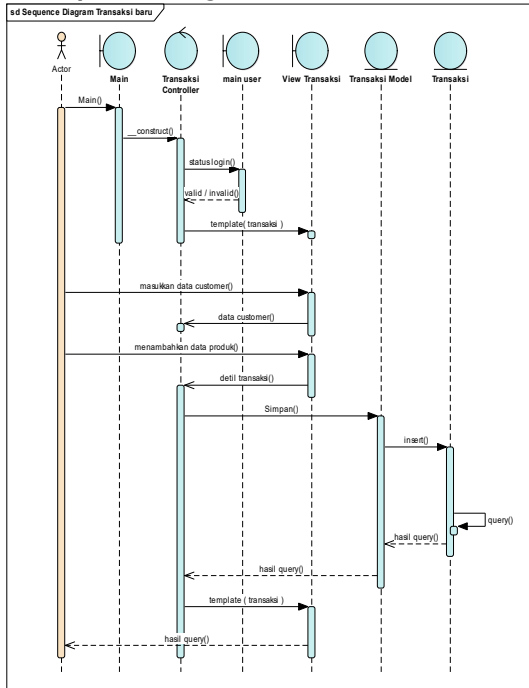
1. Sequence Diagram Login



Sumber : Hasil Penelitian (2019)

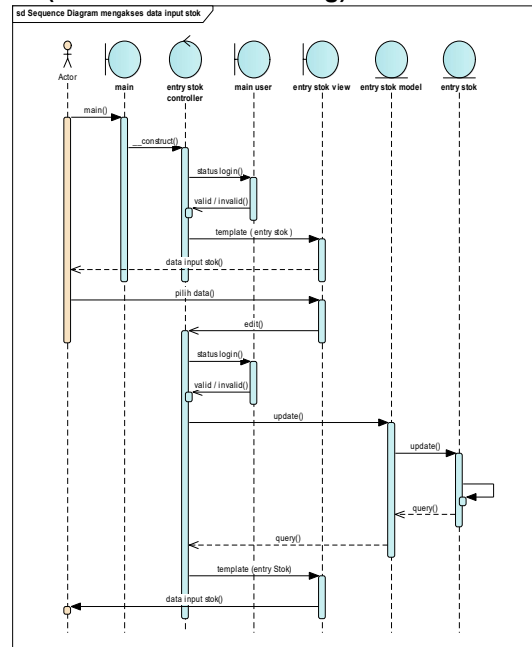
Gambar 7  
Sequence Diagram Login

2. Sequence Diagram Transaksi Baru



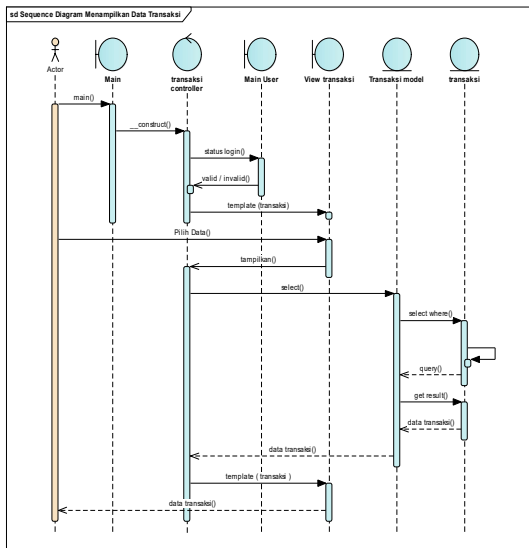
Sumber : Hasil Penelitian (2019)  
**Gambar 8**  
 Sequence Diagram Transaksi Baru

4. Sequence Diagram Input Stok (Menambah Stok Barang)



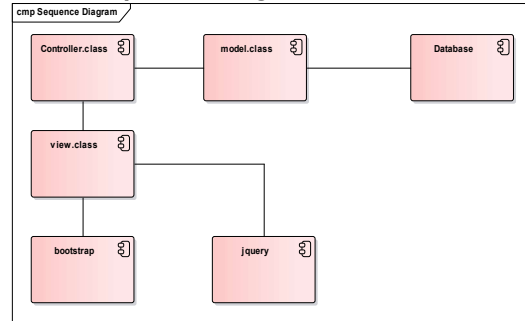
Sumber : Hasil Penelitian (2019)  
**Gambar 10**  
 Sequence Diagram Input Stok (Menambah Stok Barang)

3. Sequence Diagram Mengakses Data Transaksi



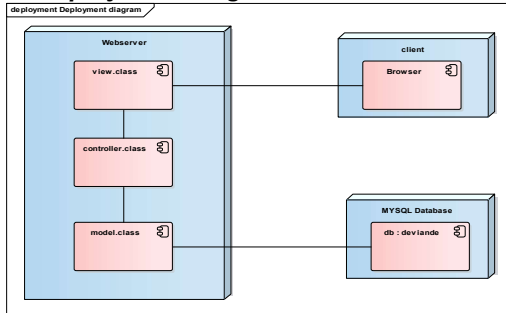
Sumber : Hasil Penelitian (2019)  
**Gambar 9**  
 Sequence Diagram Mengakses Data Transaksi

5. Component Diagram



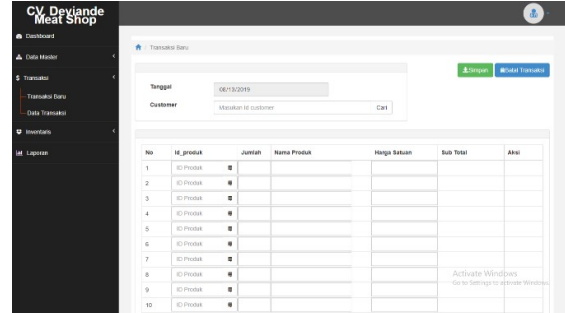
Sumber : Hasil Penelitian (2019)  
**Gambar 11**  
 Component Diagram

**6. Deployment Diagram**



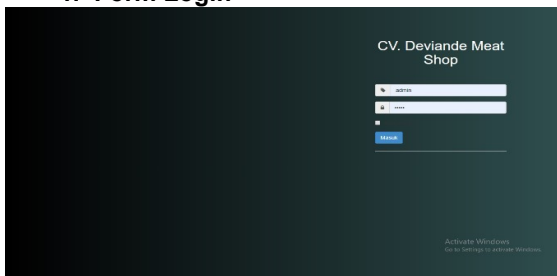
Sumber : Hasil Penelitian (2019)  
**Gambar 12**  
 Deployment Diagram

**3. Form Transaksi Baru**



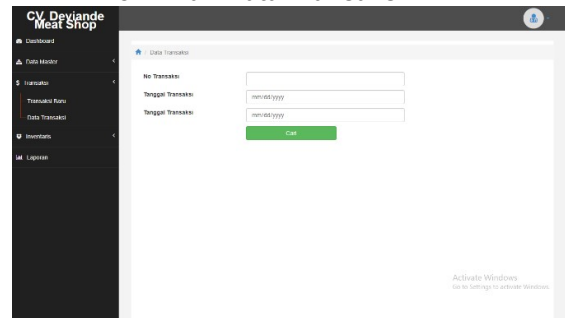
Sumber : Hasil Penelitian (2019)  
**Gambar 15**  
 Tampilan Form Transaksi Baru

**3.7 Hasil Tampilan Antarmuka**  
**1. Form Login**



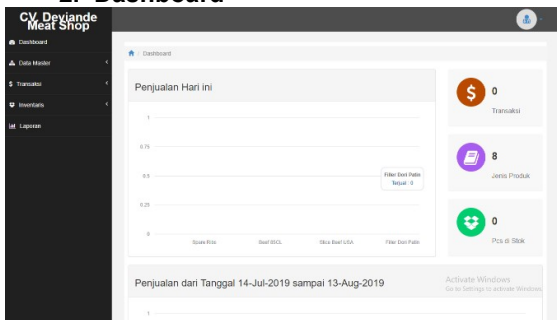
Sumber : Hasil Penelitian (2019)  
**Gambar 13**  
 Tampilan Form Login

**4. Form Cari Data Transaksi**



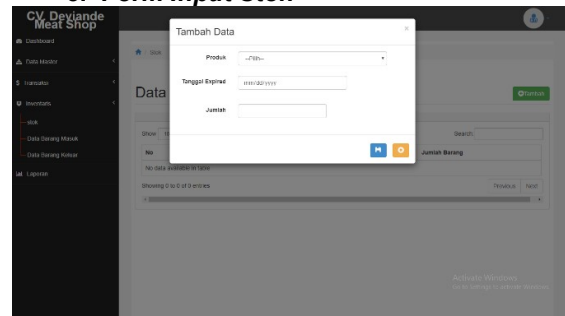
Sumber : Hasil Penelitian (2019)  
**Gambar 16**  
 Tampilan Form Cari Data Transaksi

**2. Dashboard**



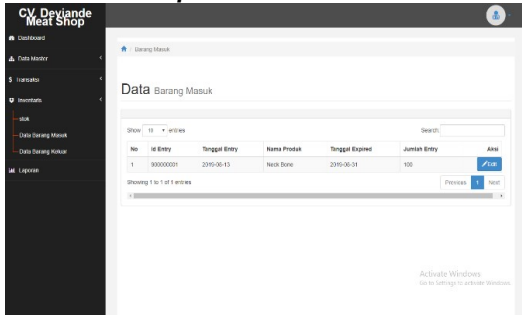
Sumber : Hasil Penelitian (2019)  
**Gambar 14**  
 Tampilan Dashboard

**5. Form Input Stok**



Sumber : Hasil Penelitian (2019)  
**Gambar 17**  
 Tampilan Form Input Stok

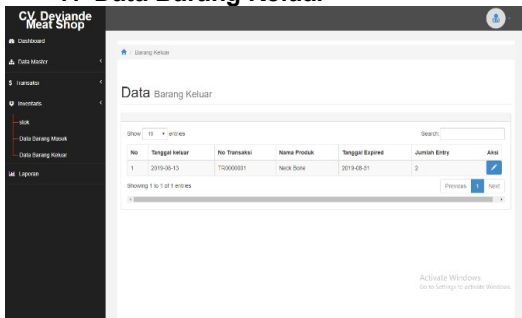
## 6. Data Input Stok



Sumber : Hasil Penelitian (2019)

**Gambar 18**  
Tampilan Data Input Stok

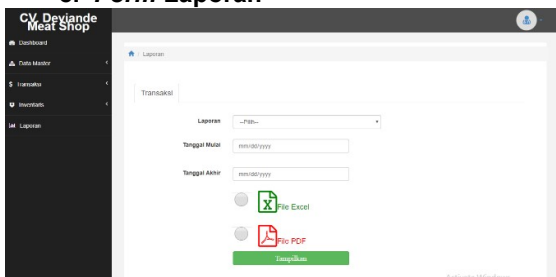
## 7. Data Barang Keluar



Sumber : Hasil Penelitian (2019)

**Gambar 19**  
Tampilan Barang Keluar

## 8. Form Laporan



Sumber : Hasil Penelitian (2019)

**Gambar 20**  
Tampilan Form Laporan

## IV. KESIMPULAN

Berdasarkan perancangan sistem yang dibangun atas permasalahan yang telah dibahas sebelumnya, maka dapat disimpulkan: Melalui aplikasi persediaan barang ini, maka akurasi persediaan barang sudah bisa dapat dipertanggung jawabkan. Penggunaan dengan aplikasi ini, dapat menghindari kelalaian yang mengakibatkan persediaan barang lewat masa aktif nya sehingga tidak layak di jual. Melalui aplikasi ini sangat memudahkan dalam laporan persediaan barang, baik pemasukkan barang dan pengeluaran barang.

## V. REFERENSI

- Abdulloh, Rohi. 2018. "7 in 1 Pemrograman Web Tingkat Lanjut." *PT Elex Media Komputindo*. Retrieved July 25, 2019.
- Coronel, Carlos and Steven Morris. 2019. "*Database Systems\_Design, Implementation, & Management*". Cengage Learning. Retrieved Januari 1, 2018.
- Jayanti, Ni Ketut. 2015. "Perancangan Sistem Informasi Persediaan Barang Menggunakan Model Reorder Point." (<https://eksplora.stikom-bali.ac.id/index.php/eksplora/article/view/78/61>)
- Karlana, Indriani and Sudarmadi. 2015. "Sistem Informasi Inventory Alat Tulis Kantor (Atk) Menggunakan Metode Waterfall (Studi Kasus : Otoritas Jasa Keuangan (OJK))." *Jurnal Techno Nusa Mandiri* 12(1):69–76. (<http://ejournal.nusamandiri.ac.id/index.php/techo/article/view/499>)
- Mulyani, Sri. 2016b. "Sistem Informasi Manajemen Rumah Sakit: Analisis Dan Perancangan." *Abdi Sistemika*. Retrieved July 15, 2019.
- Muslihudin, Muhamad and Oktafianto. 2016. "Analisis Dan Perancangan Sistem Informasi Menggunakan Model Terstruktur Dan UML." *CV Andi Offset*. Retrieved July 23, 2019.
- Rizki, Annisa, Anna Mukhayaroh, and Erene Gernaria Sihomning. 2019. "Sistem Informasi Persediaan Barang Pada Toko Mainan Nanda Toys Bekasi." *Teknologi Informatika & Komputer* 5(1):62–70. (<http://www.ejournal-umht.org/index.php/tik/article/view/199>)
- Rumpe, Bernhard. 2016. "Modeling with UML: Language, Concepts, Methods." *Springer*. Retrieved July 23, 2019.
- RAJIB, MALL. 2018. "FUNDAMENTALS OF SOFTWARE ENGINEERING, FIFTH EDITION." *Asoke K. Ghosh, PHI Learning Private Limited*. Retrieved July 23, 2019
- Sikumbang, E. D. (2016). Sistem Informasi Persediaan Barang Berbasis Website Menggunakan Metode Waterfall Dengan Konsep Pemograman Tersrtuktur. *Snipstek*, 139–143. (<https://konferensi.nusamandiri.ac.id/proceeding/index.php/snipstek/article/view/82>)