

## APLIKASI POSITORY MULTI OUTLET MENGGUNAKAN WEB SERVICE PADA JAYA ELEKTRONIK DI PURWOKERTO

Abdul Azis<sup>1)</sup>, Riyanto<sup>2)</sup>

<sup>1,2)</sup> *Sisten Informasi UNIVERSITAS AMIKOM PURWOKERTO*  
email : *abdazis9@amikompurwokerto.ac.id<sup>1)</sup>, riyanto@amikompurwokerto.ac.id<sup>2)</sup>*

### Abstraksi

Jaya Elektronik merupakan suatu usaha yang bergerak dalam bidang retail barang elektronik dan *handphone*. Jaya Elektronik memiliki dua cabang dan satu toko pusat, pada masing – masing toko dibagi menjadi dua bagian yaitu bagian penjualan dan bagian gudang, sistem transaksi yang sudah berjalan pada toko Jaya Elektronik masih secara konvensional dan belum terkomputerisasi. Dari sistem yang sudah berjalan selama ini, pemilik toko Jaya Elektronik mengalami permasalahan terkait pelaporan transaksi yang terjadi pada ketiga toko dan gudang Jaya Elektronik. Aplikasi Pository Multi Outlet menggunakan *web service* pada toko Jaya Elektronik merupakan sebuah aplikasi yang dirancang dan dibangun untuk mengelola data barang, data pemasok, data karyawan, proses transaksi penjualan, dan pembuatan laporan. Aplikasi ini dibangun menggunakan metode pengembangan *SDLC Waterfall*. Penelitian ini bertujuan untuk membangun aplikasi Pository Multi Outlet untuk membantu pemilik toko dan karyawan dalam melakukan pelayanan transaksi penjualan, pengelolaan data barang pada gudang dan pengontrolan semua transaksi yang terjadi pada tiga toko Jaya Elektronik. Penelitian ini menghasilkan aplikasi Pository Multi Outlet yang terintegrasi dengan tiga toko untuk menyelesaikan permasalahan terkait proses pelaporan transaksi yang terjadi pada masing – masing toko dan gudang Jaya Elektronik.

### Kata Kunci :

Jaya Elektronik, *Web Service*, *SDLC Waterfall*

### Abstract

*Jaya Electronic is a business that is engaged in the retail of electronic goods and cellphones. Jaya Electronic has two branches and one centralship store, each one store is divided into two section, namely the sales and warehouse section, the transaction system that has been running at the Jaya Electronic store is still conventional and not computerized. From the system that has been running so far, the owner of the Jaya Electronic store is responsible for transactions carried out at the Jaya Electronic store and warehouse. The Pository Multi Outlet application using web services at the Jaya Electronic store is an application designed and built to manage goods data, data suppliers, employee data, sales transaction processes, and report generation. This application was built using the method of developing SDLC Waterfall. This study aims to build a Pository Multi Outlet application to help store owners and employees in conducting sales transactions, managing goods data in warehouses and controlling all transactions carried out in three Jaya Electronic stores. This study resulted in an integrated Multi-Outlet Information application with three stores to complete related to the reporting process that took place in each of the store and warehouses of Jaya Electronic..*

### Keywords :

*Jaya Electronic, Web Services, SDLC Waterfall*

### Pendahuluan

Jaya Elektronik merupakan suatu usaha yang bergerak dalam bidang retail barang elektronik dan *handphone*. Toko Jaya Elektronik dikelola oleh pemilik toko dan dibantu oleh beberapa karyawan. Pada masing – masing toko dibuat menjadi dua bagian yaitu bagian penjualan dan bagian gudang. Pada bagian penjualan bertugas melayani konsumen yang datang ke toko untuk membeli barang – barang elektronik dan *handphone*, sedangkan bagian gudang bertugas mengontrol barang masuk pada gudang toko Jaya Elektronik dan juga mengontrol persediaan barang yang ada pada gudang.

Toko Jaya Elektronik memiliki dua cabang dan satu toko pusat. Layanan transaksi penjualan dan

pencatatan barang pada toko dan gudang Jaya Elektronik berjalan secara konvensional dan belum terkomputerisasi, dari layanan transaksi tersebut barang yang dipesan oleh pemilik toko kepada pemasok masuk ke dalam gudang terlebih dahulu sebelum dijual pada toko Jaya Elektronik, pada saat barang masuk ke gudang Jaya Elektronik barang dicatat satu per satu oleh karyawan gudang, setiap pembelian barang kepada pemasok nota pembelian disimpan sebagai bukti transaksi pembelian barang dan juga digunakan sebagai bahan pembuatan laporan barang masuk pada gudang, setelah barang masuk ke gudang kemudian barang dipindahkan ke toko untuk dijual. Kemudian setelah barang masuk ke dalam toko berarti barang siap untuk dijual, setiap barang yang terjual juga harus memiliki bukti

transaksi penjualan berupa nota, nota penjualan tersebut digunakan sebagai bahan pembuatan laporan penjualan oleh pemilik toko.

Sistem penjualan pada toko Jaya Elektronik dan pencatatan data barang masuk pada gudang yang sudah berjalan hanya menghasilkan data fisik sebagai bukti transaksi, data fisik tersebut nantinya digunakan sebagai bahan acuan pembuatan laporan harian dan bulanan dari ketiga toko. Pada toko Jaya Elektronik pembuatan laporan dilakukan oleh pemilik toko dengan merekap data bukti transaksi (nota) satu per satu dalam setiap harinya, sedangkan untuk toko cabang yang berada di Wangon dan Sampang, pemilik toko harus menunggu kedatangan karyawan dari toko cabang ke toko pusat untuk mengantar data bukti transaksi yang telah dilakukan pada masing – masing toko cabang. Pengantaran bukti transaksi dari toko cabang ke toko pusat harusnya diantar setiap hari, namun pada kenyataannya tidak dapat dilakukan setiap hari karena terbatasnya waktu yang dimiliki oleh karyawan, data bukti transaksi baru bisa diantar satu hari atau bahkan dua hari kemudian. Hal tersebut tentunya menghambat pembuatan laporan harian dan juga menghambat informasi mengenai perkembangan bisnis bagi pemilik toko.

Dari permasalahan tersebut diatas pemilik toko Jaya Elektronik membutuhkan sebuah aplikasi yang mampu mengatasi masalah yang berkaitan dengan penjualan, pencatatan data barang pada gudang, pelaporan dan aplikasi yang dibutuhkan oleh pemilik toko merupakan aplikasi yang terintegrasi dengan tiga toko agar ketiga toko yang dimiliki dapat terpantau langsung oleh pemilik toko.

*Point of sale* adalah sebuah sistem aplikasi yang diterapkan pada bisnis minimarket ataupun pertokoan untuk menangani pengolahan data transaksi pembelian, transaksi penjualan, transaksi *return* pembelian, dan pelaporan transaksi [1]. Persediaan atau *inventory* barang dipakai untuk menunjukkan barang – barang yang dimiliki untuk dijual. Setiap perusahaan selalu memerlukan persediaan untuk menghindari resiko tidak terpenuhinya keinginan pelanggan atau konsumen [2] . Aplikasi *point of sale* dan *inventory* oleh penulis disingkat menjadi aplikasi pository, aplikasi pository dibangun untuk menyelesaikan permasalahan yang terjadi pada toko Jaya Elektronik terkait transaksi penjualan, pengelolaan data barang pada gudang dan pelaporan semua transaksi yang terjadi pada tiga toko Jaya Elektronik, karena aplikasi *point of sale* dan *inventory* (pository) yang akan dibangun untuk ketiga toko Jaya Elektronik ini saling membutuhkan data satu sama lain, maka perlu dibangun aplikasi berbasis *web service* yang dapat memfasilitasi akses data dari aplikasi toko cabang (Jaya Elektronik Sampang dan Jaya Elektronik Wangon) ke aplikasi toko pusat (Jaya Elektronik Rawalo), dengan memanfaatkan *web service* yang

dapat melakukan komunikasi antar aplikasi sehingga proses pelaporan transaksi yang terjadi pada masing – masing toko cabang (Jaya Elektronik Sampang dan Jaya Elektronik Wangon) dapat terpantau langsung oleh pemilik toko

## Tinjauan Pustaka

### 1. Aplikasi

Aplikasi atau perangkat lunak aplikasi (bahasa Inggris: *software application*) adalah suatu subkelas perangkat lunak komputer yang memanfaatkan kemampuan komputer langsung untuk melakukan suatu tugas yang diinginkan pengguna. Biasanya dibandingkan dengan perangkat lunak sistem yang mengintegrasikan berbagai kemampuan komputer, tapi tidak secara langsung menerapkan kemampuan tersebut untuk mengerjakan suatu tugas yang menguntungkan pengguna. Contoh utama perangkat lunak aplikasi adalah pengolah kata, lembar kerja, dan pemutar media [3]

### 2. Point Of Sale

*Point of sale* (POS) merupakan kegiatan yang berorientasi pada penjualan serta sistem yang membantu proses transaksi. POS akan menjadi sangat penting di dunia bisnis karena POS diibaratkan berupa terminal uang dimana tempat menerima pembayaran dari pembeli kepada pedagang, karena pembayaran tersebut merupakan indikator bagi pembisnis untuk mengukur tingkat pendapatan mereka [4]

### 3. Inventory

*Inventory* (barang persediaan) adalah sejumlah material yang disimpan atau dirawat menurut aturan tertentu dalam tempat persediaan agar selalu dalam keadaan siap pakai dan ditatausahakan dalam buku perusahaan [5]

### 4. Web Service

Menurut Fauziah [6] *web service* dapat didefinisikan sebagai sekumpulan fungsionalitas yang dapat diakses melalui *Internet Protocol* (IP) standar. *Web service* banyak digunakan untuk aplikasi sistem yang terdistribusi secara heterogen baik dari sistem operasi sampai model objek. *Web service* merupakan entitas yang dapat diprogram dan menyediakan beberapa macam fungsi seperti informasi, aplikasi logik dan diakses banyak *platform* melalui beberapa media yaitu *Hypertext Transfer Protocol* (HTTP) dan XML.

### 5. UML

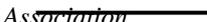
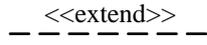
*Unified Modeling Language* (UML) adalah bahasa spesifikasi standar untuk mendokumentasikan, menspesifikasikan, dan membangun sistem perangkat lunak. *Unified Modeling Language* (UML) adalah himpunan struktur dan teknik untuk pemodelan desain program berorientasi objek (OOP) serta aplikasinya. UML adalah metodologi untuk mengembangkan sistem OOP dan sekelompok perangkat *tool* untuk mendukung pengembangan sistem tersebut [7]

Menurut [8] *UML* muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak. Adapun beberapa jenis diagram pada *UML* yang dapat membantu perancangan sistem adalah sebagai berikut:

a. *Use Case Diagram*

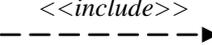
Menurut [8] *use case* atau *diagram use case* merupakan pemodelan kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada didalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu.

Tabel 1 Simbol *Use Case Diagram*

Simbol	Keterangan
	Fungsional yang disediakan sistem sebagai <i>unit-unit</i> yang saling bertukar pesan antar <i>unit</i> atau aktor; biasanya dinyatakan dengan menggunakan kata kerja diawal <i>frase</i> nama <i>use case</i>
Aktor / Actor  nama aktor	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu orang
Asosiasi / <i>Association</i> 	Komunikasi antar aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor
Ekstensi / <i>extend</i> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu; mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek; biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan

<i>Generalization</i> 	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya
---	---

Tabel 2 Simbol *Use Case Diagram* (lanjutan)

Simbol	Keterangan
Menggunakan / <i>include / uses</i>  	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini  Ada 2 sudut pandang yang cukup besar mengenai include di <i>use case</i> :  1) <i>Include</i> berarti <i>use case</i> yang ditambahkan akan selalu dipanggil, saat <i>use case</i> tambahan dijalankan.  2) <i>Include</i> berarti <i>use case</i> yang tambahan akan selalu melakukan pengecekan apakah <i>use case</i> yang ditambahkan telah dijalankan sebelum <i>use case</i> tambahan dijalankan

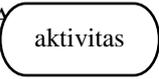
b. *Sequence Diagram*

Menurut [8] *sequence* menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambar *diagram sequence* maka harus diketahui objek-objek yang terlibat dalam sebuah *use case*. Beserta metode-metode yang dimiliki kelas yang diinstansiasi oleh objek itu. Membuat *diagram sequence* juga dibutuhkan untuk melihat skenario yang ada pada *use case*. Berikut simbol-simbol yang ada pada *diagram sequence* :

c. *Activity Diagram*

Menurut [8] diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Diagram aktivitas menggambarkan aktivitas yang dilakukan oleh sistem bukan apa yang dilakukan aktor.

Tabel 3 Simbol *Activity Diagram*

Simbol	Keterangan
Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal
Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
Percabangan /decision 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu

d. *Class diagram*

Menurut [8], *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas – kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. Atribut merupakan variabel yang dimiliki oleh kelas dan operasi atau metode adalah fungsi yang dimiliki oleh kelas. Diagram kelas dibuat agar pembuat program atau *programmer* membuat kelas-kelas sesuai rancangan didalam diagram kelas agar dokumentasi dan perangkat lunak sinkron. Kelas-kelas pada struktur sistem harus dapat melakukan fungsi-fungsi sesuai dengan kebutuhan sistem sehingga pembuat perangkat lunak atau *programmer* dapat membuat kelas-kelas didalam perangkat lunak sesuai dengan perancangan diagram kelas.

6. **HTML (Hyper Text Markup Language)**

*HTML (Hyper Text Markup Language)* adalah suatu bahasa yang digunakan untuk menulis halaman web. *HTML* dirancang untuk digunakan tanpa tergantung pada suatu platform tertentu (*Platform Independent*). Dokumen *HTML* adalah suatu dokumen teks biasa, dan disebut disebut sebagai *markup language* karena mengandung tanda-tanda (*tag*) tertentu yang digunakan untuk menentukan tampilan suatu teks dan tingkat kepentingan dari teks tersebut dalam suatu dokumen [9]

7. **PHP**

Menurut [10] *PHP* singkatan rekursif dari *PHP : Hypertext Preprocessor*, adalah bahasa pemrograman yang dapat digunakan untuk tujuan umum, sama seperti bahasa pemrograman lain : *C, C++, Pascal, Python, Perl, Ruby*, dan sebagainya.

Meskipun demikian, *PHP* lebih populer digunakan untuk pengembangan aplikasi web. Dalam proses pembuatan halaman web, *PHP* tidak memerlukan kode yang panjang seperti pada *perl* dan *python* (misalnya) karena kode *PHP* dapat disisipkan didalam kode *HTML*.

8. **CSS**

Menurut [11] *CSS (Cascading Style Sheet)* dikembangkan untuk menata gaya pengaturan halaman web. Pada awalnya *CSS* dikembangkan pada *SGML* pada tahun 1970, dan terus dikembangkan hingga saat ini *CSS* telah mendukung banyak Bahasa Markup seperti : *HTML, XHTML, XML, SVG (Scalable Vector Graphics)* dan *Mozilla XUL (XML User Interface Language)*.

9. **Javascript**

*Javascript* merupakan bahasa pemrograman yang tidak membutuhkan lisensi untuk dapat menggunakannya. Jika browser web yang kita gunakan mendukung (memiliki) *javascript* maka kita dapat membuat aplikasi berbasis web dengan menggunakan *javascript* [12]

10. **Bootstrap**

*Bootstrap* diciptakan oleh dua orang programmer *Twitter*, yaitu Mark Otto dan Jacob Thomson pada tahun 2011. *Bootstrap* awalnya diciptakan untuk membuat standarisasi pekerjaan tool dan library untuk pekerjaan programmer *Twitter*. Kemudian *bootstrap* berkembang dan populer pada kalangan programmer. *Bootstrap* adalah framework atau tools yang digunakan untuk aplikasi web ataupun situs web responsive secara cepat, mudah, dan gratis. *Bootstrap* terdiri dari *CSS* dan *HTML* untuk menghasilkan grid, layout, typography, table, form, navigation, dan lain-lain. Didalam *bootstrap* juga sudah terdapat *jQuery plugins* untuk menghasilkan komponen user interface yang cantik seperti transition, modal, dropdown, scrollspy, tooltip, tab, popover, alert, button, carousel, dan lain-lain [13]

11. **CodeIgniter**

*CodeIgniter* adalah sebuah web aplikasi Framework yang bersifat open source digunakan untuk membangun aplikasi PHP dinamis. Tujuan utama pengembangan *Codeigniter* adalah untuk membantu developer untuk mengerjakan aplikasi lebih cepat daripada menulis semua code dari awal. *Codeigniter* menyediakan berbagai macam library yang dapat mempermudah dalam pengembangan. *Codeigniter* diperkenalkan kepada publik pada 28 Februari 2006. *CodeIgniter* sendiri dibangun menggunakan konsep model view - controller development pattern [14]

12. **PhpMyAdmin**

*PhpMyAdmin* adalah perangkat lunak yang berbasis open source yang ditulis dalam bahasa pemrograman *PHP* yang digunakan untuk menangani administrasi *MySQL* melalui World Wide Web. *PhpMyAdmin* mendukung berbagai operasi *MySQL*, diantaranya mengelola basis data, tabel-tabel, bidang (fields), relasi (relations), indeks,

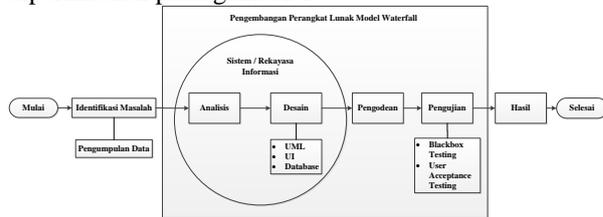
pengguna (*users*), perijinan (*permissions*), dan lain-lain [15]

### 13. Xampp

Menurut [16] *XAMPP* merupakan *tool* yang menyediakan paket perangkat lunak ke dalam satu buah paket. Dengan menginstal *XAMPP* maka tidak perlu lagi melakukan instalasi dan konfigurasi *web server Apache, PHP* dan *MySQL* secara manual. *XAMPP* akan menginstalasi dan mengkonfigurasikannya secara otomatis atau *auto* konfigurasi.

## Hasil dan Pembahasan

- A. Konsep penelitian merupakan serangkaian bagan – bagan yang menggambarkan alur dari proses penelitian dalam pembuatan aplikasi. Berikut merupakan bagan konsep penelitian yang diperlihatkan pada gambar 1

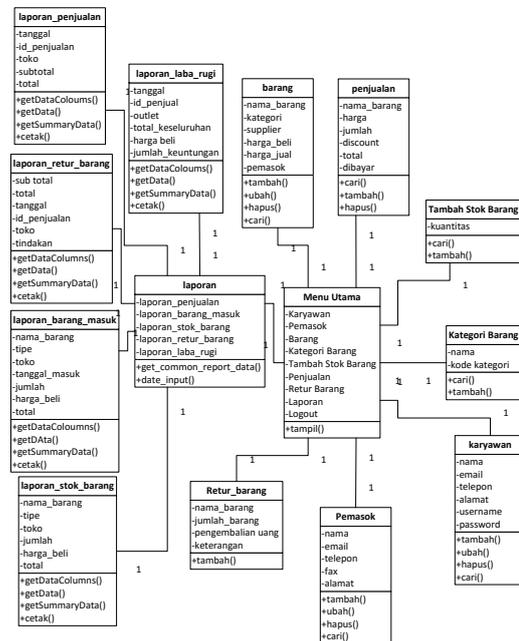


Gambar 1 Bagan Konsep Penelitian

Gambar 1 Menjelaskan konsep penelitian yang dilakukan oleh peneliti, pada bagian pertama yaitu identifikasi masalah, didalam identifikasi masalah terdapat pengumpulan data. Setelah melakukan identifikasi masalah dan melakukan pengumpulan data kemudian penulis menganalisis spesifik kebutuhan berdasarkan hasil wawancara, pada saat wawancara dengan narasumber yang menuturkan kepada peneliti tentang permasalahan yang dihadapi. selanjutnya pada bagian desain penulis membuat pemodelan sistem menggunakan *UML*, membuat desain *user interface* dan membuat rancangan *database*. Setelah desain selesai kemudian lanjut ke bagian pengodean, pada bagian ini penulis mengimplementasikan desain yang sudah dibuat sebelumnya kedalam bahasa pemrograman. Kemudian setelah pembuatan aplikasi selesai tahap berikutnya pengujian aplikasi yang telah dibuat, didalam pengujian penulis menggunakan *blackbox testing* untuk menguji fungsional aplikasi dan *user acceptance testing* untuk uji penerima pengguna aplikasi

- B. Class Diagram Jaya Elektronik

Berikut merupakan *class diagram* aplikasi Pository Multi Outlet Jaya Elektronik :

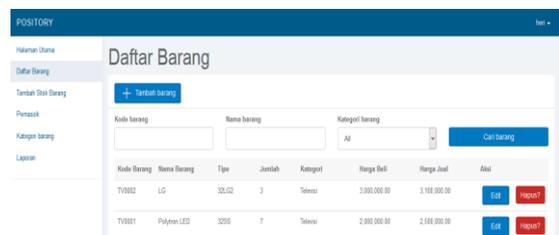


Gambar 2 Class Diagram Pository Multi Outlet Toko Jaya Elektronik

Gambar 2 Class Diagram Pository Multi Outlet Toko Jaya Elektronik menjelaskan semua fitur yang ada pada aplikasi diantaranya fitur karyawan, pemasok, barang, kategori barang, tambah stok barang, penjualan, *return* barang, dan laporan, laporan yang dihasilkan yaitu laporan berdasarkan stok barang, laporan barang masuk, laporan penjualan, laporan *return* barang, dan laporan laba rugi.

- C. Implementasi Antar Muka Daftar Barang

Berikut merupakan implementasi antar muka daftar barang pada aplikasi Pository Multi Outlet Jaya Elektronik :



Gambar 3 Implementasi Tampilan Antar Muka Daftar Barang

Gambar 3 Implementasi tampilan daftar barang untuk gudang. Didalam menu daftar barang, karyawan gudang dapat melakukan penambahan barang baru, merubah data barang, menghapus data barang dan melakukan pencarian data barang

D. Implementasi Antar Muka Tambah barang

Berikut merupakan implementasi antar muka *form* tambah barang pada aplikasi Pository Multi *Outlet* Jaya Elektronik :

Gambar 4 Implementasi Tampilan Antar Muka Tambah Barang

Gambar 4 Implementasi tampilan *form* tambah barang setelah memilih tombol tambah barang pada halaman menu daftar barang maka akan muncul *form* seperti diatas

E. Implementasi Antar Muka Laporan

Berikut merupakan implementasi antar muka laporan stok barang pada aplikasi Pository Multi *Outlet* Jaya Elektronik :

Nama	Tipe	Toko	Tanggal	Jumlah total	Harga Beli	Total
LG	321.02	Kaya Elektronik Randa	2019-01-11 01:20:30	3	3100000.00	9300000.00
Philips LED	3255	Kaya Elektronik Randa	2019-01-11 02:18:54	7	2500000.00	17500000.00
Casio	C3012	Kaya Elektronik Randa	2019-01-10 23:01:18	5	50000.00	250000.00

Gambar 5 Implementasi Tampilan Antar Muka Laporan Stok Barang

Gambar 5 Implementasi tampilan laporan stok barang. Pada tampilan laporan stok barang, *user* gudang dapat mencari data stok barang dan di *export* menjadi *file excel*.

**Kesimpulan dan Saran**

**A. Kesimpulan**

Berdasarkan proses analisis hasil dan pengujian yang telah dilakukan terhadap aplikasi Pository Multi *Outlet* pada toko Jaya Elektronik maka dapat diambil kesimpulan sebagai berikut :

1. Penelitian ini menghasilkan aplikasi Pository Multi *Outlet* yang dapat digunakan untuk melakukan proses transaksi penjualan, pengelolaan data barang pada gudang, dan proses pembuatan laporan dari tiga toko Jaya Elektronik.
2. Aplikasi Pository Multi *Outlet* yang dibangun terintegrasi dengan tiga toko untuk menyelesaikan permasalahan terkait proses

pelaporan transaksi yang terjadi pada masing – masing toko dan gudang Jaya Elektronik.

3. Dari hasil evaluasi penerimaan pengguna terhadap aplikasi menggunakan metode kuesioner, diperoleh hasil rata – rata penerimaan sebesar 31,5 yang termasuk dalam kategori sangat setuju sehingga dapat disimpulkan aplikasi Pository Multi *Outlet* dapat diterima dengan baik oleh pengguna.

**B. Saran**

Dalam penelitian yang telah dilakukan ini, peneliti menyadari tentu masih terdapat beberapa hal yang bisa dikembangkan atau menjadi acuan untuk penelitian-penelitian selanjutnya. Untuk mengembangkan penelitian yang telah dilakukan ini, maka peneliti memberikan saran antara lain adalah sebagai berikut : Aplikasi Pository Multi *Outlet* pada toko Jaya Elektronik yang telah dibangun masih memiliki kelemahan, oleh karenanya disarankan bagi para pembaca agar dapat mengembangkan pada bagian *database* dapat dibuatkan fitur *auto backup* yang berguna untuk *membackup* semua data yang ada dalam *database* dan *file backup* tersebut dapat disimpan dalam media penyimpanan lain yang dianggap lebih aman dari media penyimpanan pada PC yang digunakan.

1. Untuk fitur laporan dapat dikembangkan lagi menggunakan dasar – dasar akuntansi sehingga pemilik toko Jaya Elektronik lebih mengetahui perkembangan bisnisnya.

**Daftar Pustaka**

- [1] M. Nawang, L. Kurniawati, and D. Duta, “Rancang Bangun Sistem Informasi Pengolahan Data Persediaan Barang Berbasis Dekstop Dengan Modelwaterfall,” *J. PILARNusa Mandiri*, 2017.
- [2] A. Rahadi, M. Al Musadieg, H. Susilo, F. I. Administrasi, and U. Brawijaya, “BERBASIS KOMPUTER ( Studi Kasus pada Toko Arta Boga ),” *J. Adm. Bisnis (JAB)* Vol. 8 No. 2 Maret 2014, 2014.
- [3] T. Nurhadiyan and S. Hidayatullah, “Sistem Informasi Geografis Pemetaan Lokasi ATM BNI Terdekat di Kota Serang Berbasis Android,” *JSiI (Jurnal Sist. Informasi)*, 2017.
- [4] A. Profeta, R. Balling, and J. Roosen, “The relevance of origin information at the point of sale,” *Food Qual. Prefer.*, 2012.
- [5] A. Abdurahim and L. Fitriani, “Rancang Bangun Aplikasi Inventory Sayuran Berbasis Desktop di Perusahaan CV. Garut Power,” *J. Algoritm.*, 2017.
- [6] F. Ahmad, A. S. Yahaya, M. A. Osman, W. P. Lee, and A. B. Ismail, “Development of WILMS: Well inform landslide monitoring system for USM healthy campus,” in *Landslide Science and Practice: Early Warning, Instrumentation and Monitoring*, 2013.

- [7] R. V. Imbar and B. S. Hartanto, “Aplikasi Sistem Informasi Sumber Daya Manusia dengan Fitur DSS Menggunakan Metode Topsis pada PT . X,” *J. Sist. Inf.*, 2008.
- [8] R. A.S and S. M, *Rekayasa Perangkat Lunak*. 2015.
- [9] Ponidi and S. Fitrajaya, “Perancangan Sistem Informasi Pendataan Penduduk Berbasis Web Menggunakan Metode Waterfall Pada Kecamatan Gadingrejo,” *J. TAM (Technology Accept. Model.*, 2015.
- [10] Yuhefizar, “10 Pengertian Website Menurut Para Ahli | Situs Berita Pendidikan,” 28 *Febr. 2016*, 2016.
- [11] Sibero, “Wolrd Wide Web,” *Web Program. Power Pack*, 2014.
- [12] Rizky and D. Y. U. Multi Amalia, “Java Script,” *Pemberian Reward Berdasarkan Penilai. Kinerja Karyawan Dengan Metod. AHP Pada PT. Anegerah Protecindo*, 2018.
- [13] A. Khumaidi, “Perancangan Aplikasi Marketplace Order Baju Pintar Menggunakan Web Responsif Untuk Memudahkan Customer Mendesain Sesuai Selera,” *Ikhraith-Informatika*, 2018.
- [14] A. K. Adi Pradana, A. Izzah, and W. A. Febianto, “Rancang Bangun Situs Forum Komunitas Perfilman Berbasis CodeIgniter,” *J. Inform. dan Multimed.*, 2015.
- [15] M. Jamil and H. Bunyamin, “Pengembangan Aplikasi Sistem Informasi Laporan Keuangan Walisantri di Pondok Pesanren Al-Halim Garut,” *J. Algoritm.*, 2016.
- [16] A. R. Djaelani, Sunyono, and N. P. Lestari, “Teknik Wawancara (Interview) dalam penelitian Kualitatif,” *Pendidik. Sains, Fak. Pascasarjana, Univ. Negeri Surabaya*, 2013.