

Rancang Bangun Aplikasi *Web Reporting Point Of Sales* Pada Distributor Kain Hoggy Djaya

Robby Lukito Universitas Ciputra UC Town, Citraland, Surabaya

ABSTRAK

Laporan transaksi perusahaan merupakan faktor penting dalam berkembangnya sebuah perusahaan. Masalah umum yang ada pada perusahaan adalah kesalahan dalam laporan transaksi karena pencatatan laporan dilakukan secara manual. Untuk itu dibutuhkan sebuah aplikasi yang dapat menyajikan data laporan transaksi dari sebuah perusahaan secara cepat dan tepat. Melalui aplikasi *web reporting point of sales*, perusahaan dapat melihat laporan transaksi yang terjadi di perusahaannya sendiri dengan mudah dan dengan waktu yang singkat. Sistem dapat menyajikan laporan transaksi perusahaan yang ada dengan cepat dan mengurangi kesalahan yang terjadi. Distributor kain Hoggy Djaya adalah sebuah perusahaan distributor kain yang berada di Surabaya. Perusahaan ini masih menggunakan cara pencatatan laporan transaksi secara manual yang mengakibatkan adanya kesalahan dalam pencatatan transaksi dan kehilangan data transaksi. Dengan menerapkan *web reporting point of sales* diharapkan laporan transaksi perusahaan dapat berjalan dengan baik dan dapat menghasilkan laporan transaksi secara cepat dan tepat sehingga perusahaan dapat lebih berkembang.

Kata kunci: laporan, transaksi, *web reporting*, sistem.

1. Pendahuluan

Hoggy Djaya adalah sebuah perusahaan distributor yang bergerak dalam bidang garment khususnya kain. Untuk mengatasi pencatatan transaksi yang dilakukan, Hoggy Djaya masih mengandalkan sistem manual dengan cara menuliskan seluruh transaksi pada sebuah buku khusus untuk mencatat semua transaksi yang telah dilakukan.

Karena semua transaksi yang dilakukan tersebut di catat secara manual pada sebuah buku, Hoggy Djaya ingin untuk memiliki sebuah sistem yang sudah terkomputerisasi dan efisien untuk mempermudah dalam melihat laporan transaksi yang terjadi pada perusahaan. Selama ini banyak data transaksi yang dicatat secara konvensional menggunakan slip nota dan buku catatan sehingga banyak data transaksi yang terlewatkan dalam pencatatan.

Oleh karena itu, membangun sebuah aplikasi *web reporting point of sales* menjadi sebuah solusi dari permasalahan yang ada pada perusahaan distributor kain Hoggy Djaya dalam melihat laporan transaksi yang terjadi secara cepat dan tepat sehingga tidak ada data transaksi yang terlewatkan dalam pencatatan.

Tujuan akhir dari penulisan ini adalah merancang dan membuat sebuah aplikasi *web reporting point of sales* untuk perusahaan distributor kain Hoggy Djaya dalam melihat laporan transaksi sehingga pemilik perusahaan

dapat melihat seluruh laporan transaksi yang terjadi didalam perusahaannya secara cepat dan tepat.

2. Landasan Teori

2.1 Sistem Informasi

Menurut Jogianto (2005), sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. Sistem juga memiliki beberapa karakteristik sistem seperti berikut:

1. Komponen
2. Batasan Sistem
3. Lingkungan Luar Sistem
4. Masukan Sistem (Input)
5. Keluaran Sistem (Output)
6. Penghubung
7. Tujuan Sistem

2.2 Responsive Web Design (RWD)

Sebuah cara baru dalam pengembangan *front-end* desain sebuah *website* yang menentukan penataan elemen pada *web* yang berbeda pada setiap ukuran layar. Pertama kali konsep ini dicetuskan oleh Ethan Martcotte pada artikelnya yang berjudul "Responsive Web Design" Ethan menuliskan bahwa konsep desain *web responsive* ada tiga faktor utama yaitu:

- a. Kerangka dasar yang fleksibel
- b. Gambar yang sinkron dengan posisi lebar layar
- c. Melalui *media query*, membuat sebuah *web* dapat menyesuaikan dengan ukuran layar. (Peterson, 2014)

2.3 MySQL

MySQL merupakan bahasa pemrograman *open-source* yang paling populer dan banyak digunakan di lingkungan *Linux*. Kepopuleran ini karena ditunjang dengan performansi *query* dari basis datanya yang jarang bermasalah.

MySQL sebenarnya sebuah produk yang berjalan pada platform *Linux*. Karena sifatnya yang *open-source*, MySQL dapat berjalan pada berbagai platform seperti Windows maupun Macintosh. Selain itu MySQL juga bersifat sebagai program pengakses basis data yang bersifat jaringan sehingga dapat digunakan untuk aplikasi *multi-user* (banyak pengguna).

Kelebihan dari MySQL adalah penggunaan bahasa *query* yang dimiliki SQL (*Structured Query Language*). SQL merupakan suatu bahasa permintaan yang terstruktur dan sudah distandarisasi untuk semua program pengakses basis data seperti Oracle, PostgreSQL, termasuk MySQL.

Sebagai sebuah program yang menghasilkan basis data, MySQL tidak dapat berjalan sendiri tanpa aplikasi penghubung antarmuka. Hampir semua program dan aplikasi yang baik mendukung kemampuan MySQL, aplikasi yang *open-source* seperti PHP, maupun aplikasi yang ada di platform pengguna seperti Delphi, Visual Basic, dan lainnya. (Yank, 2012)

2.4 Database Management System (DBMS)

Database Management System, biasa disingkat sebagai DBMS merupakan sebuah perangkat lunak yang khusus dirancang untuk melakukan pengelolaan dan pengkoleksian data dalam jumlah besar. DBMS ini juga dirancang untuk dapat melakukan manipulasi data dengan mudah.

Menurut Abdul Kadir (2001), pengertian DBMS adalah yaitu satu program komputer yang digunakan untuk memasukkan, mengubah, menghapus, memanipulasi dan memperoleh data/informasi yang dilakukan dengan cara praktis dan efisien.

Menurut C.J. Date, DBMS adalah *software* yang menangani seluruh akses pada sebuah basis data untuk melayani kebutuhan pengguna.

Menurut Gordon C. Everest, DBMS merupakan sebuah sistem manajemen yang efektif untuk mengorganisasi sumber daya data.

Dari kedua pernyataan itu, dapat disimpulkan bahwa DBMS adalah semua peralatan komputer yang dilengkapi dengan bahasa yang berorientasi pada data yang sering disebut sebagai bahasa generasi ke-4. (Handiswan, 2014)

2.5 Unified Modelling Language (UML)

Menurut Rosenberg dan Stephens (2007), *Unified Modelling Language* (UML) adalah sebuah bahasa umum yang digunakan untuk merancang, visualisasi, dan

dokumentasi dalam sistem lunak (*software*). Dengan menggunakan UML dapat membuat sebuah model untuk semua jenis sistem lunak, dimana aplikasi tersebut dapat digunakan pada piranti keras (*hardware*), sistem operasi dan jaringan, serta penulisan program dengan bahasa apapun. Untuk dapat memodelkan sistem, UML mendefinisikan diagram-diagram sebagai berikut:

1. Use Case Diagram

Diagram *use case* mempresentasikan sebuah interaksi antara aktor dengan sistem dan lebih menekankan apa yang diperbuat oleh sistem.

2. Class Diagram

Diagram *class* menggambarkan struktur dan deskripsi *class*, *package*, dan objek beserta hubungan satu sama lain. *Class* merupakan implementasi dari sebuah *interface*, yaitu *class* abstrak yang hanya memiliki metoda.

3. Statechart Diagram

Diagram *statechart* menggambarkan perubahan keadaan suatu objek pada sistem akibat dari *stimuli* yang diterima. Pada umumnya diagram *statechart* menggambarkan *class* tertentu.

4. Activity Diagram

Diagram *activity* menggambarkan berbagai macam aktivitas dalam sistem yang sedang dirancang, dari awal aktivitas kemudian ke *decision* yang dapat terjadi, dan bagaimana aktivitas itu berakhir. Aktivitas diagram ini juga dapat menggambarkan proses paralel yang dapat terjadi pada beberapa tindakan.

5. Sequence Diagram

Sequence diagram menggambarkan interaksi antara objek didalam dan disekitar sistem berupa pesan yang digambarkan berdasarkan waktu. *Sequence* diagram dapat digunakan untuk menggambarkan skenario atau tahapan-tahapan yang dilakukan untuk dapat menghasilkan *output* tertentu.

6. Collaboration Diagram

Collaboration diagram juga menggambarkan interaksi antara objek seperti *sequence* diagram tetapi lebih menekankan pada peran masing-masing objek dan bukan pada saat menyampaikan pesan.

7. Component Diagram

Component diagram menggambarkan struktur dan hubungan antar *software* termasuk ketergantungannya.

8. Deployment Diagram

Deployment diagram menggambarkan detail bagaimana komponen di bentuk dalam sebuah infrastruktur sistem.

2.6 Bootstrap

Bootstrap merupakan produk *open source* yang dirilis oleh karyawan Twitter yang dikembangkan oleh Mark Otto dan Jacob Thornton. Sejak Bootstrap diluncurkan pada Agustus 2011, Bootstrap telah diambil dari dalam popularitas.

Bootstrap memungkinkan untuk desain *web* responsif dengan memiliki 12 kolom. Bootstrap menyediakan fitur CSS dan JavaScript sesuai dengan kebutuhan pengguna. (Spurlock, 2013)

Kelebihan yang dimiliki oleh Bootstrap adalah desain responsif dan sistem 12 kolom. Dengan desain responsif maka aplikasi *web* yang didesain dengan menggunakan Bootstrap akan secara otomatis dapat menyesuaikan lebar dari media yang digunakan sehingga tampilan *web* akan menjadi lebih bagus dengan menggunakan media apapun.

3. Analisis dan Rancangan Sistem

3.1 Analisis

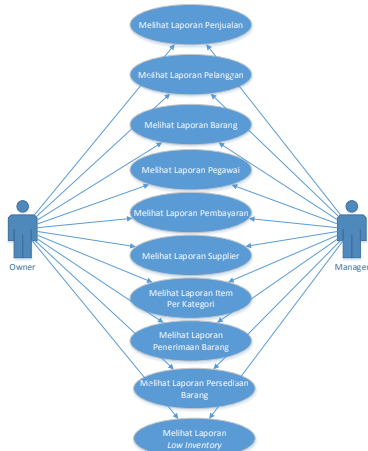
Hoggy Djaya adalah sebuah perusahaan distributor yang bergerak dalam bidang textile. Dengan sistem manual dalam mencatat seluruh transaksi yang dilakukan oleh perusahaan ini sangat menyusahkan dan juga membuang waktu yang lama untuk mencari data-data transaksi yang sudah dilakukan.

Dengan aplikasi *web reporting point of sales* sangat berguna untuk melihat laporan yang ada didalam perusahaan khususnya distributor kain Hoggy Djaya. Seiring berkembangnya sistem informasi, proses pencatatan laporan sudah tidak lagi dilakukan secara manual. Sistem informasi yang terus berkembang dapat membuat sebuah aplikasi *web reporting point of sales* yang lebih bermanfaat bagi distributor kain Hoggy Djaya untuk melihat laporan-laporan secara cepat.

Oleh karena itu, distributor kain “Hoggy Djaya” memilih untuk membuat *web reporting point of sales* yang dapat di akses secara online dengan menggunakan *website* sehingga pemilik distributor kain ini dapat melihat laporan perusahaan dimanapun pemilik perusahaan berada. Hasil laporan perusahaan yang pemilik inginkan untuk perusahaannya, yaitu:

1. Dapat melihat laporan penjualan
2. Dapat melihat laporan pembayaran
3. Dapat melihat laporan barang
4. Dapat melihat laporan persediaan barang

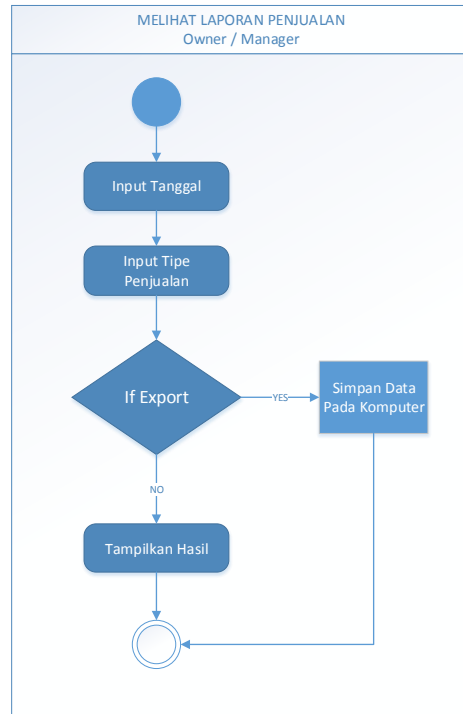
3.2 Rancangan Sistem



Gambar 1. Use Case Diagram

Pemilik dan manajer dapat melihat semua laporan transaksi yang terjadi didalam perusahaan.

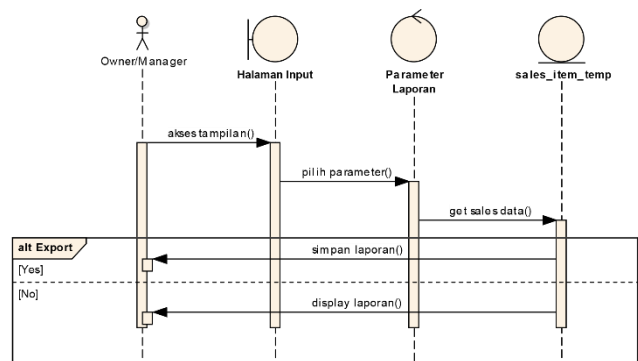
3.3 Activity Diagram



Gambar 2. Activity Diagram Melihat Laporan Penjualan

Untuk melihat laporan transaksi penjualan, pemilik atau manajer harus memasukkan tanggal yang ingin dilihat. Kemudian memasukkan tipe penjualan (penjualan atau pengembalian barang). Jika ingin mengekspor laporan penjualan, klik simpan pada komputer atau klik tidak simpan maka laporan transaksi penjualan akan ditampilkan pada layar komputer.

3.4 Sequence Diagram

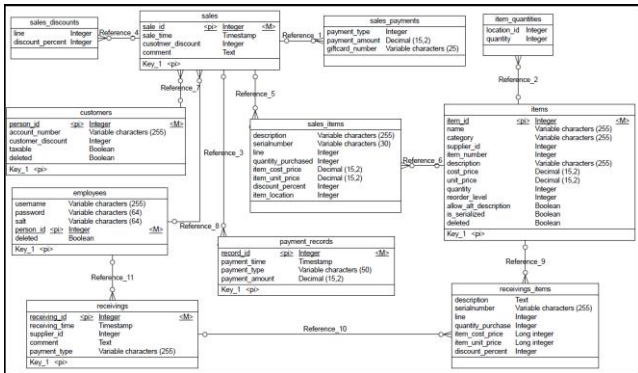


Gambar 3. Sequence Diagram Melihat Laporan Penjualan

Proses untuk melihat laporan transaksi penjualan, pengguna mengakses halaman laporan penjualan. Kemudian memasukkan tanggal dan memilih parameter yang ada. Setelah itu, sistem akan memproses data pada database sesuai dengan parameter yang ada. Jika pengguna

ingin mengeksport laporan, maka laporan penjualan akan tersimpan pada komputer. Tetapi jika pengguna tidak ingin mengeksport laporan, maka laporan penjualan akan ditampilkan pada layar komputer.

3.5 Conceptual Data Model (CDM)

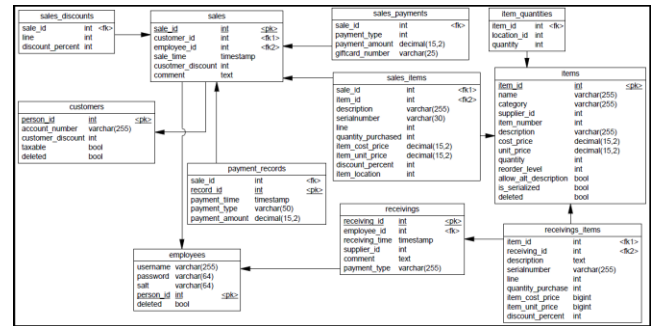


Gambar 4. Conceptual Data Model (CDM)

Penjelasan dengan menggunakan beberapa asumsi untuk conceptual data model:

1. Satu “sales” memiliki banyak “sales_payments”. Oleh karena itu, dibutuhkan relasi *one to many*. (Reference_1)
2. Satu “items” memiliki banyak “item_quantities”. Oleh karena itu, dibutuhkan relasi *one to many*. (Reference_2)
3. Satu “sales” memiliki banyak “payment_recods”. Oleh karena itu, dibutuhkan relasi *one to many*. (Reference_3)
4. Satu “sales” memiliki banyak “sales_discounts”. Oleh karena itu, dibutuhkan relasi *one to many*. (Reference_4)
5. Satu “sales” memiliki banyak “sales_items”. Oleh karena itu, dibutuhkan relasi *one to many*. (Reference_5)
6. Satu “items” memiliki banyak “sales_items”. Oleh karena itu, dibutuhkan relasi *one to many*. (Reference_6)
7. Satu “customers” memiliki banyak “sales”. Oleh karena itu, dibutuhkan relasi *one to many*. (Reference_7)
8. Satu “sales” memiliki banyak “payments_records”. Oleh karena itu, dibutuhkan relasi *one to many*. (Reference_8)
9. Satu “items” memiliki banyak “receivings_items”. Oleh karena itu, dibutuhkan relasi *one to many*. (Reference_9)
10. Satu “receivings” memiliki banyak “receivings_items”. Oleh karena itu, dibutuhkan relasi *one to many*. (Reference_10)
11. Satu “employees” memiliki banyak “receivings”. Oleh karena itu, dibutuhkan relasi *one to many*. (Reference_11)

3.6 Physical Data Model (PDM)



Gambar 5. Physical Data Model (PDM)

Penjelasan atribut untuk *Physical Data Model*:

1. Tabel “sales” memiliki atribut “sale_id” sebagai *Primary Key* (PK) yang dipakai untuk dapat mengidentifikasi seluruh data yang diproses.
2. Tabel “sales_discounts” dan “sales_payments” memiliki atribut “sale_id” sebagai *Foreign Key* (FK) yang dipakai untuk memproses diskon pada “sales”.
3. Tabel “customers” memiliki atribut “person_id” sebagai *Primary Key* (PK) yang dipakai untuk dapat diidentifikasi oleh tabel “sales” yang memiliki atribut “customer_id” sebagai *Foreign Key* (FK) yang digunakan untuk menghubungkan data customers.
4. Tabel “employees” memiliki atribut “person_id” sebagai *Primary Key* (PK) yang dipakai untuk dapat diidentifikasi oleh tabel “sales” yang memiliki atribut “employee_id” sebagai *Foreign Key* (FK) yang digunakan untuk menghubungkan data pegawai.
5. Tabel “payment_records” memiliki atribut “record_id” sebagai *Primary Key* (PK) yang dipakai sebagai data utama untuk pencatatan transaksi yang dihubungkan melalui atribut “sale_id” sebagai *Foreign Key* (FK).
6. Tabel “items” memiliki atribut “item_id” sebagai *Primary Key* (PK) yang dipakai untuk dapat menghubungkan data dengan tabel “item_quantities” yang menggunakan atribut “item_id” sebagai *Foreign Key* (FK).
7. Tabel “sales_items” memiliki atribut “sale_id” dan “item_id” sebagai *Foreign Key* (FK) yang dipakai untuk menghubungkan antara transaksi yang terjadi dengan persediaan barang yang ada.
8. Tabel “receivings_items” memiliki atribut “item_id” sebagai *Foreign Key* (FK) yang digunakan untuk mengidentifikasi barang.
9. Tabel “receivings” memiliki atribut “receivings_items” sebagai *Foreign Key* (FK) yang digunakan untuk membuat laporan penerimaan barang.
10. Tabel “receivings” memiliki atribut “employee_id” sebagai *Foreign Key* (FK) yang

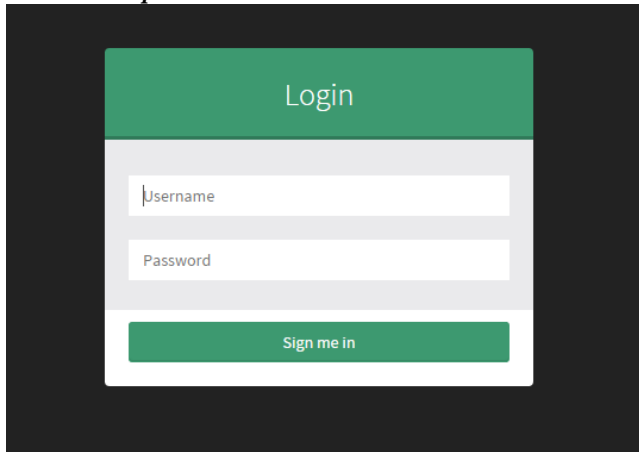
digunakan untuk mengidentifikasi pelaku pembuat transaksi.

4. Implementasi Sistem

4.1 Implementasi

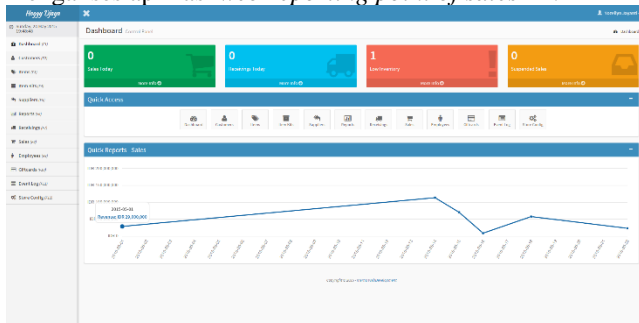
Pada proses implementasi yang dilakukan melalui tahapan-tahapan yang ada. Tahapan awal dengan menawarkan pembuatan aplikasi *web reporting point of sales* agar dapat melihat laporan transaksi perusahaan untuk distributor kain Hoggy Djaya. Kemudian dilakukan analisa terhadap kebutuhan dari perusahaan distributor kain Hoggy Djaya dan menawarkan solusi yang dapat mengatasi masalah yang ada didalam perusahaan. Setelah klien menyetujui penawaran yang sudah diajukan, maka selanjutnya dilakukan desain sistem terlebih dahulu, membuat *project requirement specification* (PRS), dan pengembangan sistem. Melalui desain sistem dan PRS yang telah dibuat, maka implementasi akan dilakukan pengembangan sistem sesuai dengan fitur-fitur yang ada didalam desain sistem dan PRS. Untuk memastikan aplikasi dapat berjalan dengan baik, perlu digunakan *user acceptance test* agar lebih memastikan aplikasi dapat berjalan dengan baik.

4.2 Sistem Aplikasi



Gambar 6. Tampilan Halaman Login

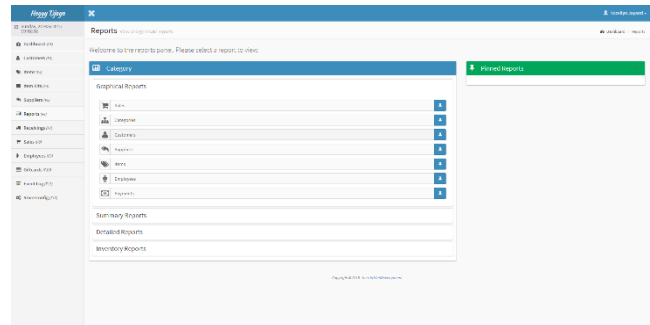
Pada halaman login ini, pengguna harus memasukkan username dan password yang dimiliki untuk dapat mengakses aplikasi *web reporting point of sales* ini.



Gambar 7. Tampilan Halaman Utama

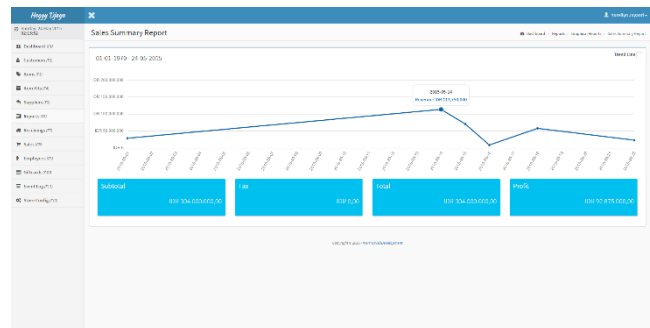
Halaman ini akan ditampilkan setelah pengguna

berhasil login dengan username dan password yang dimiliki kemudian pengguna dapat menggunakan aplikasi ini.



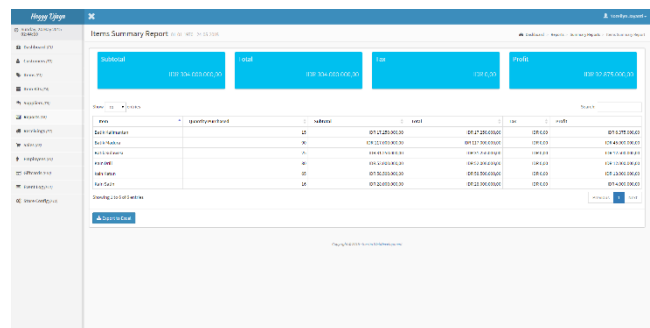
Gambar 8. Tampilan Halaman Menu Laporan

Setelah pengguna berhasil login dan dapat mengakses aplikasi, pengguna dapat melihat laporan transaksi perusahaan pada menu laporan dengan berbagai macam jenis laporan yang ditampilkan.



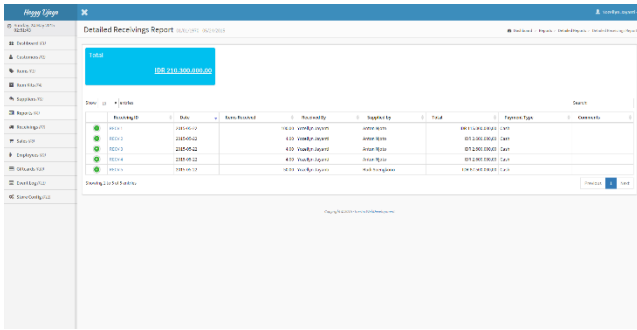
Gambar 9. Tampilan Halaman Melihat Laporan Secara Grafik

Pada halaman ini berisi informasi laporan transaksi yang terjadi didalam perusahaan yang ditampilkan pada pengguna dalam bentuk grafik.



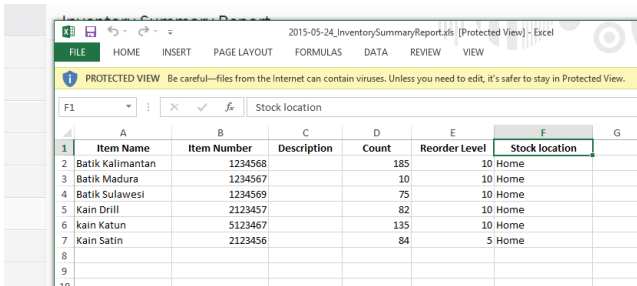
Gambar 10. Tampilan Halaman Melihat Laporan Ringkas

Pada halaman ini berisi informasi laporan transaksi yang terjadi didalam perusahaan yang ditampilkan pada pengguna secara ringkas dengan tabel.



Gambar 11. Tampilan Halaman Melihat Laporan Detail

Pada halaman ini berisi informasi laporan transaksi yang terjadi didalam perusahaan yang ditampilkan pada pengguna dengan detail dalam bentuk tabel.



Gambar 12. Hasil Ekspor Data Laporan Transaksi

Laporan yang berhasil diekspor dan disimpan pada computer pengguna jika file ekspor dibuka akan menghasilkan data dalam bentuk excel (Gambar 12) secara jelas sesuai dengan data yang ada.

4.3 User Acceptance Test (UAT)

User Acceptance Test (UAT) adalah tahap terakhir setelah aplikasi selesai dirancang untuk menguji apakah sistem dapat berjalan dengan baik sesuai dengan fitur-fitur yang ada didalam PRS oleh pengguna. Untuk fitur melihat laporan perusahaan hanya pemilik dan manajer yang dapat mengakses fitur tersebut.

Tabel 1. Tabel UAT Melihat Laporan Secara Grafik

| No. | Area | Keterangan | Hasil yang diharapkan | Hasil yang didapatkan |
|-----|------------------|--|---------------------------------------|--------------------------------------|
| 1. | <i>Sales</i> | Melihat laporan penjualan dalam bentuk grafik | Grafik dapat terlihat, Laporan sesuai | Grafik terlihat dengan grafik garis |
| 2. | <i>Category</i> | Melihat laporan barang yang terjual berdasarkan kategori dalam bentuk grafik | Grafik dapat terlihat, Laporan sesuai | Grafik terlihat dengan grafik pie |
| 3. | <i>Customers</i> | Melihat laporan pembelian pelanggan dalam bentuk grafik | Grafik dapat terlihat, Laporan sesuai | Grafik terlihat dengan grafik batang |
| 4. | <i>Suppliers</i> | Melihat laporan pembelian barang dari supplier dalam bentuk grafik | Grafik dapat terlihat, Laporan sesuai | Grafik terlihat dengan grafik pie |
| 5. | <i>Items</i> | Melihat laporan penjualan barang berdasarkan nama barang dalam bentuk grafik | Grafik dapat terlihat, Laporan sesuai | Grafik terlihat dengan grafik garis |
| 6. | <i>Employees</i> | Melihat laporan penjualan pegawai dalam bentuk grafik | Grafik dapat terlihat, Laporan sesuai | Grafik terlihat dengan grafik pie |
| 7. | <i>Payments</i> | Melihat laporan pembayaran transaksi penjualan dalam bentuk grafik | Grafik dapat terlihat, Laporan sesuai | Grafik terlihat dengan grafik pie |

Tabel 2. Tabel UAT Melihat Laporan Secara Ringkas

| No. | Area | Keterangan | Hasil yang diharapkan | Hasil yang didapatkan |
|-----|------------------|---|--|---|
| 1. | <i>Sales</i> | Melihat laporan penjualan secara ringkas | Laporan penjualan dapat dilihat dan diekspor | Laporan dapat diekspor dan file dapat dibuka. Laporan penjualan dapat dilihat secara jelas. |
| 2. | <i>Category</i> | Melihat laporan barang yang terjual berdasarkan kategori secara ringkas | Laporan barang per kategori dapat dilihat dan diekspor | Laporan dapat diekspor dan file dapat dibuka. Laporan barang per kategori dapat dilihat dengan jelas. |
| 3. | <i>Customers</i> | Melihat laporan pembelian pelanggan secara ringkas | Laporan pelanggan dapat dilihat dan diekspor | Laporan dapat diekspor dan file dapat dibuka. Laporan pelanggan dapat dilihat dengan jelas. |
| 4. | <i>Suppliers</i> | Melihat laporan pembelian barang dari supplier secara ringkas | Laporan supplier dapat dilihat dan diekspor | Laporan dapat diekspor dan file dapat dibuka. Laporan supplier dapat dilihat dengan jelas. |
| 5. | <i>Items</i> | Melihat laporan penjualan barang berdasarkan nama barang secara ringkas | Laporan barang dapat dilihat dan diekspor | Laporan dapat diekspor dan file dapat dibuka. Laporan barang dapat dilihat dengan jelas. |
| 6. | <i>Employees</i> | Melihat laporan penjualan pegawai secara ringkas | Laporan penjualan pegawai dapat dilihat dan diekspor | Laporan dapat diekspor dan file dapat dibuka. Laporan penjualan pegawai dapat dilihat dengan jelas. |
| 7. | <i>Payments</i> | Melihat laporan pembayaran transaksi penjualan secara ringkas | Laporan pembayaran dapat dilihat dan diekspor | Laporan dapat diekspor dan file dapat dibuka. Laporan pembayaran dapat dilihat dengan jelas. |

Tabel 3. Tabel UAT Melihat Laporan Secara Detail

| No. | Area | Keterangan | Hasil yang diharapkan | Hasil yang didapatkan |
|-----|-------------------|--|---|---|
| 1. | <i>Sales</i> | Melihat laporan penjualan secara detail | Laporan penjualan ditampilkan dalam bentuk tabel secara detail | Laporan penjualan yang dapat dilihat dalam bentuk tabel |
| 2. | <i>Receivings</i> | Melihat laporan penerimaan barang secara detail | Laporan penerimaan barang berbentuk tabel secara detail | Laporan penerimaan barang dapat dilihat secara detail berbentuk tabel |
| 3. | <i>Customers</i> | Melihat laporan pembelian pelanggan secara detail | Laporan pelanggan ditampilkan secara detail dalam bentuk tabel | Laporan pelanggan dilihat dalam bentuk tabel secara detail |
| 4. | <i>Items</i> | Melihat laporan penjualan barang berdasarkan nama barang secara detail | Laporan penjualan barang ditampilkan secara detail dengan menggunakan tabel | Laporan penjualan barang dapat dilihat dalam bentuk tabel secara detail |
| 5. | <i>Employees</i> | Melihat laporan penjualan oleh pegawai secara detail | Laporan penjualan oleh pegawai ditampilkan dalam bentuk tabel secara detail | Laporan penjualan oleh pegawai dapat dilihat secara detail dengan tabel |

Tabel 4. Tabel UAT Melihat Laporan Inventory

| No. | Area | Keterangan | Hasil yang diharapkan | Hasil yang didapatkan |
|-----|--------------------------|---|--|--|
| 1. | <i>Low Inventory</i> | Melihat laporan persediaan barang rendah dan laporan dapat diekspor | Laporan dapat dilihat, Laporan dapat diekspor | Laporan dapat dilihat dengan jelas. Hasil ekspor dapat dibuka dengan <i>ms-excel</i> |
| 2. | <i>Inventory Summary</i> | Melihat laporan persediaan barang yang ada | Laporan persediaan barang dapat dilihat dan dapat diekspor | Laporan dapat dilihat dengan jelas. Hasil ekspor dapat dibuka dengan <i>ms-excel</i> |

4.4 Kesimpulan dan Saran

4.4.1 Kesimpulan

Setelah melakukan implementasi dan *user acceptance test* terhadap aplikasi *web reporting point of sales* pada distributor Hoggy Djaya, maka dapat ditarik beberapa kesimpulan sebagai berikut:

1. Implementasi aplikasi *web reporting point of sales* membantu pihak distributor kain Hoggy Djaya untuk dapat melihat laporan transaksi yang terjadi didalam perusahaan secara cepat dan tepat
2. Aplikasi berbasis *website* cocok diterapkan pada distributor kain Hoggy Djaya karena pemilik perusahaan dapat melihat laporan transaksi dengan mudah meskipun tidak berada di tempat kerja
3. Fitur-fitur untuk melihat laporan transaksi berfungsi dengan baik dan sesuai dengan kebutuhan perusahaan
4. Data laporan transaksi yang diperoleh sesuai dengan transaksi yang tercatat dalam *database*

4.4.2 Saran

Berdasarkan hasil dari implementasi dan *user acceptance test* terhadap kebutuhan akan aplikasi *web reporting point of sales* untuk perusahaan distributor kain Hoggy Djaya, penulis dapat memberikan saran-saran yang bisa bermanfaat sebagai berikut:

1. *User Interface* dapat lebih dioptimalkan
2. Perbaikan dalam kecepatan loading
3. Fitur-fitur dapat lebih dikembangkan lagi
4. Dapat digunakan sebagai aplikasi untuk bisnis pemula
5. Dapat digunakan sebagai pedoman untuk pengembangan aplikasi

DAFTAR PUSTAKA

- Handiswan. (2014). Basis Data DBMS (Data Base Management System). *Makalah*.
- Jogianto. (2005). *Sistem Teknologi Informasi*. Yogyakarta: Andi.
- Kadir, A. (2003). *Pengenalan Sistem Informasi*. Yogyakarta: Andi.
- Peterson, C. (2014). *Learning Responsive Web Design*. Canada: O'Reilly Media, Inc.
- Putra, E. S. (2014). Sistem Informasi. *Makalah Pengantar Teknologi Informasi*.
- Rosenberg, D., & Stephens, M. (2007). *Use Case Driven Object Modeling with UML*. United States of America: Apress.
- Spurlock, J. (2013). *Bootstrap*. United States of America: O'Reilly Media, Inc.
- Swastika, W. (2006). *PHP 5 dan MySQL 4*. Jakarta: Dian Rakyat.
- Upton, D. (2007). *CodeIgniter for Rapid PHP Application Development*. Birmingham-Mumbai: Packt Publishing.
- Yank, K. (2012). *PHP & MYSQL NOVICE TO NINJA* (5th ed.). (K. Steele, Ed.) United States of America: sitepoint.