

APLIKASI POINT OF SALE BERBASIS WEB MENGGUNAKAN CODEIGNITER

Purwadi Budi Santoso¹, Budiman²

Teknik Informatika, Sekolah Tinggi Teknologi Mandala Bandung

Abstrak

Aplikasi *Point Of Sale* berbasis web ini merupakan aplikasi yang dibuat untuk mengolah data jual beli secara komputerisasi. Di Toko 134 Sukamenak transaksi jual beli belum terkomputerisasi. Aplikasi *Point Of Sales* ini dibangun dengan menggunakan metode *Waterfall*. Tujuan dari penelitian ini adalah memfasilitasi kasir/admin dalam memanipulasi data secara komputerisasi. Metode penelitian yang diambil adalah Metode Penelitian *Action Research*. Teknik pengumpulan data yang dilakukan yaitu observasi, dan studi pustaka untuk menganalisa kebutuhan dari aplikasi *POS*. Perancangan Aplikasi *POS* ini menggunakan *Unified Modeling Language* untuk menggambarkan proses yang terjadi pada aplikasi. Diagram yang digunakan yaitu *use case diagram*, *activity diagram*, *sequence diagram*, dan *class diagram*. Teknologi yang digunakan yaitu *PHP* dan *Javascript* sebagai bahasa pemrograman, *CodeIgniter* sebagai *framework*, dan *MySQL* sebagai *database*. Dengan hadirnya *POS* diharapkan bisa mengurangi kesalahan input data, data rusak, data hilang, Serta mempunyai fitur cetak struk otomatis. Masih banyak kekurangan dalam penelitian ini. saran untuk pengembangan ke depan seperti fitur *barcode scanner*, menampilkan total transaksi perhari dan perbulan, fitur diskon per item, fitur diskon dalam bentuk persen, pilihan metode pembayaran selain metode tunai, serta menambah fitur keuntungan bagi customer yang sudah terdaftar.

Kata kunci : Aplikasi, *Point Of Sale*, *PHP*, *MySQL*, *CodeIgniter*.

Abstract

This web-based Point Of Sale application is created to process buying and selling data in a computerized manner. At 134 Sukamenak Shop, buying and selling transactions are not yet computerized. This Point Of Sales application using the Waterfall method. this research to facilitate cashiers/admins in manipulating data in a computerized. The research method taken is the Action Research. The data collection techniques used were observation and literature study to analyze the needs of the POS application. This POS application design uses the Unified Modeling Language to describe the processes that occur in the application. The diagrams used are use case , activity , sequence , and class diagrams. The technology used is PHP and Javascript as programming languages, CodeIgniter as a framework, and MySQL as a database. With the presence of POS, it reduce data input errors, damaged data, lost, and have an automatic receipt printing feature. There are still many shortcomings in this research. For future development as the barcode scanner , displaying total daily and monthly transactions, discount feature per item, discount feature in the form of a percentage, choice of payment methods other than cash, benefit features for registered customers.

Keywords : Application, *Point Of Sale*, *PHP*, *MySQL*, *CodeIgniter*.

1. PENDAHULUAN

Di era globalisasi, teknologi berkembang sangat pesat, terutama di bidang teknologi informasi yang membutuhkan penggunaan komputer untuk membantu manusia dalam menyelesaikan tugasnya dalam kehidupan sehari-hari. *Point of sale* adalah aplikasi teknologi informasi yang berkembang pesat. Layanan pengadaan barang dan transaksi penjualan produk dapat dilakukan secara elektronik. Di Toko 134 Sukamenak Sistemnya masih menggunakan sistem pencatatan manual yang dimana terjadi tidak terdatanya informasi secara komputerisasi pada penulisan transaksi penjualan, tidak terdatanya manajemen *stock*, belum tersedianya aplikasi kalkulasi jumlah pembelanjaan otomatis, serta belum tersedianya struk pembelian total yang berisi data pembelian bagi konsumen. Oleh karena itu, sangat penting bagi Toko 134 Sukamenak untuk memiliki sistem berbasis teknologi yang dapat lebih efektif mengurangi informasi yang salah tulis dan pendataan penjualan.

Tujuan dari proposal ini adalah menganalisis transaksi di Toko 134 Sukamenak dan merancang aplikasi *point-of-sale* (POS) untuk mendukung sistem penjualan dan manajemen *stock* yang dapat membantu pengelolaan data, baik itu data penjualan maupun data *stock*. Pembuatan aplikasi POS ini dimulai dengan mengumpulkan data yang dibutuhkan menggunakan metode observasi. perancangan model aplikasi menggunakan metode diagram berbasis objek menggunakan flowchart dan alat bantu desain aplikasi berupa Unified Modelling Language (UML) hingga mengimplementasikan Prosedur aplikasi POS ini. Melalui implementasi Aplikasi *Point of Sale* (POS) ini membantu pihak-pihak yang berkepentingan atau seluruh pemangku kepentingan yang terkait

langsung dengan Aplikasi POS ini dengan tugas-tugasnya.

Berdasarkan uraian di atas, perlu dikembangkan suatu rancangan aplikasi *point of sale* dengan menggunakan *CodeIgniter*. *Codeigniter* adalah aplikasi *open source* berupa *framework PHP* dengan model MVC (*Model, View, Controller*) untuk membangun *website* dinamis menggunakan *PHP*. *Codeigniter* memudahkan dan mempercepat pengembang web untuk membuat aplikasi web dibandingkan membangun aplikasi web dari awal.

2. METODOLOGI PENELITIAN

2.1. Metode Penelitian Tindakan (Action Research)

Metode penelitian yang diambil dalam penelitian ini adalah Metode Penelitian Tindakan (*Action Research*). Penelitian Tindakan (*Action Research*) atau disingkat AR ditandai dengan pendekatan *systematic inquiry* yang memiliki ciri, prinsip, pedoman, prosedur yang harus memenuhi kriteria tertentu. Penelitian tindakan harus jelas membedakan perbedaan ciri tindakan dan penelitian, harus terlibat langsung dan bukan hanya sekadar sebagai penonton (Semiawan, 2007). Dikatakan pula bahwa penelitian tindakan yaitu suatu proses demokratis dan partisipatorik yang menyangkut pengembangan pengetahuan praktis dalam upaya mencari tujuan yang bermanfaat demi kemaslahatan kehidupan di dunia (Coghlan and Brannick, 2005).

Selanjutnya, penelitian tindakan selalu berhubungan dengan tindakan untuk mencapai hasil praktis dan menciptakan bentuk pemahaman baru, karena tindakan tanpa pengetahuan ialah buta dan teori tanpa tindakan tidak berarti (Koshy, 2005). Secara operasional bentuk penelitian tindakan yaitu rangkaian kegiatan bersama yang berkelanjutan antara para pihak

terkait dalam hal merencanakan, melaksanakan, dan mengevaluasi rangkaian upaya untuk mencapai perubahan status pola pikir, pandang, kerja, dan sikap baru yang disadari sebagai tindakan yang bersifat dinamis terhadap perubahan selanjutnya (Gay, Mills, and Airasian, 2009).

Hampir sama dengan penelitian *mixed method* atau metode gabungan, penelitian tindakan menggunakan metode pengumpulan data yang dapat dilakukan secara kualitatif dan kuantitatif, tetapi selalu diarahkan pada isu yang bersifat spesifik dan praktis, dan berusaha mencari solusi terhadap permasalahan yang dihadapi. Dengan begitu, penelitian tindakan merupakan prosedur sistematis yang dilakukan oleh guru (atau individu lain dalam pelaksanaan pendidikan) untuk mengumpulkan informasi guna memperbaiki cara penelenggaraan pendidikan, baik dari segi belajar yang dilakukan oleh peserta didik maupun dari sisi pembelajaran yang disajikan pendidik.

Berdasarkan pengertian tersebut dapat dipahami bahwa penelitian tindakan memiliki tujuan dan prinsip dasar yang sedikit berbeda dengan metode penelitian lainnya. Tujuan dasar penelitian tindakan lebih ditujukan untuk meningkatkan praktik ketimbang memproduksi pengetahuan, berfokus pada praktik sosial, bertujuan untuk peningkatan keadaan, merupakan proses siklus, diikuti dengan temuan sistematis, merupakan proses reflektif, bersifat partisipatif, dan topik atau masalahnya ditentukan oleh praktisi.

2.2. Pengumpulan Data

Dalam penelitian ini langkah awal yang penulis lakukan adalah pengumpulan data dengan cara observasi dan studi pustaka.

- Observasi

Observasi merupakan salah satu jenis teknik pengumpulan data yang banyak dilakukan. Melansir buku *Praktis Penelitian Kualitatif Teori Dasar dan Analisis Data Dalam Perspektif Kualitatif* oleh Mardawani, secara umum observasi merupakan aktivitas pengamatan terhadap objek di lapangan. Kegiatan observasi juga mengharuskan peneliti mencatat berbagai gejala yang ditemui secara sistematis. Adapun berbagai informasi yang didapatkan melalui observasi menurut Mardawani berupa tempat (ruang), pelaku, kegiatan, objek, perbuatan, peristiwa, waktu, dan perasaan. Tujuan penting dilakukannya observasi, yakni untuk memberikan gambaran realistis kepada peneliti tentang suatu perilaku atau kejadian yang berkaitan dengan aktivitas objek penelitian.

- Studi Pustaka

Mencari referensi yang terkait dengan buku tentang pemrograman *PHP & MySQL*, buku tentang *Sistem Basis Data*, buku tentang Manajemen Persediaan, buku dan *website* tentang pembuatan web menggunakan *framework Codeigniter*, dan jurnal tentang pembuatan aplikasi *Point Of Sales*.

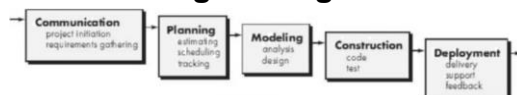
2.3. Perancangan Desain

Perencanaan desain adalah perencanaan untuk mengembangkan suatu produk yang akan dihasilkan seiring dengan proses pengembangan. Langkah selanjutnya adalah menentukan desain dan analisis untuk merumuskan pembuatan aplikasi web yang akan dikembangkan.

2.4. Pengembangan Desain

Proses pengembangan desain merupakan proses pembuatan website secara umum seperti, pembuatan database, bahasa pemrograman yang digunakan, framework yang digunakan, dan proses coding yang diolah menjadi sebuah aplikasi web. Dalam aplikasi *Point Of Sale* Berbasis web, penulis memproses menggunakan *Visual Studio Code* sebagai *text editor*. *Visual Studio Code* merupakan editor kode sumber yang di buat dan diciptakan berbagai *software* seperti *Microsoft* untuk *Windows*, *Linux*, dan *macOS*. Fitur ini termasuk dukungan untuk *debugging*, *penyorotan sintaks*, *penyelesaian* kode cerdas, cuplikan, pemfaktoran ulang kode, dan *Git* yang disematkan. Penulis juga menggunakan *MySQL* sebagai media pembuatan *database*, *PHP* sebagai Bahasa pemrograman, dan menggunakan *CodeIgniter* sebagai *Framework*. *CodeIgniter* merupakan aplikasi sumber terbuka yang berupa kerangka kerja *PHP* dengan model *MVC* (*Model*, *View*, *Controller*) untuk membangun situs web dinamis dengan menggunakan *PHP*. Sehingga memudahkan penulis untuk membuat Aplikasi *Point Of Sale* berbasis web ini.

2.4. Model Pengembangan Sistem



Gambar 1 Model *Waterfall* Pressman (2015)

Sumber :

<https://kotakode.com/blogs/13032/Mengenal-Tahapan-Metode-Waterfall>

Dalam penelitian ini, penulis menggunakan metode *waterfall* dalam Pengembangan Aplikasi *Point Of Sale* berbasis web sebagaimana ditunjukkan pada Gambar 1.

adapun tahapan yang penulis lakukan adalah sebagai berikut :

- Analisis Kebutuhan Perangkat Lunak
Pada tahap pertama ini, penulis mengumpulkan semua kebutuhan untuk membuat aplikasi berbasis web, mulai dari *database*, bahasa pemrograman, dan *framework* yang akan digunakan, serta kebutuhan lainnya seperti kebutuhan fungsional dan non fungsional dari aplikasi berbasis web tersebut yang penulis buat.
- Desain
Jika semua kebutuhan sudah terasa mencukupi maka berikutnya penulis mulai membuat *database* agar mendukung dalam bahasa pemrograman *PHP* agar *website* menjadi dinamis, Mendesain tampilan *website* yang dengan memanfaatkan fitur-fitur yang ada di *framework Codeigniter*.
- Pembuatan kode program
Setelah semua desain selesai maka langkah selanjutnya adalah pembuatan kode program, dalam hal pembuatan aplikasi *Point Of Sale* berbasis web ini, penulis memasukkannya ke dalam *Visual Studio Code*, dengan menggunakan *HTML* sebagai bahasa *markup*, *PHP* dan *Javascript* sebagai bahasa pemrogramannya. bahasa pemrograman dan *CodeIgniter* sebagai *frameworknya*. Aplikasi akan dibuat sesuai dengan kebutuhan *administrator* dan kasir untuk mengumpulkan data produk dan menjual produk, serta memberikan manfaat belanja dan pengalaman baru kepada pelanggan melalui aplikasi *Point Of Sale* berbasis web ini.
- Pengujian Unit
Setelah aplikasi *Point Of Sale* berbasis web selesai dibuat, langkah selanjutnya adalah melakukan pengujian, disini penulis menggunakan metode pengujian *blackbox*, dan penulis dapat menggunakan ini untuk menilai apakah aplikasi *Point Of Sale* berbasis web dapat digunakan sesuai dengan keinginan.

- **Pemeliharaan**

Fixing merupakan tahap akhir dari pendekatan *waterfall*, dimana aplikasi yang telah selesai dioperasikan dan dipelihara oleh pengguna, dalam hal ini penulis akan selalu melakukan perbaikan terhadap setiap *bug* yang tidak terdeteksi oleh tahap sebelumnya.

3. HASIL DAN PEMBAHASAN

3.1. Kebutuhan Perangkat Keras

Spesifikasi perangkat keras yang digunakan oleh penulis untuk mengembangkan Aplikasi *Point Of Sales* ini adalah sebagaimana pada Tabel 4.

Tabel 3 Perangkat Keras yang digunakan

No	Perangkat	Spesifikasi
1	Laptop	MacBook Pro (13 inci, M1, 2020)
2	Processor	Chip Apple M1 CPU 8-core dengan 4 core performa dan 4 core efisiensi GPU 8-core Neural Engine 16-core
3	Sistem Operasi	macOS Ventura Version 13.0.1
4	RAM	8GB
5	SSD	256 GB

3.2. Kebutuhan Perangkat Lunak

Pada pembuatan *Point Of Sale* Berbasis *web* ini penulis memerlukan perangkat lunak sebagai media pengembangan. Perangkat lunak yang digunakan antara lain.

1. *XAMPP*
2. *Visual Studio Code*
3. *GoogleChrome*

3.3. Kebutuhan Perangkat Lunak

Analisa kebutuhan—kebutuhan proses mengidentifikasi dan menguraikan kebutuhan yang diharapkan sehingga dapat dicari solusi untuk memecahkannya.

1. Kebutuhan (*requirement system*)

- a. Kebutuhan Secara Fungsional
Kebutuhan secara fungsional merupakan yang akan dilakukan oleh *Admin* dan *user* untuk menampilkan aplikasi program yang akan dibuat sebagaimana ditunjukkan pada tabel 5.

Tabel 5 Hak Akses

No	Akses	Hak Akses	
		Admin	Kasir
1	Pengelolaan data Users	√	×
2	Pengelolaan data Suppliers	√	√
3	Pengelolaan data Customers	√	√
4	Pengelolaan data Product Categories	√	√
5	Pengelolaan data Product Units	√	√
6	Pengelolaan data Product Items	√	√
7	Pengelolaan data Transaction Sales	√	√
8	Pengelolaan data Transaction Stock In	√	√
9	Pengelolaan data Sales Report	√	√

- b. Kebutuhan Secara Kinerja
Kebutuhan secara kinerja merupakan perbedaan antara sistem lama dan sistem baru sebagaimana ditunjukkan pada tabel 6

Tabel 6 Hak Akses

Sistem Lama	Sistem Baru
Belum ada sistem pengolahan data yang terkomputerisasi.	Pengolahan data transaksi menjadi terkomputerisasi dengan aplikasi <i>Point of Sale</i> .
Kesalahan mungkin terjadi dalam pengumpulan dan perhitungan data.	Dengan aplikasi <i>Point of Sale</i> , pengumpulan dan perhitungan data otomatis oleh sistem.
Dalam pengolahan data manual, mengakibatkan data bisa terjadi kesalahan input data, data rusak, dan hilang	Dengan aplikasi <i>Point of Sale</i> , meminimalisir terjadinya kesalahan <i>input</i> data, data rusak, atau data hilang karena data disimpan dalam <i>database</i> .

3.4. Desain

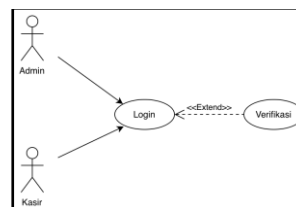
Pada tahap ini akan dijelaskan mengenai perancangan desain menggunakan UML. Berikut daftar diagram yang akan dibuat :

- Use case diagram.*
- Activity diagram.*
- Sequence diagram.*
- Deployment diagram.*

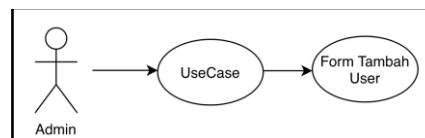
3.4.1. Use Case Diagram

1. Use Case Diagram

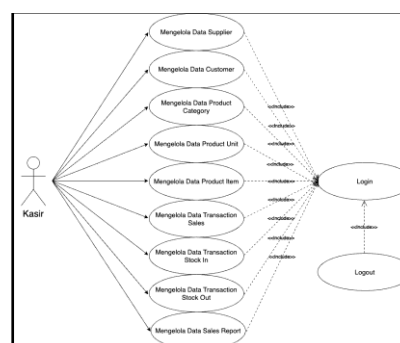
Use Case Diagram adalah model yang menggambarkan aktivitas sistem yang akan dilakukan. Diagram *use case* menggambarkan interaksi antara satu atau lebih aktor dengan sistem yang akan dibuat. *Use case diagram* digunakan untuk mengetahui fungsi apa saja yang tersedia pada sistem dan siapa saja yang memiliki akses terhadap fungsi tersebut. Adapun kasus penggunaan Sistem yang disarankan sebagaimana ditunjukkan pada gambar 1 sampai gambar 4 .



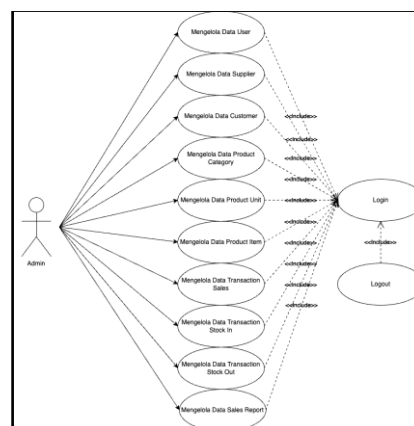
Gambar 1 Diagram Use Case Login yang diusulkan



Gambar 2 Diagram Use Case Tambah User yang diusulkan



Gambar 3 Diagram Use Case Admin yang diusulkan



Gambar 4 Diagram Use Case Kasir yang diusulkan

2. Skenario Use case

Skenario Use Case menjelaskan proses dari awal hingga akhir proses dalam diagram *use case*, dan proses yang lebih detail sama dengan metodologi berorientasi objek, sehingga menggunakan *use case* untuk menjelaskan skenario proses secara *global*. Skenario kasus penggunaan sistem yang disarankan akan didefinisikan.

Tabel 7 Skenario *Use Case Login* yang diusulkan

Aktor	Sistem
1. Membuka <i>web</i>	
2. Masukan <i>username</i> dan <i>password</i>	
	3. Validasi <i>username</i> dan <i>password</i>
	4. Menampilkan notifikasi <i>username</i> dan <i>password</i> sesuai / tidak sesuai dengan <i>database</i>
	5. Masuk ke halaman <i>dashboard</i>

Tabel 8 Skenario *Use Case Tambah Data User* yang diusulkan

Aktor	Sistem
1. Memilih menu <i>user</i>	
	2. Menampilkan data <i>users</i>
3. Admin menekan tombol <i>create</i>	
	4. Menampilkan <i>form</i> tambah <i>user</i>

Tabel 8 Skenario *Use Case Tambah Data User* yang diusulkan

Aktor	Sistem
5. Admin memberikan data registrasi	
6. Admin menekan tombol <i>save</i>	

	7. Sistem akan menyimpan data <i>user</i> ke <i>database</i>
	8. Selesai melakukan registrasi user

Tabel 9 Skenario *Use Case Transaksi Penjualan*

Aktor	Sistem
1. Memilih menu <i>Transaction</i>	
2. Memilih menu <i>Sub Menu Transaction (Sales)</i>	
	3. Menampilkan <i>form</i> transaksi Penjualan
4. Pilih tombol Pencarian	
	5. Menampilkan Data <i>Items</i>
6. Pilih salah satu item yang dibeli <i>Customer</i>	

Tabel 9 Skenario *Use Case Transaksi Penjualan*

Aktor	Sistem
	7. Masukan jumlah Qty items yang dibeli <i>customer</i>
8. Pilih tombol <i>add</i>	
	9. Menampilkan jumlah pembayaran yang harus dibayarkan oleh <i>customer</i>
10. Mengisi diskon (Jika diperlukan) dan Uang yang	

dibayarkan Customer	
11. Pilih tombol <i>Process Payment</i>	

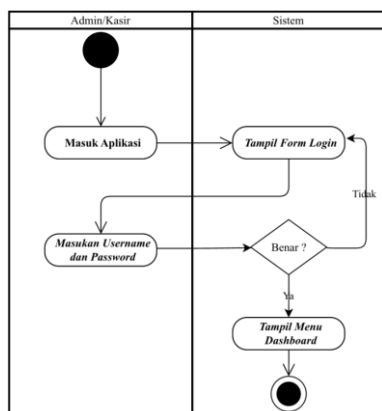
Tabel 9 Skenario *Use Case* Transaksi Penjualan

Aktor	Sistem
	12. Mengkalkulasi Transaksi Penjualan
	13. Menambahkan Data <i>Transaction</i>
	14. Mencetak Struk pembelian

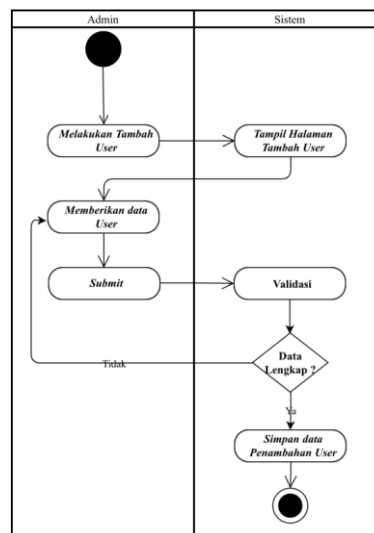
3.4.2. Activity Diagram

1. Activity Diagram

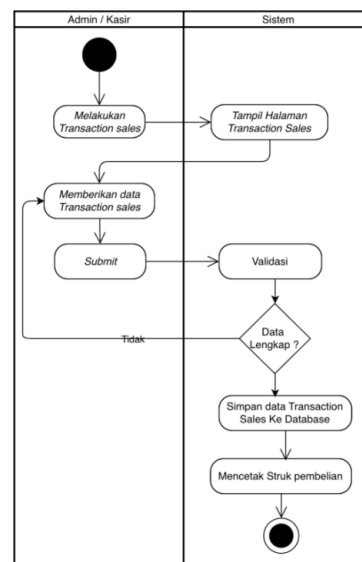
Activity diagram atau diagram aktivitas dapat digunakan untuk menunjukkan alur kerja bisnis. Itu juga dapat digunakan untuk menggambarkan aliran peristiwa dalam *use case*. *Activity diagram* akan akan didefinisikan.



Gambar 5 Activity Diagram Login Yang Diusulkan



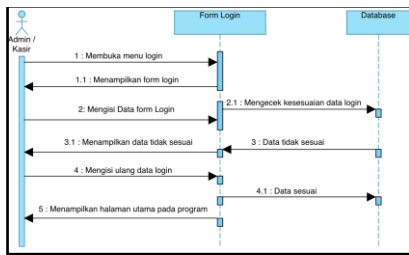
Gambar 6 Activity Diagram Tambah User Yang Diusulkan



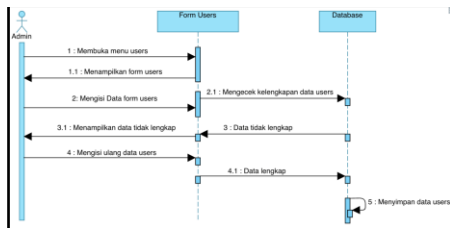
Gambar 7 Activity Diagram Transaction Sales Yang Diusulkan

3.4.3. Sequence Diagram

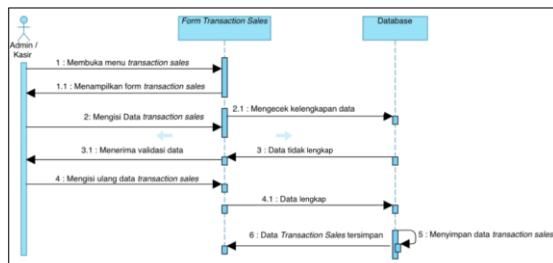
Menggambarkan kolaborasi dinamis antara banyak objek. Tujuannya adalah untuk menunjukkan urutan pesan yang dikirim antar objek dan interaksi antar objek, yang terjadi di beberapa titik dalam eksekusi sistem. Komponen utama *sequence diagram* terdiri dari objek yang ditulis dalam persegi panjang yang disebut pesan, diwakili oleh garis dengan panah, dan waktu diwakili oleh proses vertikal.



Gambar 8 Sequence Diagram Login Yang Diusulkan



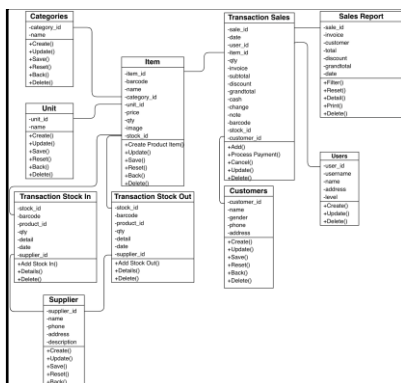
Gambar 9 Sequence Diagram Tambah Data User Diusulkan



Gambar 10 Sequence Diagram Sales Transaksi Yang Diusulkan

3.4.4. Sequence Diagram

Menjelaskan struktur kelas statis dalam sistem. Kelas mewakili hal-hal yang ditangani sistem. Melihat karakteristik dan proses sistem pemasaran produk dari bagian penjualan maka dapat dibuat suatu *class diagram*, berikut adalah *class diagram* sistem informasi penjualan.



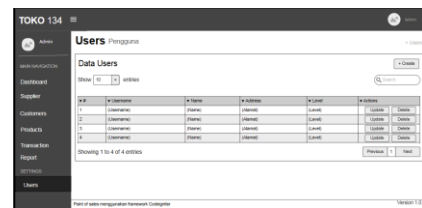
Gambar 11 Class Diagram

4.5. Perancangan User Interface

Perancangan UI Aplikasi *Point of Sale* ini meliputi *Interface Login*, *Tambah Data User*, dan *Transaction Sales*.



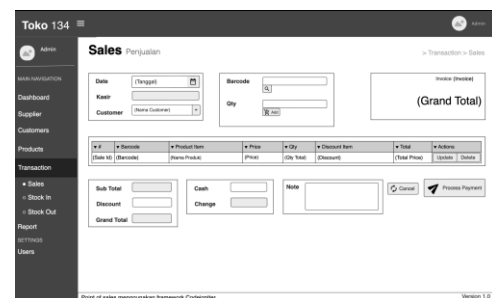
Gambar 12 Rancangan Login



Gambar 12 Rancangan halaman Data Users



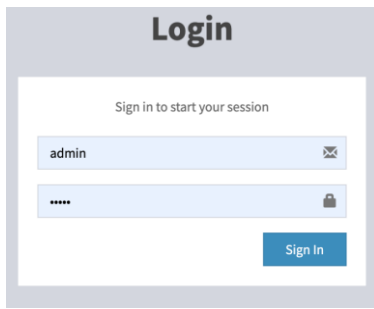
Gambar 13 Rancangan halaman Tambah Data Users



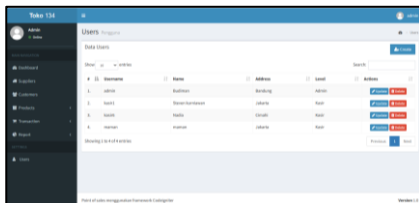
Gambar 14 Rancangan halaman Transaction Sales

4.6. Implementasi Sistem

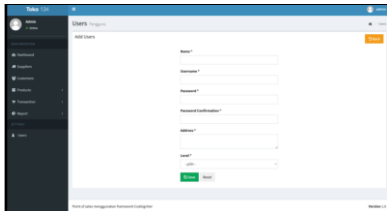
Hasil dari penelitian ini adalah sebuah aplikasi *Point Of Sale* berbasis web, pada tahap ini penulis akan mengimplementasikan *user interface* aplikasi *Point Of Sale* yang sudah dibuat.



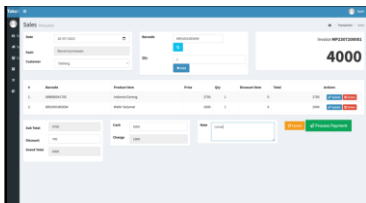
Gambar 15 Tampilan Halaman *Login*



Gambar 16 Tampilan Halaman Form Data *User*



Gambar 17 Tampilan Halaman Form Tambah Data *User*



Gambar 18 Tampilan Halaman Form Transaction *Sales*

3.7. **Black Box Testing**

Black box testing adalah metode pengujian aplikasi yang fungsionalitasnya diuji tanpa pengetahuan tentang detail implementasi, struktur kode, dan jalur internal. Pengujian hanya meninjau input dan output aplikasi yang sepenuhnya berbasis spesifikasi dan persyaratan *software*. Berikut tabel *black box testing* yang dilakukan sebagaimana ditunjukkan pada Tabel 10.

Tabel 10 Skenario *Use Case* Transaksi Penjualan

No	Pengujian	Hasil Yang Diharapkan	Hasil Program
1	<i>Login</i>	Menampilkan Halaman <i>Dashboard</i>	Menampilkan Halaman <i>Dashboard</i>
3	Menampilkan data <i>users</i>	Menampilkan data <i>users</i>	Menampilkan data <i>users</i>
4	Menambah data <i>user</i>	Data <i>user</i> sukses ditambah	Data <i>user</i> sukses ditambah
5	Memproses Penjualan	Data <i>sales</i> sukses diproses, mencetak struk belanja, dan <i>stock item</i> berkurang	Data <i>sales</i> sukses diproses, mencetak struk belanja, dan <i>stock item</i> berkurang

4. KESIMPULAN DAN SARAN

4.1. Kesimpulan

- Admin dan kasir dapat menginput data penjualan dan barang secara komputerisasi.
- Mempermudah Admin dan kasir fasilitas dalam memanipulasi data stok barang, penjualan, *customer*, dan *supplier*.
- Memudahkan transaksi penjualan dengan sistem perhitungan jumlah total belanja otomatis.
- Aplikasi web ini bisa mencetak struk belanja secara otomatis.

4.2. Saran

Masih banyak kekurangan dalam penelitian ini dan hal-hal yang perlu pengembangan lebih lanjut. Penulis memiliki saran dan pemikiran untuk pengembangan ke depan, antara lain:

- Menambah fitur *barcode scanner* yang berguna untuk transaksi tambah produk dan transaksi penjualan.

- Menambah fitur yang menampilkan total transaksi perhari dan perbulan.
- Menambah fitur diskon per item.
- Menambah fitur diskon dalam bentuk persen.
- Menambah pilihan metode pembayaran lain selain metode pembayaran tunai.
- Menambah fitur keuntungan bagi customer yang sudah terdaftar.

DAFTAR PUSTAKA

- S., Rosa dan Shalahuddin, M.(2015),
Rekayasa Perangkat Lunak. Struktur dan Berorientasi Objek, Bandung : Informatika.
- Alessandro, D.S.. (2018). *Visual Studio Code Distilled*. California: Apress.
- Dicoding, Intern .(2022). *Apa itu Activity Diagram? Beserta Pengertian, Tujuan, Komponen*.
<https://www.dicoding.com/blog/apa-itu-activity-diagram/> (diakses pada tanggal 10 Juni 2023)
- Dicoding, Intern .(2022). *Black Box Testing Untuk Menguji Perangkat Lunak*
<https://www.dicoding.com/blog/black-box-testing/> (diakses pada tanggal 10 Juni 2023)
- Dicoding, Intern.(2022). *Contoh Use Case Diagram Lengkap dengan Penjelasannya* :
<https://www.dicoding.com/blog/contoh-use-case-diagram/>(diakses pada tanggal 10 Juni 2023)
- Enterprise, J. (2015). *Trik Mudah Membuat POS dengan Excel*. Jakarta: Elex Media Komputindo.
- Fitri, M. (2017). *Perancangan aplikasi Point Of Sales (POS) berbasis web menggunakan metode siklus hidup pengembangan sistem*
<https://jurnal.unmer.ac.id/index.php/jtmi/article/view/1514> (diakses pada tanggal 11 Juni 2023)
- Idcloudhost (2017). *Mengenal Apa itu Framework CodeIgniter*
<https://idcloudhost.com/panduan/mengenal-apa-ituframework-codeigniter/> (diakses pada tanggal 11 Juni 2023)32
- Kabar, harian. (2021). *Observasi dan Wawancara dalam Penelitian Kualitatif, Ketahui Definisi & Jenisnya*
<https://kumparan.com/kabar-harian/observasi-> (diakses pada tanggal 13 Juni 2023)
- Kromann, F. (2018). *Beginning PHP and MySQL: From Novice to Professional*. California: Apress.
- Nurhadi, S. (2019). *Sistem Basis Data: Mysql di Ubuntu*. Edisi 2. Jakarta: Yayasan. Manajemen dan Teknologi Nasional..
- Pamungkas, C. A.(2017). *Pengantar Dan Implementasi Basis Data*. Yogyakarta: Deepublish.
- Pranaya, A. & Hendra, A. (2019). *Membuat Toko Online dengan Menggunakan Framework Bootstrap 4*. Cimahi: PT. Dinasti Motekar Grup.
- Ramadhanti, A.(2023). *3 Contoh Class Diagram dan Pengertiannya* .
https://www.pinhome.id/blog/contoh-class-diagram/#Pengertian_Class_Diagram (diakses pada tanggal 10 Juni 2023)
- Rachmadi, T. (2020). *Sistem Basis Data*. Lampung: Tiga Ebook.
- Rahmawati, B.F & Amar, S (2020). *Evaluasi Pembelajaran Sejarah*. Nusa Tenggara Barat: Universitas Hamzanwadi Press.

- Rusdalpn. (2021). Mengenal Tahapan Metode *Waterfall Pressman* <https://kotakode.com/blogs/13032/Mengenal-Tahapan-Metode-Waterfall>. (diakses pada tanggal 17 Juni 2023)
- Solichin, A. (2016). Pemrograman Web dengan PHP dan MySQL. Jakarta: Universitas Budi Luhur.
- Subagja, A (2018). Kolaborasi CodeIgniter dan Ajax dalam Perancangan CMS. Jakarta: PT Elex Media Komputindo.
- Subandi, & Syahidi, A. A. (2018). BASIS DATA Teori dan Praktik Menggunakan. Microsoft Office Access. Banjarmasin: Deepublish.).
- Vikaliana, R. (2020). Manajemen Persediaan. Bandung: Media Sains Indonesia.
- Yaumi, M. (2016). *Action Research* Teori, model, dan aplikasi. Jakarta: Prenada Media.