**APLIKASI PENGELOLAAN DATA TRANSAKSI PENJUALAN DAN STOK BARANG BERBASIS DESKTOP DI TOKO NANDO SPORT CENTRE CIANJUR**

**KERJA PRAKTEK**

Diajukan Sebagai Salah Satu Syarat Kelulusan Mata Kuliah Kerja Praktek Program Studi Teknik Informatika Fakultas Teknik

Universitas Suryakancana Cianjur



**OLEH**

**MUFTI AKHMAD RILMANSYAH**

**5520116131**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS TEKNIK**

**UNIVERSITAS SURYAKANCANA**

**CIANJUR**

**2019**

# **LEMBAR PERSETUJUAN**

**LAPORAN KERJA PRAKTEK**

|  |  |  |
| --- | --- | --- |
| Nama  NPM  Program Studi  Judul | :  :  :  : | MUFTI AKHMAD RILMANSYAH  5520116131  TEKNIK INFORMATIKA  **APLIKASI PENGELOLAAN DATA TRANSAKSI PENJUALAN DAN STOK BARANG BERBASIS DESKTOP DI TOKO NANDO SPORT CENTRE CIANJUR** |

Telah disetujui untuk disidangkan pada Seminar Kerja Praktek pada Program Sarjana Strata-1 (S1), Program Studi Teknik Informatika Fakultas Teknik Universitas Suryakancana.

Cianjur, 27 April 2019

|  |  |
| --- | --- |
| Pembimbing Kerja Praktek  **Hasbu Naim S., S. Kom., M. Kom.**  **NIDN : 0415018402** | Pembimbing Lapangan  **Vincensius Aditya** |

# **LEMBAR PENGESAHAN**

**LAPORAN KERJA PRAKTEK**

|  |  |  |
| --- | --- | --- |
| Nama  NPM  Program Studi  Judul | :  :  :  : | MUFTI AKHMAD RILMASNYAH  5520116131  TEKNIK INFORMATIKA  **APLIKASI PENGELOLAAN DATA TRANSAKSI PENJUALAN DAN STOK BARANG BERBASIS DESKTOP DI TOKO NANDO SPORT CENTRE CIANJUR** |

Telah disidangkan dan dinyatakan Lulus Seminar Kerja Praktek pada Program Sarjana Strata-1 (S1), Program Studi Teknik Informatika Fakultas Teknik Universitas Suryakancana pada tanggal .. .. 2019.

|  |  |
| --- | --- |
| Nama Penguji | Tanda Tangan |

1. Hasbu Naim S, S. Kom., M. Kom.
2. Siti Nazilah, ST., M. Kom.

Mengetahui :

Ketua Program Studi

Teknik Informatika

**Ai Musrifah, ST., M. Kom.**

**NIDN : 0403018001**

# **LEMBAR PERNYATAAN KEASLIAN**

**LAPORAN KERJA PRAKTEK**

|  |  |  |
| --- | --- | --- |
| Nama  NPM  Program Studi  Judul | :  :  :  : | MUFTI AKHMAD RILMANSYAH  5520116131  TEKNIK INFORMATIKA  **APLIKASI PENGELOLAAN DATA TRANSAKSI PENJUALAN DAN STOK BARANG BERBASIS DESKTOP DI TOKO NANDO SPORT CENTRE CIANJUR** |

Dengan ini saya menyatakan bahwa dalam laporan penelitian ini tidak terdapat karya yang pernah diajukan untuk memperoleh **kelulusan Kerja Praktek ataupun gelar kesarjanaan** di suatu Perguruan Tinggi, dan sepanjang sepengetahuan saya juga tidak terdapat karya atau pendapat yang pernah di tulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka.

Cianjur, 27 April 2019

**Mufti Akhmad Rilmansyah**

**5520116131**

**APLIKASI PENGELOLAAN DATA TRANSAKSI PENJUALAN DAN STOK BARANG BERBASIS DESKTOP DI TOKO NANDO SPORT CENTRE CIANJUR**

**MUFTI AKHMAD RILMANSYAH**

**5520116131**

**Teknik Informatika**

# **ABSTRAK**

Nando Sport Centre merupakan sebuah toko yang bergerak di bidang penjualan barang-barang olah raga yang berada di Jl. Ps. Baru, Muka Kecamatan Cianjur. Toko tersebut menjual berbagai macam perlengkapan olah raga mulai sepatu, baju, celana, jaket dan perlengkapan olahraga lainnya. Nando Sport Centre saat menjalankan kegiatan sehari-harinya seperti pencatatan transaksi penjualan, pencatatan transaksi pembelian, pencarian barang, pengelolaan stok, dan pembuatan laporan-laporan masih menggunakan cara-cara manual yaitu mencatat didalam sebuah buku sehingga memerlukan waktu yang cukup lama dan resiko kesalahan manusia (*human error*) yang cukup besar.

Pembuatan aplikasi yang digunakan adalah menggunakan metode rekayasa perangkat lunak paradigma model *waterfall* dengan UML (*unified modeling language*) dengan alat yang digunakan untuk merancang yaitu *Usecase* Diagram*, Activity* Diagram*, Class* Diagram*, State* Diagramdan *Sequence* Diagram. Sedangkan alat pembangunan aplikasi *database* menggunakan PhpMyAdmin dengan bahasa pemograman DELPHI.

Aplikasi yang dirancang mempunyai kelebihan dalam membantu *user* untuk pengelolaan data transaksi penjualan, pengelolaan data transaksi pembelian, pencarian barang, pengelolaan stok, dan pengelolaan laporan.

Kata Kunci :*Desktop,* Pengelolaan Data Transaksi Penjualan dan Stok Barang, *waterfall,* Delphi.

# **KATA PENGANTAR**

Alhamdulillah, segala puji dan syukur ke hadirat Allah SWT, karena dengan ridho dan karunia-Nya sehingga penulis dapat menyelesaikan laporan Kerja praktek yang berjudul **“Aplikasi Pengelolaan Data Transaksi Penjualan Dan Stok Barang Berbasis Desktop Di Toko Nando Sport Centre Cianjur“**.

Laporan ini merupakan salah satu persyaratan kelulusan pada mata kuliah Kerja Praktek di Program Studi Teknik Informatika di Fakultas Teknik Universitas Suryakancana Cianjur.

Penelitian dan penyusunan laporan Kerja Praktek ini adalah suatu proses yang tidak lepas dari bantuan, bimbingan, serta dukungan berbagai pihak. Untuk itu penulis mengucapkan rasa terima kasih dan penghargaan yang sebesar-besarnya kepada :

1. Yang tercinta kepada kedua orang tua saya, yang telah memberikan do’a dan dorongan kepada saya baik berupa moril maupun materil dalam menyelesaikan studi ini.
2. Ibu Ai Musrifah, ST., M. Kom. selaku Ketua Program Studi Teknik Informatika.
3. Bapak Hasbu Naim S., S. Kom., M. Kom. selaku dosen pembimbing yang meluangkan waktu untuk memberikan petunjuk, arahan, motivasi, serta ilmu beliau dalam menyelesaikan Laporan Kerja Praktek ini.
4. Bapak selaku pembimbing lapangan beserta jajarannya tempat penulis melakukan penelitian.
5. Rekan-rekan terdekat yang senantiasa selalu memberikan dukungan serta semangat selama proses pengerjaan Kerja Praktek ini.
6. Rekan-rekan Informatika khususnya IF’D 2016 yang selalu memberikan dukungan dan semangat.

# **DAFTAR ISI**

[**LEMBAR PERSETUJUAN** i](#_Toc518073856)

[**LEMBAR PENGESAHAN** ii](#_Toc518073857)

[**LEMBAR PERNYATAAN KEASLIAN** iii](#_Toc518073858)

[**ABSTRAK** iv](#_Toc518073859)

[**KATA PENGANTAR** v](#_Toc518073860)

[**DAFTAR ISI** vi](#_Toc518073861)

[**DAFTAR TABEL** x](#_Toc518073862)

[**DAFTAR GAMBAR** xi](#_Toc518073863)

[**DAFTAR SIMBOL USE CASE DIAGRAM** xv](#_Toc518073864)

[**DAFTAR SIMBOL SWIMLANE DIAGRAM** xvi](#_Toc518073865)

[**DAFTAR SIMBOL CLASS DIAGRAM** xviii](#_Toc518073866)

[**DAFTAR SIMBOL SEQUENCE DIAGRAM** xx](#_Toc518073867)

[**BAB I PENDAHULUAN** 21](#_Toc518073868)

[**1.1** **Latar Belakang Masalah** 21](#_Toc518073869)

[**1.2** **Identifikasi Masalah** 22](#_Toc518073870)

[**1.3** **Maksud Dan Tujuan** 22](#_Toc518073871)

[**1.4** **Batasan Masalah** 23](#_Toc518073872)

[**1.5** **Metode Penelitian** 23](#_Toc518073873)

[**1.6** **Jadwal Penelitian** 27](#_Toc518073874)

[**1.7** **Sistematika Penulisan** 28](#_Toc518073875)

[**BAB II TINJAUAN PUSTAKA** 29](#_Toc518073876)

[**2.1** **Tinjauan Pustaka** 29](#_Toc518073877)

[**2.1.1** **Sejarah Singkat Nando Sport Centre Cianjur** 29](#_Toc518073878)

[**2.1.2** **Struktur Organisasi** 29](#_Toc518073879)

[**2.1.3** **Deskripsi Kerja** 30](#_Toc518073880)

[**2.2** **Landasan Teori** 31](#_Toc518073881)

[**2.2.1** **Data** 31](#_Toc518073882)

[**2.2.2** **Pengelolaan Data** 32](#_Toc518073883)

[**2.2.3** **Transaksi** 32](#_Toc518073884)

[**2.2.4** **Penjualan** 32](#_Toc518073884)

[**2.2.5** **Persediaan Barang** 33](#_Toc518073884)

[**2.2.6** **Pengertian Aplikasi** 33](#_Toc518073884)

[**2.2.7** **Metode *Waterfall* Menurut Roger S. Pessman** 33](#_Toc518073885)

[**2.2.8** **Konsep Dasar Aplikasi *Desktop*** 34](#_Toc518073886)

[**2.2.9** **Konsep Dasar Sistem** 35](#_Toc518073887)

[**2.2.10** **Konsep Dasar Informasi** 35](#_Toc518073888)

[**2.2.11** **Konsep Dasar Sistem Informasi** 36](#_Toc518073889)

[**2.2.12** **Konsep Dasar *Desktop*** 38](#_Toc518073890)

[**2.2.13** **Keunggulan Aplikasi Berbasis *Desktop*** 39](#_Toc518073891)

[**2.2.14** **Konsep Dasar *Database*** 39](#_Toc518073892)

[**2.2.15** **Konsep Dasar DBMS (*Database Management System*)** 39](#_Toc518073893)

[**2.2.16** **Pengujian *Black Box*** 40](#_Toc518073894)

[**2.2.17** **Unified Modelling Language** 40](#_Toc518073895)

[**2.3** ***Tool-Tool* yang Digunakan** 47](#_Toc518073896)

[**2.3.1** **MySQL** 47](#_Toc518073897)

[**2.3.2** **XAMPP** 48](#_Toc518073898)

[**2.3.3** **PhpMyAdmin** 48](#_Toc518073899)

[**2.3.4** **Delphi XE8** 49](#_Toc518073901)

[**2.3.5** **Visual Paradigm** 51](#_Toc518073902)

[**2.3.6** **Balsamiq Mockuop** 52](#_Toc518073903)

[**2.3.7** **MySQL *Driver* ODBC** 53](#_Toc518073904)

[**BAB III ANALISIS DAN PERANCANGAN** 54](#_Toc518073905)

[**3.1** **Analisa Kebutuhan** 54](#_Toc518073906)

[**3.1.1** **Analisis Sistem** 54](#_Toc518073907)

[**3.1.2** **Analisis Masalah** 54](#_Toc518073908)

[**3.1.3** **Analisis Sistem yang Sedang Berjalan** 54](#_Toc518073909)

[**3.1.4** **Analisis Aliran Informasi** 58](#_Toc518073910)

[**3.1.4.1** **Analisis Dokumen** 58](#_Toc518073911)

[**3.1.5** **Analisis Kebutuhan Non Fungsional** 59](#_Toc518073914)

[**3.1.5.1** **Analisis Perangkat Keras** 59](#_Toc518073914)

[**3.1.5.2** **Analisis Perangkat Lunak** 59](#_Toc518073912)

[**3.1.5.3** **Analisis Pengguna** 60](#_Toc518073913)

[**3.1.6** **Analisis Kebutuhan Fungsional** 61](#_Toc518073914)

[**3.2** **Perancangan Sistem** 62](#_Toc518073915)

[**3.2.1** **Daftar *Use case*** 62](#_Toc518073916)

[**3.2.2** ***Use Case* Diagram** 62](#_Toc518073917)

[**3.2.3** ***Use Case Scenario*** 64](#_Toc518073918)

[**3.2.4** ***Activity* Diagram** 72](#_Toc518073928)

[**3.2.5** ***Class* Diagram** 81](#_Toc518073938)

[**3.2.6** ***State* Diagram** 82](#_Toc518073939)

[**3.2.7** ***Sequence* Diagram** 86](#_Toc518073946)

[**3.3** **Perancangan *Database* dan Struktur Tabel** 98](#_Toc518073957)

[**3.3.1** **Model *Entity Relationship* Diagram** 98](#_Toc518073958)

[**3.3.2** **Strruktur Tabel** 99](#_Toc518073959)

[**3.4** **Perancangan Struktur Menu** 101](#_Toc518073966)

[**3.5** **Perancangan Antar Muka** 102](#_Toc518073968)

[**3.6** **Perancangan Hasil Cetak** 119](#_Toc518073989)

[**3.7** **Perancangan Pesan** 126](#_Toc518073993)

[**BAB IV IMPLEMENTASI DAN PENGUJIAN** 131](#_Toc518074002)

[**4.1** **Implementasi** 131](#_Toc518074003)

[**4.1.1** **Kebutuhan Perangkat Keras Pembangun** 131](#_Toc518074004)

[**4.1.2** **Kebutuhan Perangkat Lunak Pembangun** 131](#_Toc518074005)

[**4.1.3** **Implementasi Aplikasi** 132](#_Toc518074006)

[**4.1.4** ***Souce Code Program*** 150](#_Toc518074038)

[**4.2** **Pengujian** 150](#_Toc518074039)

[**4.2.1 Kasus Dan Hasil Pengujian** 151](#_Toc518074040)

[**BAB V PENUTUP** 158](#_Toc518074041)

[**5.1** **Simpulan** 158](#_Toc518074042)

[**5.2** **Saran** 158](#_Toc518074043)

[**DAFTAR PUSTAKA** 159](#_Toc518074044)

[**DAFTAR RIWAYAT HIDUP** 160](#_Toc518074045)

[**DAFTAR LAMPIRAN** 161](#_Toc518074046)

# **DAFTAR TABEL**

[Tabel 1.1 Perbandingan Model Waterfall dengan Model Prototyping 26](#_Toc518074244)

[Tabel 1.2 Jadwal Penelitian 27](#_Toc518074245)

[Tabel 3.1 Hasil Analisis Dokumen 59](#_Toc518074246)

[Tabel 3.2 Tabel Kebutuhan Non Fungsional 60](#_Toc518074247)

[Tabel 3.3 Tabel Kebutuhan Fungsional 60](#_Toc518074247)

[Tabel 3.4 Daftar Use Case 62](#_Toc518074248)

[Tabel 3.5 *Use Case Scenario* (U1) : *Login* 64](#_Toc518074249)

[Tabel 3.6 *Use Case Scenario* (U2) : Kelola Master Data Barang 65](#_Toc518074250)

[Tabel 3.7 *Use Case Scenario* (U3) : Kelola Data Transaksi 67](#_Toc518074251)

[Tabel 3.8 *Use Case Scenario* (U4) : Kelola Stok Barang 70](#_Toc518074252)

[Tabel 3.9 *Use Case Scenario* (U5) : Kelola Laporan 71](#_Toc518074253)

[Tabel 3.10 *Use Case Scenario* (U6) :*Logout* 72](#_Toc518074254)

[Tabel 3.11 Tabel *User* 99](#_Toc518074258)

[Tabel 3.12 Tabel Barang 99](#_Toc518074259)

[Tabel 3.13 Tabel Djual 100](#_Toc518074260)

[Tabel 3.14 Tabel Jual 100](#_Toc518074261)

[Tabel 3.15 Tabel DBeli 100](#_Toc518074262)

[Tabel 3.16 Tabel Beli 101](#_Toc518074263)

[Tabel 4.1 Daftar Perangkat Keras Pembangun 131](#_Toc518074264)

[Tabel 4.2 Daftar Perangkat Lunak Pembangun 132](#_Toc518074265)

[Tabel 4.3. Hasil Pengujian 151](#_Toc518074266)

# **DAFTAR GAMBAR**

[Gambar 1. 1 Model Waterfall 24](#_Toc518074267)

[Gambar 2.1 Struktur Organisasi Nando Sport Centre Cianjur 30](#_Toc518074268)

[Gambar 2.2 Contoh *Use Case* Diagram 41](#_Toc518074269)

[Gambar 2.3 Contoh *Class* Diagram 42](#_Toc518074269)

[Gambar 2.4 Contoh *Statechart* Diagram 42](#_Toc518074269)

[Gambar 2.5 Contoh *Activity* Diagram 42](#_Toc518074269)

[Gambar 2.6 Contoh *Sequence* Diagram 43](#_Toc518074269)

[Gambar 2.7 Contoh *Collaboration* Diagram 44](#_Toc518074269)

[Gambar 2.8 Contoh *Component* Diagram 44](#_Toc518074269)

[Gambar 2.9 Contoh *Deployment* Diagram 45](#_Toc518074269)

[Gambar 2.10 Contoh Simbol *Use Case* Diagram Dalam UML 46](#_Toc518074269)

[Gambar 2.11 Logo XAMPP 48](#_Toc518074270)

[Gambar 2.12 Logo PhpMyAdmin 49](#_Toc518074271)

[Gambar 2.13 Logo Delphi RAD Studio XE8 51](#_Toc518074272)

[Gambar 2.14 Visual Paradigm For UML 52](#_Toc518074273)

[Gambar 2.15 Logo Balsamiq Mockup 3 53](#_Toc518074274)

[Gambar 3.1 *Activity* Diagram Sistem yang sedang Berjalan Barang Masuk 57](#_Toc518074275)

[Gambar 3.2 *Activity* Diagram Sistem yang sedang Berjalan Barang Keluar 57](#_Toc518074275)

[Gambar 3.3 *Activity* Diagram Sistem yang sedang Berjalan Transaksi Penjualan 58](#_Toc518074275)

[Gambar 3.4 *Use Case* Diagram APATIPS 63](#_Toc518074276)

[Gambar 3.5 *Activity* Diagram (U1) : *Login* 73](#_Toc518074277)

[Gambar 3.6 *Activity* Diagram (U2) : Kelola Master Data Barang 74](#_Toc518074278)

[Gambar 3.7 *Activity* Diagram (U3) : Kelola Data Transaksi 76](#_Toc518074279)

[Gambar 3.8 *Activity* Diagram (U4) : Kelola Stok Barang 78](#_Toc518074280)

[Gambar 3.9 *Activity* Diagram (U5) : Kelola Laporan 80](#_Toc518074281)

[Gambar 3.10 *Activity* Diagram (U6) : *Logout* 81](#_Toc518074282)

[Gambar 3.11 *Class* Diagram APATIPS 82](#_Toc518074286)

[Gambar 3.12 *State* Diagram *Login* 83](#_Toc518074287)

[Gambar 3.13 *State* Diagram Kelola Master Data Barang 83](#_Toc518074288)

[Gambar 3.14 *State* Diagram Kelola Data Transaksi 84](#_Toc518074289)

[Gambar 3.15 *State* Diagram Kelola Stok Barang 85](#_Toc518074290)

[Gambar 3.16 *State* Diagram Kelola Laporan 86](#_Toc518074291)

[Gambar 3.17 *Sequence* Diagram *Login* 87](#_Toc518074293)

[Gambar 3.18 *Sequence* Diagram Kelola Master Data Barang 88](#_Toc518074294)

[Gambar 3.19 *Sequence* Diagram Kelola Data Transaksi Penjualan 90](#_Toc518074295)

[Gambar 3.20 *Sequence* Diagram Kelola Data Transaksi Pembelian 92](#_Toc518074296)

[Gambar 3.21 *Sequence* Diagram Kelola Stok Barang Keseluruhan 94](#_Toc518074297)

[Gambar 3.22 *Sequence* Diagram Kelola Stok Barang Habis dan Minim 95](#_Toc518074298)

[Gambar 3.23 *Sequence* Diagram Kelola Laporan 96](#_Toc518074299)

[Gambar 3.24 *Sequence* Diagram *Logout* 97](#_Toc518074300)

[Gambar 3.25 *Entity Realtionship* Diagram APATIPS 98](#_Toc518074303)

[Gambar 3.26 Struktur Menu Admin 102](#_Toc518074304)

[Gambar 3.27 Antar Muka Halaman *Login* 103](#_Toc518074305)

[Gambar 3.28 Antar Muka Halaman Utama 104](#_Toc518074307)

[Gambar 3.29 Antar Muka Halaman Kelola Master Data Barang 105](#_Toc518074308)

[Gambar 3.30 Antar Muka Halaman Tambah Data Barang 106](#_Toc518074309)

[Gambar 3.31 Antar Muka Halaman Ubah Data Barang 107](#_Toc518074310)

[Gambar 3.32 Antar Muka Halaman Hapus Data Barang 107](#_Toc518074311)

[Gambar 3.33 Antar Muka Halaman Kelola Transaksi Penjualan 108](#_Toc518074312)

[Gambar 3.34 Antar Muka Halaman Tambah Transaksi Penjualan 109](#_Toc518074313)

[Gambar 3.35 Antar Muka Halaman Tombol Pilihan Data Barang 110](#_Toc518074316)

[Gambar 3.36 Antar Muka Halaman Ubah Transaksi Penjualan 110](#_Toc518074314)

[Gambar 3.37 Antar Muka Halaman Hapus Transaksi Penjualan 111](#_Toc518074314)

[Gambar 3.38 Antar Muka Halaman Kelola Transaksi Pembelian 112](#_Toc518074312)

[Gambar 3.39 Antar Muka Halaman Tambah Transaksi Pembelian 113](#_Toc518074313)

[Gambar 3.40 Antar Muka Halaman Ubah Transaksi Pembelian 114](#_Toc518074314)

[Gambar 3.41 Antar Muka Halaman Hapus Transaksi Pembelian 115](#_Toc518074314)

[Gambar 3.42 Antar Muka Halaman Kelola Stok Barang Keseluruhan 116](#_Toc518074316)

[Gambar 3.43 Antar Muka Halaman Kelola Stok Barang Habis dan Minim 117](#_Toc518074317)

[Gambar 3.44 Antar Muka Halaman Kelola Laporan 118](#_Toc518074318)

[Gambar 3.45 Antar Muka Halaman *Logout* 118](#_Toc518074319)

[Gambar 3.46 Hasil Cetak Stok Barang Keseluruhan 119](#_Toc518074325)

[Gambar 3.47 Hasil Cetak Stok Barang Habis dan Minim 120](#_Toc518074326)

Gambar [3.48 Hasil Cetak Laporan Detail Penjualan 121](#_Toc518074327)

Gambar [3.49 Hasil Cetak Laporan Penjualan 122](#_Toc518074327)

Gambar [3.50 Hasil Cetak Laporan Detail Pembelian 123](#_Toc518074327)

Gambar [3.51 Hasil Cetak Laporan Pembelian 124](#_Toc518074327)

Gambar [3.52 Hasil Cetak Nota Penjualan 125](#_Toc518074327)

Gambar [3.53 Pesan *Login* Berhasil 126](#_Toc518074328)

Gambar [3.54 Pesan *Login* Gagal 126](#_Toc518074329)

Gambar [3.55 Pesan Data Berhasil Di Tambahkan 127](#_Toc518074330)

Gambar [3.56 Pesan Data Berhasil Di Perbarui 127](#_Toc518074331)

[Gambar 3.57 Pesan Data Berhasil Di Hapus 128](#_Toc518074332)

[Gambar 3.58 Pesan Data *Field* Masih Kosong 128](#_Toc518074333)

[Gambar 3.59 Pesan DataKonfirmasi Simpan 129](#_Toc518074333)

[Gambar 3.60 Pesan DataKonfirmasi Cetak Nota Penjuaaln 129](#_Toc518074333)

[Gambar 3.61 Pesan Data Yang Anda Cari Di Temukan 130](#_Toc518074334)

[Gambar 3.62 Pesan Data Tidak Di Temukan 130](#_Toc518074335)

[Gambar 4.1 Implementasi Halaman *Login* 132](#_Toc518074336)

[Gambar 4.2 Implementasi Halaman Utama Admin 133](#_Toc518074338)

[Gambar 4.3 Implementasi Halaman Kelola Master Data Barang 134](#_Toc518074339)

[Gambar 4.4 Implementasi Halaman Tambah Master Data 134](#_Toc518074340)

[Gambar 4.5 Implementasi Halaman Ubah Master data 135](#_Toc518074341)

[Gambar 4.6 Implementasi Halaman Hapus Master Data 135](#_Toc518074342)

[Gambar 4.7 Implementasi Halaman Kelola Transaksi Penjualan 136](#_Toc518074343)

[Gambar 4.8 Implementasi Halaman Tambah Transaksi Penjualan 136](#_Toc518074343)

[Gambar 4.9 Implementasi Halaman Pilihan Barang untuk Transaksi 137](#_Toc518074343)

[Gambar 4.10 Implementasi Halaman Ubah Transaksi Penjualan 137](#_Toc518074344)

[Gambar 4.11 Implementasi Halaman Hapus Transaksi Penjualan 138](#_Toc518074346)

[Gambar 4.12 Implementasi Halaman Kelola Transaksi Pembelian 138](#_Toc518074343)

[Gambar 4.13 Implementasi Halaman Tambah Transaksi Pembelian 139](#_Toc518074343)

[Gambar 4.14 Implementasi Halaman Ubah Transaksi Pembelian 139](#_Toc518074344)

[Gambar 4.15 Implementasi Halaman Hapus Transaksi Pembelian 140](#_Toc518074346)

[Gambar 4.16 Implementasi Halaman Stok Barang Keseluruhan 140](#_Toc518074344)

[Gambar 4.17 Implementasi Halaman Stok Barang Habis dan Minim 141](#_Toc518074346)

[Gambar 4.18 Implementasi Halaman Laporan 141](#_Toc518074351)

[Gambar 4.19 Implementasi Halaman *Logout* 142](#_Toc518074355)

[Gambar 4.20 Implementasi Hasil Cetak Laporan Stok Keseluruhan 142](#_Toc518074356)

[Gambar 4.21 Implementasi Hasil Cetak Laporan Stok Habis dan Minim 143](#_Toc518074357)

Gambar [4.22 Implementasi Hasil Cetak Laporan Penjualan 143](#_Toc518074358)

Gambar [4.23 Implementasi Hasil Cetak Laporan Detail Penjualan 144](#_Toc518074358)

Gambar [4.24 Implementasi Hasil Cetak Laporan Pembelian 144](#_Toc518074358)

Gambar [4.25 Implementasi Hasil Cetak Laporan Detail Pembelian 145](#_Toc518074358)

Gambar [4.26 Implementasi Hasil Cetak Nota Penjualan 145](#_Toc518074358)

Gambar [4.27 Implementasi Pesan *Login* Berhasil 146](#_Toc518074359)

Gambar [4.28 Implementasi Pesan *Login* Gagal 146](#_Toc518074360)

Gambar [4.29 Implementasi Pesan Data Berhasil Di Tambahkan 146](#_Toc518074361)

Gambar [4.30 Implementasi Pesan Data Berhasil Di Perbarui 147](#_Toc518074362)

[Gambar 4.31 Implementasi Pesan Data Berhasil Di Hapus 147](#_Toc518074363)

Gambar [4.32 Implementasi Pesan Data Konfrimasi Simpan 148](#_Toc518074362)

[Gambar 4.33 Implementasi Pesan Data Konfirmasi Cetak Nota 148](#_Toc518074363)

[Gambar 4.34 Implementasi Pesan Data Field Masih Kosong 149](#_Toc518074364)

[Gambar 4.35 Implementasi Pesan Data Yang Anda Cari Di Temukan 149](#_Toc518074365)

[Gambar 4.36 Implementasi Pesan Data Tidak Di Temukan 149](#_Toc518074366)

# **DAFTAR SIMBOL USE CASE DIAGRAM**

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Nama Simbol** | **Gambar** | **Keterangan** |
| 1 | Aktor |  | Menggambarkan pengguna *software* aplikasi (*user*). |
| 2 | *Use case* |  | Menggambarkan fungsionalitas dari suatu sistem, sehingga customer atau pengguna sistem paham dan mengerti mengenai kegunaan sistem yang akan dibangun. |
| 3 | *Association* |  | Menggambarkan hubungan antara *actor* dengan *Use case*. |
| 4 | *Include* |  | Pemanggilan *Use case* oleh *Use case* lain atau untuk menggambarkan suatu *Use case* termasuk didalam *Use case* lain (diharuskan) relasi ini dapat digunakan untuk menghindari penulisan deskripsi yang berulang ulang. |
| 5 | *Note* |  | Menggambarkan dokumentasi  dari *Use case* |

# **DAFTAR SIMBOL SWIMLANE DIAGRAM**

| **No** | **Nama Simbol** | **Gambar** | **Keterangan** |
| --- | --- | --- | --- |
| 1 | *Start State* |  | Menggambarkan awal dari suatu aktivitas yang berjalan pada sistem |
| 2 | *End State* |  | Menggambarkan akhir dari suatu aktivitas yang berjalan pada sistem. |
| 3 | *Activity* |  | Menggambarkan aktivitas yang dilakukan pada sistem. |
| 4 | *Swimlane* |  | Menggambarkan pembagian /pengelompokan berdasarkan tugas dan fungsi tersendiri. |
| 5 | *Transition to*  *self* |  | Menggambarkan hubungan antara state atau *activity* yang kembali kepada state atau *activity* itu sendiri. |
| 6 | *Transition State* |  | Menggambarkan hubungan antara dua *state*, dua *activity* ataupun antara state dan *activity*. |
| 7 | *Decision* |  | Menggambarkan kondisi dari sebuah aktivitas yang bernilai benar/salah. |
| 8 | *State* |  | Menggambarkan kondisi, situasi ataupun tempat untuk beberapa aktivitas. |
| 9 | *Fork* |  | Menggambarkan aktivitas yang dimulai dengan sebuah aktivitas dan diikuti oleh dua atau lebih aktivitas yang harus dikerjakan. |
| 10 | *Join* |  | Menggambarkan aktivitas yang dimulai dengan dua atau lebih aktivitas yang sudah dilakukan dan menghasilkan sebuah aktivitas. |

# **DAFTAR SIMBOL CLASS DIAGRAM**

| **No.** | **Nama Simbol** | **Gambar** | **Keterangan** |
| --- | --- | --- | --- |
| 1 | *Class* Diagram |  | * ***Class*** : Menggambarkan keadaan (atribut/properti) dari suatu objek. *Class* memiliki tiga area pokok, yaitu: nama, atribut,method. * **Nama** menggambarkan nama dari *class*/objek. * **Atribut** menggambarkan batasan dari nilai yang dapat dimiliki oleh properti tersebut. * **Method** menggambarkan implementasi dari layanan yang dapat diminta dari beberapa object dari *class* , yang mempengaruhi *behaviour*. |
| 2 | *Association* |  | Menggambarkan mekanisme komunikasi suatu objek dengan objek lainnya. Atau dapat juga menggambarkan ketergantungan antarkelas. |
| 3 | *Aggregate* |  | Menggambarkan bahwa suatu objek secara fisik dibentuk dari objek-objek lain, atau secara logis mengandung objek lain. |
| 4 | *Multiplicity* |  | Menggambarkan banyaknya object yang terhubung satu dengan yang lainnya. Contoh :  1 Tepat satu  0..\* Nol atau lebih  1..\* Satu atau lebih  0..1 Nol atau satu  5..8 range 5 s.d. 8  4..6,9 range 4 s.d. 6 dan 9 |

# **DAFTAR SIMBOL SEQUENCE DIAGRAM**

| **No.** | **Nama Simbol** | **Gambar** | **Keterangan** |
| --- | --- | --- | --- |
| 1 | Aktor |  | Menggambarkan seseorang atau sesuatu (seperti perangkat, sistem lain) yang berinteraksi dengan sistem. |
| 2 | *Boundary* |  | Menggambarkan interaksi antara satu atau lebih *actor* dengan sistem, memodelkan bagian dari sistem yang bergantung pada pihak lain disekitarnya dan merupakan pembatas sistem dengan dunia luar. |
| 3 | *Control* |  | Menggambarkan “perilaku  mengatur”, mengkoordinasikan perilaku sistem dan dinamika dari suatu sistem, menangani tugas utama dan mengontrol alur kerja suatu sistem. |
| 4 | *Entity* |  | Menggambarkan informasi yang harus disimpan oleh sistem  (struktur data dari sebuah sistem). |
| 5 | *Object Message* |  | Menggambarkan pesan/hubungan antar objek, yang menunjukan urutan kejadian yang terjadi. |

# **BAB I PENDAHULUAN**

## **Latar Belakang Masalah**

Nando Sport Centre merupakan sebuah toko yang bergerak di bidang penjualan perlengkapan olahraga yang berada di Jl. Ps. Baru, Muka Kecamatan Cianjur. Toko tersebut menjual berbagai macam perlengkapan olah raga mulai sepatu, baju, celana, jaket, kaos kaki dan perlengkapan olahraga lainnya. Srategi produksi yang digunakannya dalam memenuhi permintaan konsumen tidak dilakukan berdasarkan pesanan melainkan persediaan yang disesuaikan dengan permintaan pasar pada periode selanjutnya (*make to stock*) sehingga resiko yang terdapat pada strategi ini adalah tingginya investasi yang harus ditanamkan pada *inventory*.

Saat menjalankan kegiatan sehari-harinya toko masih menggunakan cara-cara manual, seperti pencatatan transaksi penjualan, pencatatan transaksi pembelian, pencarian barang, perhitungan stok, dan pembuatan laporan-laporan yang mana memerlukan waktu yang cukup lama dan resiko kesalahan manusia (*human error*) yang cukup besar. Pada pencatatan transaksi penjualan masih dilakukan manual dengan mencatat didalam sebuah buku yang dapat menyebabkan proses transaksi tersebut terhambat dan kurang efektif, salah satunya apabila buku tersebut hilang atau rusak akan mengakibatkan masalah seperti barang yang seharusnya belum habis disangka habis sehingga melakukan pemesanan kembali, maka yang terjadi adalah penumpukan barang. Selain itu sering juga tercecer dan hilangnya nota-nota pembelian dan lain-lain sehingga saat pencatatan transaksi pembelian mengalami keterhambatan.

Adapun untuk mengatasi permasalahan yang ada di Toko Nando Sport Centre tersebut diperlukan implementasi yang menggunakan sistem komputerisasi khususnya *database* untuk menyimpan dan mengolah data barang dengan tujuan untuk mempermudah dalam pencatatan transaksi pembelian, penjualan, pencarian barang dan penyampaian informasi terhadap stok barang (*inventory*). Apabila di kaitkan dengan perkembangan teknologi saat ini, yang mempengaruhi kemajuan di berbagai bidang salah satunya perangkat lunak, dalam suatu Instansi atau perusahaan Teknologi Informasi memberikan banyak kemudahan dalam pengelolaan sebuah data. Aplikasi desktop adalah aplikasi database yang mampu beroperasi secara *offline* yang ditempatkan dalam sebuah peragkat komputer ataupun *laptop* dengan cara menginstalnya terlebih dahulu. Sehingga dapat dengan mudah untuk penginputan data, perekapan, dan pengeditan maupun penghapusan data.

Berdasarkan permasalahan yang terjadi Toko Nando Sport Centre maka dibutuhkan suatu aplikasi yang dapat memudahkan pengolahan data transaksi penjualan serta stok barang dengan efektif dan efisien. Oleh karena itu, melalui Kerja Praktek ini akan dibuat sebuah perangkat lunak berbasis desktop dengan MYSQLsebagai DBMS, bahasa pemrograman Delphidan aplikasi DELPHI XE8 sebagai media untuk mengimplementasikannya, dan juga nantinya akan dirancang dengan metode tertentu berdasarkan analisis permasalahan yang ada dan sesuai dengan kebutuhan.

Berdasarkan latar belakang tersebut, maka judul kerja praktek yang akan diambil adalah **“APLIKASI PENGELOLAAN DATA TRANSAKSI PENJUALAN DAN STOK BARANG BERBASIS DESKTOP DI TOKO NANDO SPORT CENTRE CIANJUR”.**

## **Rumusan Masalah**

Berdasarkan latar belakang masalah yang telah diuraikan di atas, maka dapat dirumuskan permasalahan yang timbul yaitu :

1. Bagaimana membuat proses mengolah data transaksi jual beli menjadi lebih mudah dan cepat.
2. Bagaimana cara mengelola stok barang di gudang dengan display barang di toko menjadi lebih terorganisir.
3. Bagaimana cara pembuatan laporan transaksi jual beli lebih efektif dan efisien.

Berdasarkan permasalahan diatas maka harus dilakukan sebuah perbaikan sistem yang lebih terkomputerisasi sehingga proses lebih efisien dan dapat memudahkan pekerjaan pegawai toko.

## **Maksud Dan Tujuan**

Maksud dari kerja praktek ini yaitu Pembuatan sebuah aplikasi pengelolaan data transaksi penjualan dan stok barang berbasis desktop di Toko Nando Sport Cianjur dengan menggunakan bahasa pemrograman Delphi dan aplikasi DELPHI XE8sebagai media untuk mengimplementasikannya. Adapun tujuan utama dari pelaksanaan kerja praktek ini yaitu :

1. Mempermudah proses pengolahan data transaksi penjualan dan pembelian sehingga lebih cepat dan praktis.
2. Pembuatan laporan transaksi jual beli lebih mudah, akurat, tepat dan lebih cepat.
3. Pengelolaan stok barang di gudang dengan display barang di toko lebih mudah, terorganisir dan teratur.

## **Batasan Masalah**

Permasalahan yang akan dibahas dalam penelitian ini adalah :

1. Pengelolaan data transaksi penjualan dan pembelian oleh Admin.
2. Pembuatan laporan data barang transaksi penjualan dan pembelian oleh Admin.
3. Pengelolaan stok barang dari manual menjadi terkomputerisasi oleh Admin.

## **Metode Penelitian**

Metode penelitian yang digunakan adalah deskriptif, yaitu suatu metode dengan cara mengumpulkan data, mengolah, serta menganalisa data yang telah terkumpul kemudian ditarik kesimpulan dan diberkan saran-saran yang diperlukan. Metode yang digunakan adalah :

1. Pengumpulan data

Dalam pembuatan Aplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang Berbasis Desktop Di Toko Nando Sport Centre Cianjur, menggunakan beberapa metode penelitian sebagai berikut :

1. Wawancara

Merupakan percakapan dua orang atau lebih, diantara penulis dan pemilik Toko Nando Sport Centre Cianjur yang mana wawancara ini dilakukan untuk mendapatkan informasi yang berkaitan dengan judul kerja praktek.

1. Observasi

Merupakan penelitian secara langsung ke tempat Toko Nando Nando Sport Centre Cianjur dengan menganalisis data-data yang berkaitan dengan judul kerja praktek serta mencari informasi mengenai sistem yang akan dibuat.

1. Tinjauan Pustaka

Merupakan pencarian data dari berbagai sumber serta membaca dan mempelajari buku-buku yang berkaitan dengan permasalahan-permasalahan yang akan dibahas untuk merancang suatu perangkat lunak.

1. Rekayasa Perangkat Lunak

Metode rekayasa perangkat lunak yang digunakan adalah metode *waterfall*. Menurut Pressman metode *waterfall* adalah model klasik yang bersifat sistematis, berurutan dalam membangun *software*.

Berikut adalah gambaran bagaimana pembuatan sebuah *software* dari sebuah *waterfall* model (Pressman, 2010).

|  |
| --- |
|  |
| Gambar 1. 1 Model Waterfall |
| (Sumber : Pressman, 2010) |

Model air terjun atau yang sering disebut model *waterfall* sering dinamakan siklus hidup klasik (*classic life cycle*), dimana hal ini menggambarkan pendekatan yang sistematis dan juga berurutan pada pengembangan perangkat lunak, dimulai dengan spesifikasi kebutuhan pengguna lalu berlanjut melalui tahapan-tahapan seperti perencanaan (*planning*), permodelan (*modeling*), konstruksi (*construction*), serta penyerahan sistem ke para pelanggan/pengguna (*deployment*), yang diakhiri dengan dukungan pada perangkat lunak lengkap yang dihasilkan (*Pressman*, 2010).

Berikut tahapan-tahapan dalam metode *waterfall* menurut Pressman :

1. *Communication*

*Comunication* / komunikasi merupakan sebuah tahapan awal dari metode *waterfall* sebagai penghubung antara penulis dengan Pemilik Toko. Pada tahap ini penulis melakukan komunikasi dengan Pemilik Toko untuk mencari masalah - masalah yang dihadapi dan juga untuk mendapatkan informasi seputar Toko Nando Sport Centre. Masalah dan informasi yang didapat dijadikan sebagai acuan dalam pembuatan aplikasi*.*

1. *Planning*

Proses *Planning* ini merupakan proses lanjutan dari proses *Communication* ( *Analysis Requirement* ). Pada tahap ini penulis mulai menentukan target waktu untuk menyelesaikan seluruh pekerjaan nya, dengan cara memperkirakan beban pekerjaan, menjadwalkan pengerjaanya juga mengamati pekerjaan yang telah dilakukan.

1. *Modeling*

Tahapan *modeling* ini menggunakan metode *Unified Modelling Language* (UML) yaitu pendeskripsian dan desain sistem perangkat lunak, khususnya sistem yang dibangun menggunakan pemograman berorientasi objek dimana dalam tahapan model ini yaitu menggunakan *Usecase* Diagram*, Activity* Diagram*, Class* Diagram*, State* Diagramdan *Sequence* Diagram*.*

1. *Construction*

Pada tahapan ini penulis mulai mengimplementasikan model yang telah dibuat kedalam sebuah aplikasi nyata dengan menggunakan bahasa pemrograman sampai menghasilkan sebuah aplikasi yang dapat digunakan.

1. *Deployment*

Pada tahap *deployment* penulis melakukan uji coba produk yang telah dibuat diimplementasikan pada instansi tempat melakukan kerja praktek.

Berikut ini adalah tabel 1.1 yang berisi perbandingan antara Model *Waterfall* dengan Model *Prototyping*.

##### Tabel 1.1 Perbandingan Model *Waterfall* dengan Model *Prototyping*

|  |  |
| --- | --- |
| Model *Waterfall* | Model *Prototyping* |
| Model pengembangan *software*, dimana kemajuan proses tersebut dipandang menurun langkah demi langkah sperti jatuhnya air terjun yang mengalir kebawah. | Salah satu *paradigm* yang mengembangkan sistem informasi. Bukan hanya sekedar evolusi saja tetapi juga merupakan revolusi yang mengembangkan sistem informasi. |
| Resiko ketidakpastian tinggi | Lebih mementingkan keberhasilan |
| Simple dan mudah diinplementasikan | Implementasinya sangat mudah, maksudnya disini adalah *user* dapat mengenal perangkat lunak yang akan dikembangkan dan tidak merasa asing sejak *user* memilikinya. |
| Cocok untuk proyek kecil yang kita buat | Pengembangan lebih cepat dalam hal ini program bisa langsung dibuat untuk melihat perkembangan tahap demi tahap. |

Alasan menggunakan model *Waterfall* karena metode ini memungkinkan untuk departementalisasi dan kontrol. proses pengembangan model fase *one by one*, sehingga meminimalis kesalahan yang mungkin akan terjadi. Pengembangan bergerak dari konsep, yaitu melalui desain, implementasi, pengujian, instalasi, penyelesaian masalah, dan berakhir di operasi dan pemeliharaan.

## **Jadwal Penelitian**

Tempat Kerja Praktek akan dilaksanakan di :

Nama Instansi : Nando Sport Centre Cianjur

Alamat : Jl. Ps. Baru, Muka Kecamatan Cianjur

Waktu : Februari 2019 – Mei 2019

Waktu yang diperlukan dalam pembuatan laporan kerja praktek ini adalah seperti yang tertera pada tabel 1.2 :

##### Tabel 1.2 Jadwal Penelitian

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Uraian Langkah Pekerjaan** | **Bulan** | | | |
| **Februari** | **Maret** | **April** | **Mei** |
| 1. | Rumusan Masalah |  |  |  |  |
| 2. | Pengumpulan Data |  |  |  |  |
| 3. | Analisis |  |  |  |  |
| 4. | Desain |  |  |  |  |
| 5. | Implementasi |  |  |  |  |
| 6. | Pengujian |  |  |  |  |
| 7. | Penyusunan Laporan |  |  |  |  |
| 8. | Seminar |  |  |  |  |

## **Sistematika Penulisan**

Sistematika penulisan laporan kerja praktek ini dibagi menjadi 5 (lima) bab. Berikut penjelasan tentang masing-masing bab :

**BAB I PENDAHULUAN**

Bab ini menjelaskan tentang latar belakang, rumusan masalah, tujuan dan manfaat, batasan masalah, tempat dan waktu, metode pelaksanaan dan sistematika penulisan.

**BAB II TINJAUAN PUSTAKA**

Bab ini menjelaskan mengenai gambaran organisasi tempat *survey* yang di dalamnya mencakup nama organisasi tempat *survey*, alamat organisasi, sejarah singkat organisasi, struktur organisasi, dan *job* deskripsi serta landasan teori yang digunakan dalam pengembangan sistem yang dibangun. Serta menjelaskan uraian tentang pengertian sistem, komponen sistem, dan model pengembangan sistem serta tentang teori-teori yang mendukung pembuatan sistem.

**BAB III ANALISA DAN PERANCANGAN**

Bab ini menjelaskan tentang analisa sistem yang akan segera dibuat dan bagaimana merancangnya sehingga menjadi sebuah sistem. Selain itu mencari informasi – informasi yang sangat berkaitan dengan sistem yang akan dibangun.

**BAB IV IMPLEMENTASI DAN PENGUJIAN**

Bab ini menjelaskan pembuatan sistem yang merupakan pengimplementasian dari hasil analisa dan perancangan yang berisi gambaran struktur menu serta tampilan layar dan langkah pengujian dari sistem yang dibangun.

**BAB V PENUTUP**

Bab ini berisi kesimpulan tentang seluruh pembahasan dan pemecahan masalah yang telah dilakukan dan saran-saran untuk mengembangkan sistem informasi berikutnya.

# 

# **BAB II TINJAUAN PUSTAKA**

## **2.1 Tinjauan Pustaka**

Tinjauan pustaka membahas tentang gambaran organisasi tempat survei yang didala mnya mencakup sejarah singkat organisasi, struktur organisasi dan deskripsi kerja.

### **2.1.1 Sejarah Singkat Nando Sport Centre Cianjur**

Nando Sport Centre Cianjur merupakan sebuah toko yang bergerak di bidang penjualan perlengkapan olahraga Toko tersebut menjual berbagai macam perlengkapan olahraga mulai dari sepatu, baju, celana, jaket, kaos kaki dan atribut lainnya. Nando Sport Centre beralamat di Jl. Ps. Baru Muka Kecamatan Cianjur. Berdiri sekitar tahun 2011 sampai dengan sekarang, toko Nando Sport Centre ini selalu memberikan kualitas barang dan pelayanan yang baik untuk para pelanggannya sehingga para pelanggan puas dengan barang yang dibeli dan pelayanan yang diberikannnya.

Seiring berjalannnya waktu Toko Nando Sport Centre awalnya hanya membuka usaha di ruko tidak terlalu besar sekarang dari hari ke hari usaha toko tersebut berkembang sehingga dapat berpindah ke tempat ruko yang lebih besar dan nama Toko olahraga Nando Sport Centre menjadi sudah tidak asing lagi bagi masyarakat Cianjur khususnya bagi para pecinta olahraga yang ingin membeli perlengkapannya.

### **2.1.2 Struktur Organisasi**

Struktur Organisasi adalah suatu bagian yang menunjukan tugas-tugas untuk setiap bagian dalam suatu perusahaan untuk mencapai tujuan perusahaan tersebut. Berikut struktur Organisasi Toko Nando Sport Centre Cianjur pada gambar 2.1

|  |
| --- |
| Pegawai  Bag. Gudang  Kepala Pemilik |
| Gambar 2.1 Struktur Organigram Toko Nando Sport Centre |
| (Sumber: Wawancara dengan pemilik toko, 2019) |

### **2.1.3 Deskripsi Kerja**

Deskripsi kerja merupakan ringkasan dari tugas, kewajiban, dan tanggung jawab dari pekerjaan. Hal ini biasanya mengatur hingga hal-hal detail dari sebuah pekerjaan.

Deskripsi kerja Nando Sport Centre Cianjur adalah sebagai berikut :

### **2.1.3.1 Kepala Pemilik**

Deskripsi kerja dari Kepala Pemilik adalah sebagai berikut :

1. Bertanggung jawab atas semua kegiatan atau proses bisnis yang ada di toko
2. Mengeluarkan kebijakan terhadap peraturan toko
3. Membeli persediaan barang
4. Mengelola keuangan pemasukan perusahaan
5. Mengatur keuangan pengeluaran perusahaan
6. Membuat semua laporan yang ada di toko
7. Merencanakan dan merumuskan kebijakan strategis yang menyangkut pemasaran

### **2.1.3.1 Bagian Gudang**

Deskripsi kerja dari Kepala Pemilik adalah sebagai berikut :

1. Mengecek barang yang datang
2. Mengecek *packing list* barang datang dengan dokumen dan data barang yang dibeli
3. Mengecek kembali kondisi barang yang datang
4. Mengelompokkan barang sesuai dengan jenis barang
5. Bertanggung jawab terhadap semua barang

### **2.1.3.1 Pegawai**

Deskripsi kerja dari Kepala Pemilik adalah sebagai berikut :

1. Memberikan pelayanan kepada pelanggan yang ingin membeli barang
2. Memberikan informasi yang benar kepada pelanggan tentang harga barang
3. Menjaga kebersihan dan kerapihan toko
4. Mengecek harga barang
5. Melaporkan barang yang kosong kepada bagian gudang

## **2.2 Landasan Teori**

Landasan teori akan menjelaskan teori – teori mengenai laporan kerja praktek. Teori – teori ini diambil dari buku literatur. Teori yang dibahas meliputi konsep dasar praktek kerja lapangan, serta konsep dasar aplikasi desktop, alat bantu analisis dan perancangan dan perangkat lunak yang digunakan dalam pembangunan sistem informasi.

### **Data**

Data dapat didefinisikan sebagai deskripsi dari suatu kejadian yang kita hadapi. Data dapat berupa catatan-catatan dalam kertas, buku, atau tersimpan sebagai *file* dalam *database.* Data akan menjadi bahan dalam suatu proses pengelolaan data. Oleh karena itu, suatu data belum bisa dapat berbicara banyak sebelum diolah lebih lanjut (Ladzamudin, 2005:8).

### **Pengelolaan Data**

Jogiyanto (2005:2) mengungkapkan : “Pengelolaan data adalah manipulasi dari data ke dalam bentuk yang lebih berguna dan lebih berarti, berupa suatu informasi”.

Proses pengelolaan data terbagi dalam tiga tahapan, yang disebut dengan siklus pengelolaan data (*Data processing cycle)* yaitu :

1. Pada tahapan *input*

Yaitu dapat dilakukan proses pemasukan data ke dalam komputer lewat media *input* (*input devices*).

1. Pada tahapan *processing*

Yaitu dilakukan pengelolaan data yang sudah dimasukkan, yang dilakukan oleh alat pemroses (*process devices*) yang dapat berupa proses perhitungan, perbandingan, pengendalian, atau pencarian di *storage*.

1. Pada tahapan *output*

Yaitu dilakukan proses menghasilkan *output* dari hasil pengelolaan data ke alat *output* (*output devices*) yaitu berupa infromasi.

### **Transaksi**

Transaksi dapat didefinisikan pertemuan antara dua belah pihak ( penjual dan Pembeli) yang saling menguntungkan dengan adanya data/bukti/dokumen pendukung yang dimasukkan kedalam jurnal setelah melalui pencatatan (Bastian, 2007:27).

### **Penjualan**

Penjualan yang berarti berkumpulnya seorang pembeli dan penjual dengan tujuan melaksanakan tukar menukar barang dan jasa berdasarkan pertimbangan yang berharga misalnya pertimbangan uang (Winardi, 2005:26).

Tujuan penjualan yaitu untuk mengubah minat konsumen pada produk ke dalam komitmen untuk membeli produk tersebut, mencapai volume penjualan tertentu , menunjang pertumbuhan perusahaan. Namun intinya tujuan penjualan yaitu untuk mendapatkan keuntungan bagi perusahaan.

### **Persediaan Barang**

Ristono (2009) mendefinisikan persediaan sebagai barang-barang yang disimpan untuk digunakan atau dijual pada masa atau periode yang akan datang. Persediaan terdiri dari persediaan bahan baku, persediaan bahan setengah jadi dan persediaan barang jadi. Persediaan bahan baku dan bahan setengah jadi disimpan sebelum digunakan atau dimasukkan ke dalam proses produksi, sedangkan persediaan barang jadi atau barang dagangan disimpan sebelum dijual atau dipasarkan. Dengan demikian setiap perusahaan yang melakukan kegiatan usaha umumnya memiliki persediaan.

Stok atau persediaan adalah suatu teknik untuk manajemen material yang berkaitan dengan persediaan. Manajemen material dalam stok atau *inventory*  dilakukan dengan beberapa *input*  yang digunakan adalah permintaan yang terjadi (*demand*) dan biaya-biaya yang terkait dengan penyimpanan, serta biaya apabila terjadi kekurangan persediaan (*shortage*). Secara teknis stok atau *inventory* adalah suatu teknik yang berkaitan dengan penetapan terhadap besarnya persediaan bahan yang harus diadakan untuk menjamin kelancaran dalam kegiatan operasi produksi, serta menetapkan jadwal pengadaan dan jumlah pemesanan barang yang seharusnya dilakukan oleh perusahaan (Ristono, 2009).

### **Pengertian Aplikasi**

Aplikasi merupakan satu unit perangkat lunak yang dibuat untuk melayani kebutuhan akan beberapa aktivitas seperti sistem perniagaan, game playanan masyarakat, periklanan, atau semua proses yang hampir dilakukan manusia (Pramana, 2010).

### **Metode *Waterfall* Menurut Roger S. Pessman**

Model air terjun atau yang sering disebut model *waterfall* sering dinamakan siklus hidup klasik (*classic life cycle*), dimana hal ini menggambarkan pendekatan yang sistematis dan juga berurutan pada pengembangan perangkat lunak, dimulai dengan spesifikasi kebutuhan pengguna lalu berlanjut melalui tahapan-tahapan perencanaan (*planning*), permodelan (*modeling*), konstruksi (*construction*), serta penyerahan sistem ke para pelanggan/pengguna (*deployment*), yang diakhiri dengan dukungan pada perangkat lunak lengkap yang dihasilkan (Pressman, 2010).

Berikut tahapan-tahapan dalam metode *waterfall* menurut Pressman :

1. *Communication*

*Comunication* / komunikasi merupakan sebuah tahapan awal dari metode *waterfall* sebagai penghubung antara penulis dengan Pemilik Toko. Pada tahap ini penulis melakukan komunikasi dengan Pemilik Toko untuk mencari masalah - masalah yang dihadapi dan juga untuk mendapatkan informasi seputar Toko Nando Sport Centre. Masalah dan informasi yang didapat dijadikan sebagai acuan dalam pembuatan aplikasi*.*

1. *Planning*

Proses *Planning* ini merupakan proses lanjutan dari proses *Communication* ( *Analysis Requirement* ). Pada tahap ini penulis mulai menentukan target waktu untuk menyelesaikan seluruh pekerjaan nya, dengan cara memperkirakan beban pekerjaan, menjadwalkan pengerjaanya juga mengamati pekerjaan yang telah dilakukan.

1. *Modeling*

Tahapan *modeling* ini menggunakan metode *Unified Modelling Language* (UML) yaitu pendeskripsian dan desain sistem perangkat lunak, khususnya sistem yang dibangun menggunakan pemograman berorientasi objek dimana dalam tahapan model ini yaitu menggunakan *Usecase* Diagram*, Activity* Diagram*, Class* Diagram*, State* Diagramdan *Sequence* Diagram*.*

1. *Construction*

Pada tahapan ini penulis mulai mengimplementasikan model yang telah dibuat kedalam sebuah aplikasi nyata dengan menggunakan bahasa pemrograman sampai menghasilkan sebuah aplikasi yang dapat digunakan.

1. *Deployment*

Pada tahap *deployment* penulis melakukan uji coba produk yang telah dibuat diimplementasikan pada instansi tempat melakukan kerja praktek.

### **Konsep Dasar Aplikasi *Desktop***

Aplikasi dalam bahasa awam sering disebut sebagai sebuah kumpulan program atau script. Aplikasi Desktop yang dibangun dengan menggunakan Struts framework terdiri dari komponen-komponen individual yang digabungkan menjadi satu aplikasi. Aplikasi tersebut dapat diinstal dan dieksekusi oleh aplikasi container. Komponen-komponen tersebut dapat digabungkan karena mereka terletak dalam sebuah konteks aplikasi yang sama, yang menjadikan mereka bergantung satu dengan yang lainnya, baik secara langsung ataupun tidak langsung. (Sutarto & Sutarto, 2009).

### **Konsep Dasar Sistem**

Secara umum, suatu sistem dapat diartikan sebagai suatu kumpulan elemen-elemen yang saling terintegrasi untuk mewujudkan satu tujuan akhir yang sama. Uraian tersebut seperti yang telah dikemukakan oleh Jogiyanto Hartono (Hartono, 2004) yang mendefinisikan bahwa suatu sistem dapat terdiri dari sistem-sistem, bagian (*subsystems*). Masing-masing subsistem dapat terdiri atas komponen-komponen subsistem yang saling berinteraksi dan saling berhubungan membentuk satu kesatuan sehingga tujuan atau sasaran sistem tersebut dapat tercapai. Hal yang sama juga dipaparkan oleh Robert G. Murdic dalam buku Analisis dan Desain Sistem Informasi karangan Al-Bahra Bin Ladjamudin (Ladjamudin, 2008) bahwa Sistem adalah seperangkat elemen-elemen yang terintegrasi dengan maksud yang sama untuk mencapai suatu tujuan bersama.

### **Konsep Dasar Informasi**

1. Pengertian Data

Data berasal dari bahasa Latin yaitu datum yang berarti fakta, kenyataan, kejadian atau peristiwa. Jadi data atau fakta adalah kenyataan dari sesuatu kejadian atau peristiwa. Data dapat didefinisikan yaitu kumpulan fakta-fakta yang berupa fisik dan bukan fisik, kejadian-kejadian dan prosedur yang belum diolah manusia atau peralatan yang digunakan oleh manusia. Kegunaan dari data adalah sebagai bahan dasar yang objektif di dalam proses penyusunan kebijaksanaan dan keputusan oleh pimpinan organisasi. Data merupakan keterangan yang masih mentah (belum diolah). Agar dapat dipergunakan, maka data tersebut harus diolah terlebih dahulu ke dalam bentuk informasi yang sesuai dengan keperluan yang dibutuhkan. (Kenneth C. Laudon, 2018).

1. Pengertian Informasi

Sumber informasi adalah data. Sedangkan data adalah kenyataan yang menggambarkan kejadian-kejadian dan kesatuan nyata. Kejadian (*event*) adalah sesuatu yang terjadi pada saat tertentu. Informasi diperoleh setelah data-data mentah diproses atau diolah. Al -Bahra Bin Ladjamudin (2008) mengungkapkan : “Informasi adalah data yang telah diolah menjadi bentuk yang lebih berarti dan berguna bagi penerimanya untuk mengambil keputusan masa kini maupun yang akan datang”.

### **Konsep Dasar Sistem Informasi**

Sistem Informasi merupakan suatu sistem dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan (Al -Bahra Bin Ladjamudin, 2008).

Mengacu pada definisi sistem yang telah dipaparkan sebelumnya, maka sistem informasi dapat definisikan sebagai suatu sistem yang dibuat oleh manusia yang terdiri atas komponen-komponen dalam suatu organisasi untuk memenuhi kebutuhan akan informasi. Komponen-komponen tersebut antara lain :

1. *Hardware* (perangkat keras), seperti : *keyboard*, *monitor*, *microprocessor*dan lain sebagainya.
2. *Software* (perangkat lunak), merupakan kumpulan dari perintah/fungsi yangditulis dengan aturan tertentu untuk memerintahkan komponen melaksanakan tugas tertentu.
3. *Brainware* (manusia) adalah mereka yang terlibat dalam kegiatan sistem informasi seperti operator, pemimpim sistem informasi dan sebagainya.
4. Data, merupakan komponen dasar dari informasi yang akan diproses lebihlanjut untuk menghasilkan informasi.
5. Prosedur atau metode-metode, menghubungkan berbagai perintah danaturan yang akan menentukan rancangan dan penggunaan sistem informasi.

Adapun sistem infomasi dapat diklasifikasikan menjadi empat (Turban, 1999) antara lain :

1. Berdasarkan Level Organisasi

Berdasarkan level organisasi, sistem informasi dikelompokan menjadi 3, antara lain :

1. Sistem informasi departemen (*departemental information systems*), sistem informasi yang hanya digunakan dalam sebuah departemen.
2. Sistem informasi perusahaan (*enterprise information systems*), sistem terpadu yang dapat digunakan oleh sejumlah departemen secara bersama-sama.
3. Sistem informasi antar organisasi (*inter organizational systems*), sistem informasi yang menghubungkan dua organisasi atau lebih.
4. Berdasarkan Area Fungsional

Pengklasifikasian sistem informasi ini didasari oleh fungsional yang masih bersifat tradisional dari departemen dalam suatu organisasi. Sistem informasi akuntansi, sistem informasi yang menyediakan informasi yang dipakai oleh fungsi akuntansi. sistem ini mencakup semua transaksi yang berhubungan dengan keuangan dalam perusahaan.

1. Sistem informasi keuangan, sistem informasi yang menyediakan informasi pada fungsi keuangan yang menyangkut keuangan perusahaan.
2. Sistem informasi manufaktur, sistem informasi yang bekerja samadengan sistem informasi lain untuk mendukung manajemen perusahaan dalam menyelesaikan masalah yang berhubungan dengan produk atau jasa yang dihasilkan perusahaan.
3. Sistem informasi pemasaran, sistem informasi yang menyediakan informasi yang dipakai oleh fungsi pemasaran.
4. Sistem informasi SDM, sistem informasi yang menyediakan informasiyang dipakai oleh fungsi personalia.
5. Berdasarkan Dukungan yang Tersedia
6. Sistem Pemrosesan Transaksi, menghimpun dan menyimpan informasi transaksi. Fokus utama sistem ini adalah data-data transaksi.
7. Sistem Informasi Manajemen**,** sistem informasi yang digunakan untuk mendukung operasi, manajemen, dan pengambilan keputusan dalam suatu organisasi.
8. Sistem Otomasi Perkantoran**,** sistem yang memberikan fasilitas tugas-tugas pemrosesan informasi sehari-hari di dalam perkantoran dan organisasi bisnis.
9. Sistem Pendukung Keputusan**,** sistem informasi interaktif yang menyediakan informasi, pemodelan, dan manipulasi data yang digunakan untuk membantu pengambilan keputusan di mana tak seorangpun tahu secara pasti bagaimana keputusan seharusnya dibuat.
10. Sistem Informasi Eksekutif**,** sistem informasi yang menyediakan fasilitas yang fleksibel bagi manajer dan eksekutif dalam mengakses informasi eksternal dan internal yang berguna untuk mengidentifikasi masalah dan mengenali peluang.
11. Sistem Pendukung Cerdas**,** menyediakan pengetahuan pakar pada bidang tertentu untuk membantupemecahan masalah.
12. Berdasarkan Aktivitas Management
13. Sistem berbasis *mainframe*.
14. Sistem *stand alone*.
15. Sistem tersebar.

### **Konsep Dasar *Desktop***

*Desktop Based Application* adalah suatu aplikasi yang dapat berjalan sendiri atau independen tanpa menggunakan browser atau koneksi Internet di suatu komputer otonom dengan operating system atau platform tertentu. Aplikasi Desktop difokuskan kepada aplikasi yang lebih independen. Tentu Tujuannya adalah untuk mempermudahkan para pengguna aplikasi desktop dalam hal memodifikasi pengaturan aplikasi sehingga efektifitas, efesinsi waktu, dana, dan tenaga dapat lebih ditekankan semaksimal mungkin. (Konixbam, 2009)

### **Keunggulan Aplikasi Berbasis *Desktop***

Berikut ini adalah beberapa keunggulan dari aplikasi berbasis desktop (Joy, 2011) antara lain :

1. Keamanan sistem lebih baik daripada aplikasi berbasis web dikarenakan dengan menggunakan basis web, serangan virus atau keamanan password yang dapat ditembus sehingga menimbulkan keraguan untuk mencoba aplikasi berbasis web.
2. Fitur lebih lengkap dibanding aplikasi berbasis web.
3. Biaya pengeluaran dalam pembuatan aplikasi berbasis Desktop lebih murah.
4. Kehandalan dari performa sistem yang lebih baik daripada aplikasi berbasis web karena jika server aplikasi berbasis web sedang tidak bagus maka anda akan mengalami gangguan dalam proses bisnis anda.
5. Akses secara Offline sehingga lebih nyaman menggunakan aplikasi tanpa perlu koneksi dengan internet.

### **Konsep Dasar *Database***

Berikut merupakan definisi basis data atau *database* yang telah dikemukakan oleh beberapa ahli diantaranya :

1. Basis data merupakan kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (redudansi) yang tidak perlu, untuk memenuhi berbagai kebutuhan (Fathansyah, 2002).
2. Basis data merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya (Jogiyanto, 2005:2).

### **Konsep Dasar DBMS (*Database Management System*)**

*Database Management System* adalah sebuah perangkat lunak yang memungkinkan pengguna mendefinisi, membentuk dan mengatur basis data dan yang mengendalikan akses ke basis data. DBMS berinteraksi dengan pengguna aplikasi program dan basis data. Database Management System dapat diartikan juga sebagai sistem perangkat lunak yang memungkinkan pengguna dapat mendefinisikan, membuat, merawat, dan mengatur akses ke Basis Data (Begg, 2010).

Biasanya DBMS memungkinkan pengguna untuk mendefinisikan Database melalui sebuah Data Definition Language (DDL), menspesifikasikan tipe data,struktur dan batasan pada data yang disimpan pada Database.

Kemudian juga memungkinkan insert, update, delete, dan mengambil data dari Database melalui Data Manipulation Language (DML), mempunyai pusat penyimpanan untuk semua data dan deskripsi data memungkinkan DML untuk menyediakan fasilitas umum untuk data tersebut yang umumnya disebut bahasa query

### **2.2.16 Pengujian *Black Box***

Pengujian *black box* merupakan pengujian untuk mengetahui apakah semua fungsi perangkat lunak telah berjalan semestinya sesuai dengan kebutuhan fungsional yang telah didefinisikan (Rouf, 2005).

Metode ini memungkinkan perekayasa perangkat lunak mendapatkan serangkaian kondisi input yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program, pengujian *black box* dapat menemukan kesalahan dalam kategori sebagai berikut :

* + - 1. Fungsi-fungsi yang tidak benar atau hilang.
      2. Kesalahan *interface*.
      3. Kesalahan dalam struktur data atau akses basis data *eksternal*.
      4. Inisialisasi dan kesalahan terminasi.
      5. Validitas fungsional.
      6. Kesensitifan sistem terhadap nilai *input* tertentu dan batasan dari suatu data.

### **2.2.17 *Unified Modelling Language***

*Unified Modelling Language* (UML) adalah suatu alat untuk memvisualisasikan dan mendokumentasikan hasil analisa dan desain yang berisi sintak dalam memodelkan sistem secara *visual* juga merupakan satu kumpulan konvensi pemodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem *software* yang terkait dengan objek (Haviludin, 2011).

Menurut Verdi Yasin (Yasin, 2012) UML adalah sebuah "bahasa" yg telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. Dengan menggunakan UML kita dapat membuat model untuk semua jenis aplikasi piranti lunak, dimana aplikasi tersebut dapat berjalan pada piranti keras, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa pemrograman apapun. seperti bahasa - bahasa lainnya, UML mendefinisikan notasi dan *syntax/* semantik. Notasi UML merupakan sekumpulan bentuk khusus untuk menggambarkan berbagai diagram piranti lunak. Setiap bentuk memiliki makna tertentu, dan UML *syntax* mendefinisikan bagaimana bentuk-bentuk tersebut dapat dikombinasikan. Notasi UML terutama diturunkan dari 3 notasi yang telah ada sebelumnya: *OOD (Object-Oriented Design), OMT (Object Modeling Technique), dan OOSE (Object-Oriented Software Engineering).*

Menurut Yasin (2012) di dalam UML terdapat beberapa diagram yaitu :

1. *Use case* Diagram

*Use case* Diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah *use case* mempresentasikan sebuah interaksi antara aktor dengan sistem.

|  |
| --- |
| D:\1 SEMESTER 6\KP\LAPORAN\BAB 3\PARADIGM\Use Case Diagram2.jpg |
| Gambar 2.2 Contoh Use case Diagram |

1. *Class* Diagram

*Class* Diagram adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek.

|  |
| --- |
| D:\1 SEMESTER 6\KP\LAPORAN\tabel contoh uml\classdiagram.png |
| Gambar 2.3 Contoh Class Diagram |

1. *Statechart* Diagram

*Statechart* Diagram menggambarkan transisi dan perubahan keadaan (dari satu *state* ke *state* lainnya) suatu objek pada sistem sebagai akibat dari *stimuli* yang diterima.

|  |
| --- |
| D:\1 SEMESTER 6\KP\LAPORAN\tabel contoh uml\05-simple-state-machine-diagram.png |
| Gambar 2.4 Contoh Statechart Diagram |

1. *Activity* Diagram

*Activity* Diagra*s* menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing – masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir.

|  |
| --- |
| D:\1 SEMESTER 6\KP\LAPORAN\tabel contoh uml\Activity-Diagram-for-communication-case-and-find-marker.png |
| Gambar 2.5 Contoh Activity Diagram |

1. *Sequence* Diagram

*Sequence* Diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu.

|  |
| --- |
| D:\1 SEMESTER 6\KP\LAPORAN\tabel contoh uml\sequence-example-facebook-authentication.png |
| Gambar 2.6 Contoh Sequence Diagram |

1. *Collaboration* Diagram

Collaboration diagram juga menggambarkan interaksi antar objek seperti *sequence* Diagram, tetapi lebih menekankan pada peran masing – masing objek dan bukan pada waktu penyampaian *message.*

|  |
| --- |
| D:\1 SEMESTER 6\KP\LAPORAN\tabel contoh uml\slide_8.jpg |
| Gambar 2.7 Contoh Collaboration Diagram |

1. *Component* Diagram

*Component* Diagram menggambarkan struktur dan hubungan antar komponen piranti lunak, termasuk ketergantungan (*dependency*) diantaranya.

|  |
| --- |
| D:\1 SEMESTER 6\KP\LAPORAN\tabel contoh uml\image069.gif |
| Gambar 2.8 Contoh Component Diagram |

1. *Deployment* Diagram

*Deployment* Diagram menggambarkan detil bagaimana komponen *di-deploy* dalam infrastruktur sistem, dimana komponen akan terletak (pada mesin, *server* atau piranti keras apa), bagaimana kemampuan jaringan pada lokasi tersebut, spesifikasi *server*, dan hal – hal lain yang bersifat fisikal.

|  |
| --- |
| D:\1 SEMESTER 6\KP\LAPORAN\tabel contoh uml\deployment-example-web-application.png |
| Gambar 2.9 Contoh Deployment Diagram |

|  |
| --- |
|  |
| Gambar 2.10 Contoh Simbol Use Case dalam UML |

## ***Tool-Tool* yang Digunakan**

Agar memudahkan dalam pembuatan sistem, maka diperlukan beberapa alat atau *tool-tool* yang digunakan dalam pembuatan sistem informasi administrasi dan teknik untuk proses pelaksanaan pembuatan sistem baru.

### **2.3.1 MySQL**

MySQL merupakan software sistem manajemen database (*Database Management System* – DBMS) yang sangat popular di kalangan pemrogram *web*, terutama di lingkungan *Linux* dengan menggunakan skrip dan Ped. Fungsi MySQL dapat dikatakan sebagai *interpreter query*, karena setiap kita menggunakan *query* SQL (perintah SQL) kita harus meletakkannya di dalam fungsi ini. Dengan kata lain, SQL tidak dapat dijadikan tanpa adanya fungsi MySQL.

MySQL termasuk jenis *relational database management system* (RDBMS). Sehingga istilah seperti tabel, baris dan kolom tetap digunakan dalam MySQL. Pada MySQL, sebuah *database* mengandung beberapa tabel, tabel terdiri dari sejumlah baris dan kolom. SQL merupakan kependekan *Structured Query language*. SQL digunakan untuk berkomunikasi dengan sebuah *database*. SQL adalah bahasa yang meliputi perintah-perintah untuk menyimpan, menerima, memelihara, dan mengatur akses-akses ke basis data serta digunakan untuk memanipulasi dan menampilkan data dari *database* (Rosari, 2008).

MySQL juga dapat berjalan pada personal komputer (banyak pengembangan dari MySQL terjadi pada system yang tidak mahal yaitu *Linux System*). Tetapi MySQL juga *portable* dan dapat berjalan pada sistem operasi yang komersial seperti misalnya *Windows, Solaris, Irix*.

MySQL menggunakan bahasa SQL. SQL (*Structured Query Language*) adalah bahasa *standard* yang digunakan untuk mengakses *server database*. Beberapa keunggulan MySQL dibandingkan dengan database lain adalah:

1. Kecepatan: MySQL cepat. Para pengembang berpendapat bahwa MySQL adalah *database* yang tercepat yang didapat. Pendapat ini dapat di selidiki dengan mengunjungi <http://www.mysql.com/benchmark.html>.
2. Kemudahan dalam penggunaan: MySQL adalah *simple database system* dengan performa tinggi dan tidak kompleks untuk *setup*, dan *administrator*, dibanding dengan system yang lebih besar.
3. Biaya: MySQL gratis untuk semua pengguna.
4. Mendukung bahasa *Query*: MySQL memahami SQL, juga dapat mengakses MySQL menggunakan aplikasi yang mendukung ODBC.
5. Kemampuan: Banyak client dapat berhubungan dengan *server* pada saat yang bersamaan. *Clients* dapat menggunakan *multiple database* secara bersamaan.

### **2.3.2 XAMPP**

XAMPP adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai *server* yang berdiri sendiri (*localhost*), yang terdiri atas program *Apache*, HTTP *Server*, MySQL *database*, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan *Perl*. Nama XAMPP merupakan singkatan dari X (empat sistem operasi apapun), Apache, MySQL, PHP dan Perl. Program ini tersedia dalam GNU (*General Public License*) dan bebas (gratis) dan merupakan *web* *server* yang mudah digunakan yang dapat melayani tampilan halaman *web* yang dinamis.

|  |
| --- |
|  |
| Gambar 2.11 Logo XAMPP |

### **2.3.3 PhpMyAdmin**

PhpMyAdmin adalah aplikasi *web* yang dibuat oleh *phpMyAdmin.net*. phpMyAdmin digunakan untuk administrasi *database* MySQL”. Program ini digunakan untuk mengakses *database* MySQL. Perintah untuk membuat tabel dapat menggunakan *form* yang sudah tersedia pada PhpMyAdminatau dapat langsung menuliskan *script* pada menu SQL. PhpMyAdmindijalankan dengan cara mengetik [http://localhost/](http://localhost/phpmyadmin)*[phpmyadmin](http://localhost/phpmyadmin)* pada web *browser*. (Sibero, 2011)

|  |
| --- |
|  |
| Gambar 2.12 Logo PhpMyAdmin |

### **2.3.4 Delphi XE8**

Delphi merupakan bahasa pemrograman berbasis Windows yang menyediakan fasilitas pembuatan aplikasi visual seperti Visual Basic. Delphi memberikan kemudahan dalam menggunakan kode program, kompilasi yang cepat, penggunaan file unit ganda untuk pemrograman modular, pengembangan perangkat lunak, pola desain yang menarik serta diperkuat dengan bahasa pemrograman yang terstruktur dalam bahasa pemrograman Object Pascal. Delphi memiliki tampilan khusus yang didukung suatu lingkup kerja komponen Delphi untuk membangun suatu aplikasi dengan menggunakan Visual Component Library (VCL). Sebagian besar pengembang Delphi menuliskan dan mengkompilasi kode program dalam IDE (Integrated Development Environment).

Bahasa pemrograman Delphi adalah pemrograman berorientasi objek yang merupakan pengembangan dari bahasa pemrograman Pascal. Pascal adalah pemrograman berbasis DOS yang dibuat pada tahun 1971 oleh Niklaus Wirth dari Swiss. Kata Pascal diambil dari nama matematikiawan dan ahli filsafat dari Perancis, yaitu Blaise Pascal yang lahir tahun 1623 dan meninggal tahun 1662. Pascal dirilis tahun 1983 oleh Borland International, salah satu perusahaan software  terkemuka di California.

Pada tahun 1993, Borland International mengembangkan bahasa pemrograman pascal yang bersifat visual yang disebut Delphi dan resmi dipasarkan pada tahun 1995. Pemrograman ini dibuat secara modern yang berjalan di Sistem Operasi Windows mulai dari versinya yang pertama yaitu Delphi 1 dan di tahun-tahun berikutnya Delphi terus dikembangkan mengikuti kebutuhan zaman. (Kani, Firmansyah, & Sufandi, 2010)

Delphi menawarkan sejumlah keuntungan dalam pengembangan aplikasi, antara lain:

1. Komunitas pengguna yang besar pada [Usenet](https://id.wikipedia.org/wiki/Usenet) maupun [web](https://id.wikipedia.org/wiki/World_Wide_Web).
2. Dapat mengkompilasi menjadi *single executable* (aplikasi portable), memudahkan distribusi dan meminimalisir masalah yang terkait dengan *versioning*
3. Banyaknya dukungan dari pihak ketiga terhadap VCL (biasanya tersedia berikut source codenya) ataupun tools pendukung lainnya (dokumentasi, *tool debugging*)
4. Optimasi kompiler yang cukup cepat
5. Mendukung multiple platform dari source code yang sama

Sedangkan untuk kerugiannya antara lain :

1. Partial single vendor lock-in (Borland dapat menetapkan standar bahasa, kompatibilitas yang harus mengikutinya)
2. Akses pada platform dan library pihak ketiga membutuhkan file-file header yang diterjemahkan ke dalam bahasa pascal
3. Dokumentasi atas platform dan teknik-teknik yang menyertainya sulit ditemukan dalam bahasa pascal (contoh akses COM dan Win32)

|  |
| --- |
|  |
| Gambar 2.13 Logo Delphi RAD Studio XE8 |

### **2.3.5 Visual Paradigm**

Visual Paradigm sebuah *software* model dengan sistem visualisasi memungkinkan model yang telah dibuat dapat digunakan sebagai representasi proyek-proyek lain dilengkapi dengan beberapa fitur yang ada didalmnya sampai pada menganalisa sebuah proyek yang akan dikerjakan. Diagram dapat disusun sedemikian rupa sehingga dapat dipustakakan menjadi proyek per proyek yang saling berkaitan. Hal ini dapat juga membantu memisahkan terhadap pekerjaan proyek sampai *level* terkecil.

Visual Paradigm dapat membandingkan perubahan antara diagram yang satu dengan yang lain yang saling berkaitan dan dapat digunakan untuk melacak perbedaan antara diagram-diagram yang ada. Dengan dukungan *design* layaknya *Freehand* dapat menambahkan ke diagram untuk penjelasan lebih detail apa yang menjadi tujuan. Fasilitas lain dapat menggunakan mouse pen atau pena untuk sketching bentuk jalan, kita juga dapat menambahkan bentuk solid seperti bintang, dan panah *rectangles*.

Kelebihan Visual Paradigm :

1. Diagram dapat disusun sedemikian rupa sehingga dapat dipustakakan menjadi proyek per proyek yang saling berkaitan
2. Membantu memisahkan terhadap pekerjaan proyek sampai level terkecil.
3. Mendukung generalisasi atur di kelas diagram

|  |
| --- |
|  |
| Gambar 2.14 Visual Paradigm For UML |

### **2.3.6 Balsamiq Mockup**

Balsamiq mockup adalah *software* yang digunakan dalam pembuatan tampilan *User Interface* sebuah aplikasi. *Software* ini sudah menyediakan *tools* yang dapat memudahkan dalam membuat desain *prototyping* aplikasi yang akan kita buat. Mockup artinya model atau replika mesin atau struktur, yang digunakan untuk tujuan instruksional atau eksperimental. *Software* ini berfokus pada konten yang ingin digambar dan fungsionalitas yang dibutuhkan oleh pengguna.

|  |
| --- |
|  |
| Gambar 2.15 Logo Balsamiq Mockup 3 |

### **2.3.7 MySQL *Driver* ODBC**

*Open Database Connectivity* (ODBC) adalah *Aplication Programming Interface* (API). *Database* yang khusus digunakan untuk mengakses *database* relasional. ODBC terdapat dalam setiap komputer yang menggunakan sistem operasi *windows*, karena ODBC bagian dari *Windows Open System Architecture* (WOSA). Dalam ODBC disediakan berbagai *Aplication Programming Interface* (API) yang berguna untuk menyediakan dan memberikan stkitar bagi berbagai kegiatan pemrograman. Keuntungan utama menggunakan ODBC ini adalah fleksibilitas, fleksibel disini artinya pengubahan jenis *database* yang dipergunkan oleh sebuah aplikasi yang tidak akan mempengaruhi kode program aplikasi tersebut

# **BAB III ANALISIS DAN PERANCANGAN**

## **Analisa Kebutuhan**

### **Analisis Sistem**

Analisis sistem ini akan membahas mengenai analisis masalah, analisis sistem yang sedang berjalan, analisis kebutuhan fungsional dan analisis kebutuhan nonfungsional.

### **Analisis Masalah**

Nando Sport Centre merupakan sebuah toko yang bergerak di bidang penjualan perlengkapan olahraga yang berada di Jl. Ps. Baru, Muka Kecamatan Cianjur. Toko tersebut menjual berbagai macam perlengkapan olah raga mulai sepatu, baju, celana, jaket, kaos kaki dan perlengkapan olahraga lainnya. Srategi produksi yang digunakannya dalam memenuhi permintaan konsumen tidak dilakukan berdasarkan pesanan melainkan persediaan yang disesuaikan dengan permintaan pasar pada periode selanjutnya (*make to stock*) sehingga resiko yang terdapat pada strategi ini adalah tingginya investasi yang harus ditanamkan pada *inventory*.

Saat menjalankan kegiatan sehari-harinya toko masih menggunakan cara-cara manual, seperti pencatatan transaksi penjualan, pencatatan transaksi pembelian, pencarian barang, perhitungan stok, dan pembuatan laporan-laporan yang mana memerlukan waktu yang cukup lama dan resiko kesalahan manusia (*human error*) yang cukup besar. Pada pencatatan transaksi penjualan masih dilakukan manual dengan mencatat didalam sebuah buku yang dapat menyebabkan proses transaksi tersebut terhambat dan kurang efektif, salah satunya apabila buku tersebut hilang atau rusak akan mengakibatkan masalah seperti barang yang seharusnya belum habis disangka habis sehingga melakukan pemesanan kembali, maka yang terjadi adalah penumpukan barang. Selain itu sering juga tercecer dan hilangnya nota-nota pembelian dan lain-lain sehingga saat pencatatan transaksi pembelian mengalami keterhambatan.

Adapun untuk mengatasi permasalahan yang ada di Toko Nando Sport Centre tersebut diperlukan implementasi yang menggunakan sistem komputerisasi khususnya *database* untuk menyimpan dan mengolah data barang dengan tujuan untuk mempermudah dalam pencatatan transaksi pembelian, penjualan, pencarian barang dan penyampaian informasi terhadap stok barang (*inventory*). Apabila dikaitkan dengan perkembangan teknologi saat ini, yang mempengaruhi kemajuan di berbagai bidang salah satunya perangkat lunak, dalam suatu Instansi atau perusahaan Teknologi Informasi memberikan banyak kemudahan dalam pengelolaan sebuah data. Aplikasi desktop adalah aplikasi database yang mampu beroperasi secara *offline* yang ditempatkan dalam sebuah peragkat komputer ataupun *laptop* dengan cara menginstalnya terlebih dahulu. Sehingga dapat dengan mudah untuk penginputan data, perekapan, dan pengeditan maupun penghapusan data.

Berdasarkan permasalahan yang terjadi Toko Nando Sport Centre maka dibutuhkan suatu aplikasi yang dapat memudahkan pengolahan data transaksi penjualan serta stok barang dengan efektif dan efisien. Oleh karena itu, melalui Kerja Praktek ini akan dibuat sebuah perangkat lunak berbasis desktop. Aplikasi berbasis desktop memiliki kelebihan tersendiri diantaranya :

1. Memiliki performa yang baik karena berjalan di atas spesifikasi hardware yang sudah ditentukan sebelumnya.
2. Tidak memerlukan waktu untuk arus data dari server atau sebaliknya.
3. Mudah melakukan penginputan data dan pembuatan laporan toko.

### **Analisis Sistem yang Sedang Berjalan**

Analisis yang sedang berjalan berisi tentang pemaparan proses bisnis yang ada di Nando Sport Centre Cianjur saat ini. Analisis yang dimaksudkan agar perangkat lunak yang dibangun sesuai dengan kebutuhan toko. Adapun proses bisnis yang ada di Nando Sport Centre Cianjur adalah :

1. Barang masuk ke Gudang
   1. Pemilik melakukan pembelian barang langsung ke Supplier
   2. Supplier memberikan kwitansi barang yang sudah dibeli
   3. Pemilik menerima kwitansi dan membawa barang-barang yang sudah dibeli ke toko dan menyerahkannya pada bagian gudang
   4. Bagian gudang menerima dan mencatat data barang yang masuk
   5. Bagian gudang menyimpan dan mengecek kembali kondisi barang-barang di dalam gudang
2. Barang keluar gudang
   1. Pelanggan datang ke toko melakukan permintaan barang ke penjaga toko
   2. Setelah itu penjaga toko mempersiapkan barang yang diminta pelanggan ke bagian gudang
   3. Bagian gudang akan menyediakan permintaan barang dari penjaga toko yang diminta pelanggan
   4. Selanjutnya bagian gudang akan memberikan barang yang diminta pelanggan pada penjaga toko
   5. Setelah itu bagian gudang mencatat data barang yang keluar dalam buku
   6. Penjaga toko memberikan barang yang diminta pelanggan
   7. Setelah itu pelanggan akan menerima dan memverifikasi barang yang dipesan
   8. Selanjutnya pelangan melakukan transaksi pembayaran kepada penjaga toko
   9. Kemudian penjaga toko menerima transaksi pembayaran
3. Transaksi Penjualan barang
   * + 1. Pelanggan datang langsung ke toko kemudian memilih barang yang ingin dibeli
       2. Jika barang ada maka pelanggan melakukan transaksi pembayaran dengan penjaga toko
       3. Kemudian penjaga toko akan mencatat data hasil transaksi ke dalam buku
       4. Jika barang tidak ada maka pelanggan tidak jadi membeli barang

#### **Activity Diagram Sistem Yang Sedang Berjalan**

1. **Barang masuk ke gudang**

|  |
| --- |
| D:\1 SEMESTER 6\KP\LAPORAN\BAB 3\PARADIGM\PARADIGM.jpg |
| Gambar 3.1 Activity Diagram Barang masuk gudang |

1. **Barang keluar gudang**

|  |
| --- |
| D:\1 SEMESTER 6\KP\LAPORAN\BAB 3\PARADIGM\barang keluar.jpg |
| Gambar 3.2 Activity Diagram Barang keluar gudang |

1. **Transaksi Penjualan Barang**

|  |
| --- |
| D:\1 SEMESTER 6\KP\LAPORAN\BAB 3\PARADIGM\transaksi penjualan.jpg |
| Gambar 3.3 Activity Diagram Transaksi penjualan barang |

### **Analisis Aliran Informasi**

Analisis aliran data informasi pada sistem informasi menggambarkan bentuk data dan informasi mengenai kebutuhan data. Analisis aliran informasi akan membahas mengenai kebutuhan data, informasi yang dihasilkan dan dokumen yang terlibat.

#### **Analisis Dokumen**

Analisis dokumen adalah proses mengidentifikasi permasalahan yang ada dalam tiap-tiap dokumen. Proses identifikasi dilakukan dengan cara tiap-tiap dokumen dan pernyataan, hal ini dimaksudkan untuk memudahkan dalam melakukan identifikasi.

Berikut adalah tabel 3.1 yang mengidentifikasi dokumen yang ada di Nando Sport Centre Cianjur.

Tabel 3.1 Hasil Analisis Dokumen

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Nama Dokumen** | **Fungsi** | **Tujuan** | **Bentuk Informasi** |
| 1. | Struk Pembayaran | Dokumen ini digunakan untuk mencatat hasil dari transaksi antara pelanggan/*customer* yang dicatat oleh pegawai toko | Pelanggan (*Customer*) | Kwitansi |
| 2. | Struk Pembelian barang dari supplier | Dokumen ini digunakan untuk mencatat pembelian barang dari supplier | Pemilik Toko | Kwitansi |

### **Analisis Kebutuhan Non Fungsional**

Analisis kebutuhan nonfungsional menggambarkan kebutuhan yang diperlukan untuk menjalankan sistem yang dibangun. Analisis kebutuhan nonfungsional dilakukan untuk mengetahui spesifikasi kebutuhan untuk sistem.

Spesifikasi kebutuhan melibatkan analisis perangkat keras (*hardware)*, analisis perangkat lunak (*software*), analisis pengguna (*user*).

#### **Analisis Perangkat Keras**

Untuk menjalankan aplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang ini dibutuhkan perangkat keras sehingga aplikasi yang dibangun dapat berjalan dengan baik. Spesifikasi minimun perangkat keras untuk menjalankan aplikasi ini adalah sebagai berikut :

##### Tabel 3.2 Kebutuhan Spesifikasi Perangkat Keras Minimal

|  |  |  |
| --- | --- | --- |
| **No.** | **Perangkat** | **Spesifikasi** |
| 1 | *Processor* | Pentium 4 |
| 2 | *Harddisk* | 8 GB |
| 3 | *Memory* | 512 MB |
| 4 | VGA | 256 MB |
| 5 | *Monitor* | Resolusi 1024 x 768 pixel |
| 6 | *Mouse* dan *Keyboard* | Ya |

#### **Analisis Perangkat Lunak**

Spesifikasi perangkat lunak yang dibutuhkan untuk mendukung aplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang yang akan dibangun adalah sebagai berikut:

1. Sistem Operasi Windows 7

Sistem Operasi digunakan untuk menjadi perantara antara perangkat lunak dengan perangkat keras komputer. Sistem Operasi menjadi pengatur atau bertanggung jawab dalam penggunaan perangkat keras dan bertugas untuk menjalankan perangkat lunak atau aplikasi.

1. *DBMS XAMPP*

Dengan menggunakan *DBMS*, maka pengguna akan diberikan fasilitas untuk melakukan penyimpanan, pengolahan dan transformasi data di dalam basis data.

1. *Mysql Driver ODBC*

Untuk mengkomunikasikan aplikasi berbasis desktop khususnya yang menggunakan *IDE Delphi*, maka digunakan *Mysql Driver ODBC* agar aplikasi berbasis desktop dapat terhubung dengan *MySQL*.

#### **Analisis Pengguna**

Analisis pengguna menunjukkan siapa saja yang terlibat dalam aplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang di Nando Sport Centre Cianjur, serta hak akses apa saja dalam aplikasi tersebut. Aplikasiyang dibangun memiliki satu pengguna yaitu :

1. Admin Sistem

Admin sistem ini mempunyai hak akses penuh terhadap aplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang di Nando Sport Centre Cianjur. Admin sistem ini juga yang mengendalikan sistem dan mengatur jalannya sistem. Admin sistem dalam hal ini juga bisa merupakan pemilik dari Nando Sport Centre Cianjur maupun pegawai yang ditunjuk dan dipercaya oleh pemilik.

### **Analisis Kebutuhan Fungsional**

Kebutuhan fungsional menjelaskan mengenai proses-proses apa saja yang akan dilakukan oleh sistem. Deskripsi dari kebutuhan aktivitas-aktivitas dan layanan-layanan yang harus disediakan oleh sistem, kebutuhan fungsional sistem menggambarkan layanan secara detail. Berikut adalah fungsi atau modul yang dibutuhkan oleh sistem yang nantinya akan digunakan oleh Admin*.*

Tabel 3.3 Tabel Kebutuhan Fungsional

|  |  |  |  |
| --- | --- | --- | --- |
| No | Modul | *User* | Keterangan |
| 1. | 1. *Login* AdminSistem  2. Kelola Master Data Barang  3. Kelola Data Transaksi  4. Kelola Stok Barang  5. Kelola Laporan  6. *Logout* | Admin Sistem | Orang yang mengelola semua data di dalam aplikasi*.* |

Analisa pemodelan sistem yang dilakukan dengan menggunakan UML (*Unified Modeling Language*)sebagai penggambaran, perencanaan dan pembuatan sistem yang akan dibangun. Adapun tahapan analisis sistem menggunakan UML adalah *Use case* Diagram*, Use Case Scenario, Activity* Diagram*, Class* Diagram*, State* Diagram dan *Sequence* Diagram*.*

## **Perancangan Sistem**

### **Daftar *Use case***

*Use case* menggambarkan secara sederhana fungsi-fungsi utama dari sistem dan berbagai user yang akan berinteraksi dengan sistem tersebut*. Use case* merepresentasikan cara yang dilakukan aktor melalui fungsi dari awal dari akhir. Berikut ini adalah tabel yang menyajikan daftar *Use Case* dari sistem atau aplikasi yang akan di bangun.

###### Tabel 3.4 Daftar Use Case

|  |  |  |
| --- | --- | --- |
| **Kode** | **Business *Use case*** | **Keterangan** |
| U1 | *Login* | Admin, dapat melakukan *Login* aplikasi. |
| U2 | Kelola Master Data Barang | Admin sistem dapat mengelola data barang. |
| U3 | Kelola Data Transaksi | Admin sistem dapat mengelola data transaksi penjualan dan pembelian. |
| U4 | Kelola Stok Barang | Admin sistem dapat mengecek, dan mencetak stok barang. |
| U5 | Kelola Laporan | Admin sistem dapat mencetak berbagai laporan |
| U6 | *Logout* | Admin sistem, dapat keluar dari aplikasi. |

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

*Use Case* Diagrammendeskripsikan kelakuan sistem dari sudut pandang pengguna, berguna untuk membantu memahami kebutuhan. *Use case* adalah dasar dari diagram lain. *Use case* adalah abstraksi dari interaksi antara *system* dan *actor*. *Use case* berkerja dengan mendeskripsikan tipe interaksi antara *actor* sebuah *system* dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah *system* dipakai.

Berikut *Use Case* DiagramdariAplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang di Nando Sport Centre Cianjur.

|  |  |  |
| --- | --- | --- |
| |  | | --- | | D:\1 SEMESTER 6\KP\LAPORAN\BAB 3\GAMBAR UML\Usecase.jpg | | Gambar 3.4 Use Case Diagram Aplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang di Nando Sport Centre Cianjur | |

Keterangan :

Terdapat satu aktor dalam *Use Case* Diagram aplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang yaitu Admin yang mempunyai hubungan dengan *use case login* untuk masuk ke sistem, *use case* kelola master data barang untuk mengelola data barang, *use case* kelola data transaksi untuk mengelola data transaksi penjualan dan pembelian, *use case* kelola stok barang untuk mengetahui stok barang terkini dari toko, *use case* kelola laporan untuk membuat dan mencetak laporan¸ dan *use case logout*  untuk keluar dari sistem dan kembali ke form login.

### ***Use Case Scenario***

*Use case scenario* mendeskripsikan urutan langkah-langkah dalam proses yang dilakukan aktor terhadap sistem maupun yang dilakukan sistem terhadap aktor. Berikut *Use Case Scenario* dariAplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang di Nando Sport Centre Cianjur.

### ***Use Case Scenario Login***

##### Tabel 3.5 *Use Case Scenario* (U1) : *Login*

|  |  |
| --- | --- |
| **Identifikasi** | |
| **Nama** | U1. *Login* |
| **Aktor** | Admin |
| ***Use Case* yang Berkaitan** | - |
| **Kondisi Awal** | Halaman *Login* |
| **Skenario Utama** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. Admin membuka aplikasi | 1. Menampilkan halaman *Login* |
| 1. Admin mengisi *username* dan *password* 2. Jika Ingin Melihat Password : Menekan checkbox yang tersedia 3. Menekan tombol *Login* | 1. Validasi *username* dan *password* |
|  | 1. Jika benar, menampilkan halaman utama sistem 2. Jika salah, menampilkan pesan kesalahan dan kembali mengisi *username* dan *password* |
| **Kondisi Akhir** | Menampilkan halaman utama sistem |

##### 

### ***Use Case Scenario* Kelola Master Data Barang**

##### Tabel 3.6 *Use Case Scenario* (U2) : Kelola Master Data Barang

|  |  |
| --- | --- |
| **Identifikasi** | |
| **Nama** | U2. Kelola Master Data Barang |
| **Aktor** | Admin |
| ***Use Case* yang Berkaitan** | U1. *Login* |
| **Kondisi Awal** | Halaman Utama |
| **Skenario Utama** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. Admin telah melakukan *Login* | 1. Sistem menampilkan halaman utama |
| 1. Memilih menu Kelola Master Data Barang | 1. Menampilkan halaman Kelola Master Data Barang |
| 1. Tambah data barang : menekan tombol tambah | 1. Menampilkan form tambah data barang |
| 1. Mengisi form tambah data barang |  |
| 1. Menekan tombol simpan | 1. Apabila pengisian data benar maka sistem akan menyimpan data 2. Menampilkan pesan data berhasil ditambahkan 3. Kembali ke halaman Kelola Master Data Barang 4. Apabila pengisian data salah maka sistem menampilkan pesan kesalahan 5. Kembali ke halaman Kelola Master Data Barang |
| 1. Ubah data barang : memilih data yang akan di ubah 2. Menekan tombol ubah | 1. Menampilkan *Form* ubah data barang |
| 1. Mengubah data barang 2. Menekan tombol simpan | 1. Apabila proses pengubahan data benar, maka sistem memperbarui data 2. Menampilkan pesan data barang berhasil diperbarui 3. Kembali ke halaman Kelola Master Data Barang 4. Apabila proses pengubahan data salah, maka sistem menampilkan pesan kesalahan 5. Kembali ke *Form*  ubah data barang |
| 1. Hapus data barang : memilih data yang akan di hapus 2. Menekan tombol hapus | 1. Menampilkan pesan konfirmasi penghapusan |
| 1. Menekan tombol Ya | 1. Menghapus data 2. Menampilkan pesan data berhasil dihapus 3. Kembali ke halaman Kelola Master Data Barang |
| 1. Menekan tombol Tidak | 1. Kembali memilih data yang akan di hapus |
| 1. Cari data : memilih cari data berdasarkan apa dan mengisi kata kunci pada kolom yang tersedia 2. Menekan tombol cari | 1. Jika data yang di cari ada, maka sistem akan menampilkan pesan data berhasil ditemukan dan menampilkan data hasil pencarian 2. Jika data yang di cari tidak ada, maka sistem akan menampilkan pesan data yang di cari tidak ada 3. Kembali ke kolom cari |
| 1. Refresh Tabel : Menekan tombol refsresh | 1. Memuat ulang seluruh data pada tabel 2. Membersihkan kolom cari data |
| **Kondisi Akhir** | Menampilkan Halaman Kelola Master Data Barang |

### ***Use Case Scenario* Kelola Data Transaksi**

##### Tabel 3.7 *Use Case Scenario* (U3) : Kelola Data Transaksi

|  |  |
| --- | --- |
| **Identifikasi** | |
| **Nama** | U3. Kelola Data Transaksi |
| **Aktor** | Admin |
| ***Use Case* yang Berkaitan** | U1. *Login*, U4. Kelola Stok Barang, U5. Kelola Laporan |
| **Kondisi Awal** | Halaman Utama |
| **Skenario Utama** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. Admin telah melakukan *Login* | 1. Sistem menampilkan halaman utama |
| 1. Memilih menu Kelola Transaksi | 1. Menampilkan sub menu Kelola Transaksi |
| 1. Memilih salah satu sub menu Kelola Transaksi (Kelola Penjualan, atau Kelola Pembelian) | 1. Menampilkan sub menu halaman Kelola Transaksi yang dipilih (Kelola Penjualan, atau Kelola Pembelian) |
| 1. Tambah data transaksi : menekan tombol tambah (Kelola Penjualan atau Kelola Pembelian) | 1. Menampilkan *Form* tambah data transaksi (Kelola Penjualan atau Kelola Pembelian) |
| 1. Mengisi *Form* tambah transaksi (Kelola Penjualan atau Kelola Pembelian) 2. Menekan tombol simpan | 1. Apabila pengisian data benar makan sistem menyimpan data 2. Menampilkan pesan data berhasil di tambahkan 3. Menampilkan pesan konfirmasi untuk cetak nota penjualan |
| 1. Menekan Tombol Ya | 1. Mencetak Nota penjualan 2. Kembali ke halaman Kelola Transaksi yang dipilih |
| 1. Menekan Tombol Tidak | 1. Kembali ke halaman Kelola Transaksi yang dipilih 2. Apabila pengisian data salah maka sistem menampilkan pesan kesalahan 3. Kembali ke *Form* Tambah Data Transaksi |
| 1. Ubah Data Transaksi : Memilih data yang akan di ubah (Kelola Penjualan atau Kelola Pembelian) 2. Menekan tombol Ubah | 1. Menampilkan *Form* Ubah Transaksi (Kelola Penjualan atau Kelola Pembelian) |
| 1. Mengubah data Transaksi (Kelola Penjualan atau Kelola Pembelian) 2. Menekan tombol Simpan | 1. Apabila proses pengubahan data benar, maka sistem memperbarui data 2. Menampilkan pesan Data Berhasil Diperbarui 3. Kembali ke halaman Kelola Transaksi yang dipilih 4. Apabila proses pengubahan data salah, maka sistem menampilkan pesan kesalahan 5. Kembali ke *Form*  Ubah Data Transaksi |
| 1. Hapus data transaksi : Memilih data yang akan di hapus (Kelola Penjualan atau Kelola Pembelian) 2. Menekan tombol Hapus | 1. Menampilkan pesan konfirmasi penghapusan |
| 1. Menekan tombol Ya | 1. Menghapus data 2. Menampilkan pesan Data Berhasil Di Hapus 3. Kembali ke halaman Kelola Transaksi yang dipilih (Kelola Penjualan atau Kelola Pembelian) |
| 1. Menekan tombol Tidak | 1. Kembali memilih data yang akan di hapus |
| 1. Cari data : Memasukkan kata kunci no nota pada kolom yang tersedia (Kelola Penjualan, atau Kelola Pembelian) 2. Menekan tombol cari | 1. Jika data yang di cari ada, maka sistem akan menampilkan data hasil pencarian 2. Jika data yang di cari tidak ada, maka sistem akan menampilakan pesan data yang di cari tidak ada 3. Kembali ke kolom cari |
| 1. Refresh Tabel : Menekan tombol refsresh (Kelola Penjualan, atau Kelola Pembelian) | 1. Memuat ulang seluruh data pada tabel 2. Membersihkan kolom cari data |
| **Kondisi Akhir** | Menampilkan Halaman Kelola Data Transaksi yang dipilih admin (Kelola Penjualan atau Kelola Pembelian) |

### ***Use Case Scenario* Kelola Stok Barang**

##### Tabel 3.8 *Use Case Scenario* (U4) : Kelola Stok Barang

|  |  |
| --- | --- |
| **Identifikasi** | |
| **Nama** | U4. Kelola Stok Barang |
| **Aktor** | Admin |
| ***Use Case* yang Berkaitan** | U1. *Login*, U3. Kelola Transaksi Transaksi, U5. Kelola Laporan |
| **Kondisi Awal** | Halaman Utama |
| **Skenario Utama** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. Admin telah melakukan *Login* | 1. Sistem menampilkan halaman utama |
| 1. Memilih menu Kelola Stok Barang | 1. Menampilkan sub menu Kelola Stok Barang |
| 1. Memilih salah satu sub menu Kelola Stok Barang (Stok Barang Keseluruhan atau Stok Barang Habis dan Minim) | 1. Menampilkan halaman sub menu Kelola Stok Barang yang dipilih |
| 1. Cari data : Memilih Pilihan cari berdasarkan apa dan Memasukkan kata kunci untuk mencari data pada kolom yang tersedia 2. Menekan tombol Cari | 1. Jika data yang di cari ada, maka sistem akan menampilkan data hasil pencarian 2. Jika data yang di sortir tidak ada, maka sistem akan menampilkan pesan kesalahan 3. Kembali ke kolom cari |
| 1. Refresh Tabel : Menekan tombol (Stok Barang Keseluruhan atau Stok Barang Habis dan Minim) | 1. Memuat ulang seluruh data pada tabel 2. Membersihkan kolom cari data |
| 1. Cetak data stok barang : Menekan tombol cetak | 1. Menampilkan Form Stok Barang Siap Cetak |
| 1. Menekan Tombol Cetak | 1. Mencetak Stok Barang |
| **Kondisi Akhir** | Halaman Sub Menu Kelola Stok Barang Yang Dipilih |

### ***Use Case Scenario* Kelola Laporan**

##### Tabel 3.9 *Use Case Scenario* (U6) : Kelola Laporan

|  |  |
| --- | --- |
| **Identifikasi** | |
| **Nama** | U6. Kelola Laporan |
| **Aktor** | Admin |
| ***Use Case* yang Berkaitan** | U1. *Login*, U3. Kelola Data Transaksi |
| **Kondisi Awal** | Halaman Utama |
| **Skenario Utama** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. Admin telah melakukan *Login* | 1. Sistem menampilkan halaman Utama |
| 1. Memilih menu Kelola Laporan | 1. Menampilkan halaman Kelola Laporan |
| 1. Menekan combobox pilihan laporan | 1. Menampilkan pilihan laporan yang dapat dicetak |
| 1. Memilih laporan yang akan dicetak 2. Memilih tanggal untuk sortir laporan 3. Menekan tombol cetak laporan | 1. Menampilkan *Form* laporan yang dipilih siap cetak |
| 1. Menekan tombol cetak | 1. Mencetak laporan |
| **Kondisi Akhir** | Halaman Kelola Laporan |

### ***Use Case Scenario Logout***

##### Tabel 3.10 *Use Case Scenario* (U6) : *Logout*

|  |  |
| --- | --- |
| **Identifikasi** | |
| **Nama** | U6. *Logout* |
| **Aktor** | Admin |
| ***Use Case* yang Berkaitan** | U1. *Login* |
| **Kondisi Awal** | Halaman Utama |
| **Skenario Utama** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. Memilih menu *logout* | 1. Menampilkan pesan konfirmasi |
| 1. Menekan tombol ya | 1. Menampilkan halaman *login* |
| 1. Menekan tombol tidak | 1. Menampilkan halaman utama |
| **Kondisi Akhir** | Halaman *Login* |

### ***Activity* Diagram**

*Activity* Diagram merupakan diagram yang memperlihatkan aliran aktivitas-aktivitas yang dideskripsikan oleh *use case* dan pada saat yang bersamaan memperlihatkan aktor mana atau kelas analisa mana yang bertanggung jawab untuk aksi tertentu yang dideskripsikan oleh kotak aktivitas. Berikut merupakan *Activity* Diagram dari Aplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang di Nando Sport Centre Cianjur:

### ***Activity* Diagram *Login***

Didalam *Activity* Diagram *Login*, aktor yang terlibat atau berinteraksi dengan sistem adalah Admin *Login* dilakukan oleh aktor untuk masuk kedalam aplikasi. Berikut *Activity* Diagram *Login* dariAplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang di Nando Sport Centre Cianjur.

|  |
| --- |
| D:\1 SEMESTER 6\KP\LAPORAN\BAB 3\GAMBAR UML\Swimlane1.jpg |
| Gambar 3.5 Activity Diagram (U1) : Login |

Keterangan :

Untuk melakukan *login* kedalam sistem admin perlu membuka aplikasi terlebih dahulu lalu sistem akan menampilkan halaman *login* selanjutnya admin memasukkan *username* dan *password* dan menekan tombol login. Jika *username* dan *password* sesuai maka sistem akan menampilkan halaman utama, tetapi jika tidak sistem akan menampilkan pesan kesalahan dan admin diharuskan untuk mengisi *username* dan *password* kembali.

### ***Activity* Diagram Kelola Master Data Barang**

Didalam *Activity* Diagram Kelola Master Data Barang ini, aktor yang terlibat atau berinteraksi dengan sistem adalah Admin. Kelola Master Data Barang ini dilakukan untuk mendata barang yang ada di Nando Sport Centre Cianjur. Berikut *Activity* Diagram Kelola Master Data Barang dariAplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang di Nando Sport Centre Cianjur.

|  |
| --- |
| D:\1 SEMESTER 6\KP\LAPORAN\BAB 3\GAMBAR UML\BARU.jpg\U2 Kelola Master Data Barang (Baru).jpg |
| Gambar 3.6 Activity Diagram (U2) : Kelola Master Data Barang |

Keterangan :

Untuk melakukan Kelola Master Data Barang admin sudah login kedalam sistem dan menampilkan halaman utama. Lalu Admin memilih menu Kelola Master Data Barang dan sistem menampilkan halaman Kelola Master Data Barang. Pada halaman ini admin dapat melakukan 4 aksi yaitu tambah data barang, ubah data barang, hapus data barang, dan cari data barang.

### ***Activity* Diagram Kelola Data Transaksi**

Didalam *Activity* Diagram Kelola Data Transaksi ini, aktor yang terlibat atau berinteraksi dengan sistem adalah Admin. Kelola Data Transkasi ini dilakukan untuk mendata transaksi penjualan dan pembelian yang dilakukan di Nando Sport Centre Cianjur. Berikut *Activity* Diagram Kelola Master Data Barang dariAplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang di Nando Sport Centre Cianjur.

|  |
| --- |
|  |
| Gambar 3.7 Activity Diagram (U3) : Kelola Data Transaksi |

Keterangan :

Untuk melakukan Kelola Data Transaksi admin sudah login kedalam sistem dan menampilkan halaman utama. Selanjutnya Admin memilih menu Kelola Data Transaksi lalu sistem menampilkan Sub Menu Kelola Data Transaksi. Kemudian Admin memilih salah satu diantara Sub Menu Kelola Data Transaksi dan sistem menampilkan halaman Kelola Data Transaksi yang dipilih admin. Pada 2 halaman Sub Menu Kelola Data Transaksi yaitu Penjualan dan Pembelian ini admin dapat melakukan 4 aksi yaitu tambah data transaksi, ubah data transaksi, hapus data transaksi, dan cari data transaksi. Pada 1 halaman Sub Menu Kelola Data Transaksi yaitu Transaksi Penjualan admin bisa melakukan cetak nota penjualan.

### ***Activity* Diagram Kelola Stok Barang**

Didalam *Activity* Diagram Kelola Stok Barang ini, aktor yang terlibat atau berinteraksi dengan sistem adalah Admin. Kelola Stok Barang ini dilakukan untuk melihat, mensortir dan mencetak stok barang yang dilakukan di Nando Sport Centre Cianjur. Berikut *Activity* Diagram Kelola Stok Barang dariAplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang di Nando Sport Centre Cianjur.

|  |
| --- |
|  |
| Gambar 3.8 Activity Diagram (U4) : Kelola Stok Barang |

Keterangan :

Untuk melakukan Kelola Stok Barang admin sudah login kedalam sistem dan menampilkan halaman utama. Selanjutnya Admin memilih menu Kelola Stok Barang lalu sistem menampilkan Sub Menu Kelola Stok Barang Keseluruhan dan Sub Menu Kelola Stok Barang Habis dan Minim. Kemudian Admin memilih salah satu diantara Sub Menu Kelola Stok Barang dan sistem menampilkan halaman Kelola Stok Barang yang dipilih admin. Pada 2 halaman Sub Menu Kelola Stok Barang ini admin dapat melakukan 2 aksi yaitu cari stok barang dan mencetak stok barang.

### ***Activity* Diagram Kelola Laporan**

Didalam *Activity* Diagram Kelola Laporan ini, aktor yang terlibat atau berinteraksi dengan sistem adalah Admin. Kelola Laporan ini dilakukan untuk mencetak laporan di Nando Sport Centre Cianjur. Berikut *Activity* Diagram Kelola Laporan dariAplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang di Nando Sport Centre Cianjur.

|  |
| --- |
|  |
| Gambar 3.9 Activity Diagram (U5) : Kelola Laporan |

Keterangan :

Untuk melakukan Kelola Laporan admin sudah login kedalam sistem dan menampilkan halaman utama. Lalu Admin memilih menu Kelola Laporan dan sistem menampilkan halaman Kelola Laporan. Pada halaman ini admin dapat memilih laporan yang ingin dicetak lalu mensortir data laporan sesuai tanggal yang dipilih setelah itu admin menekan tombol cetak lalu sistem akan menampilkan *form* laporan yang akan dicetak tersebut. Selanjutnya admin menekan tombol cetak yang tersedia lalu sistem mencetak laporan tersebut.

### ***Activity* Diagram *Logout***

Didalam *Activity* Diagram *Logout*, aktor yang terlibat atau berinteraksi dengan sistem adalah Admin. *Logout* dilakukan oleh aktor untuk keluar dari aplikasi. Berikut *Activity* Diagram *Logout* dariAplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang di Nando Sport Centre Cianjur.

|  |
| --- |
| D:\1 SEMESTER 6\KP\LAPORAN\BAB 3\GAMBAR UML\Swimlane6.jpg |
| Gambar 3.10 Activity Diagram (U6) : Logout |

Keterangan :

Untuk melakukan *logout* dari sistem admin memilih menu *logout* lalu sistem akan menampilkan pesan konfirmasi, jika admin menekan tombol ya maka berhasil keluar dan menampilkan halaman login kembali tetapi jika menekan tombol tidak maka sistem akan menampilkan kembali halaman utama.

### ***Class* Diagram**

*Class* Diagramadalah pandangan aplikasi yang statis. *Class* Diagramtidak hanya menggambarkan visualisasi, menggambarkan dan mendokumentasikan aspek yang berbeda dalam sistem, tetapi juga untuk kontruksi eksekusi kode dalam *software* aplikasi. *Class* Diagramdigunakan untuk mengelompokkan hal-hal inti dari setiap proses yang ingin dilakukan. Semua proses dimasukkan ke dalam tiap-tiap *Class* dan saling dihubungkan pada *Class*-*Class* lainnya yang saling berhubungan. Berikut merupakan *Class* Diagramdari Aplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang di Nando Sport Centre Cianjur :

|  |
| --- |
|  |
| Gambar 3.11 Class Diagram Aplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang di Nando Sport Centre Cianjur |

Keterangan :

Hubungan antar *class login* dan *class* admin adalah agregasi. Hubungan *class* master data, *class* stok barang, *class* detail transaksi penjulan, class penjualan, *class* detail transaksi pembelian dan *class* pembelian adalah komposisi. Hubungan *class* laporan dengan *class-class* lain adalah asosiasi.

### ***State* Diagram**

*State* Diagram adalah suatu metode untuk merepresentasikan jalannya suatu sistem yang digambarkan berdasarkan kejadian sebab sistem tersebut keadaanya harus diubah. *State* Diagram menunjukan sebuah aksi atau proses aktifasi berdasarkan akibat dari kejadian yang berupa fakta. Diagram *state UML* merepresentasikan keadaan-keadaan (*state*) aktif untuk masing-masing kelas dan *event-event* (pemicu) yang menyebabkan perubahan-perubahan diantara keadaan-keadaan (*state)* aktif tersebut. Berikut merupakan *State* Diagram dari Aplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang di Nando Sport Centre Cianjur :

### ***State* Diagram *Login***

Terdapat dua fungsi ketika sistem mengakses objek *Login*. Fungsi tersebut memicu sistem untuk melakukan *event* sesuai dengan fungsi yang telah dijalankan. Berikut merupakan *State* Diagram *Login* :

|  |
| --- |
| D:\1 SEMESTER 6\KP\LAPORAN\BAB 3\GAMBAR UML\State U1.jpg\Login.jpg |
| Gambar 3.12 State Diagram Login |

Keterangan :

Dimulai dari halaman\_login() maka akan menampilkan halaman *login* dan selesai, kemudian sign\_in() maka akan menampilkan halaman utama dan selesai, kemudian sign\_out() maka akan menampilkan halaman *login* dan selesai.

### ***State* Diagram Kelola Master Data Barang**

Terdapat enam fungsi ketika sistem mengakses objek *Login*. Fungsi tersebut memicu sistem untuk melakukan *event* sesuai dengan fungsi yang telah dijalankan. Berikut merupakan *State* DiagramKelola Master Data Barang :

|  |
| --- |
| D:\1 SEMESTER 6\KP\LAPORAN\BAB 3\GAMBAR UML\State U2.jpg |
| Gambar 3.13 State Diagram Kelola Master Data Barang |

Keterangan :

Dimulai halaman\_utama() maka akan menampilkan halaman utama dan selesai, kemudian tampil\_barang() maka akan menampilkan halaman kelola master data barang dan selesai, kemudian tambah\_barang() maka akan menampilkan data barang berhasil ditambahkan dan selesai, kemudian ubah\_barang() maka akan menampilkan data barang berhasil diperbarui dan selesai, kemudian hapus\_barang() maka akan menampilkan data barang berhasil dihapus dan selesai, kemudian cari\_barang() maka akan menampilkan data barang berhasil ditemukan dan selesai.

### ***State* DiagramKelola Data Transaksi**

Terdapat tujuh fungsi ketika sistem mengakses objek *Login*. Fungsi tersebut memicu sistem untuk melakukan *event* sesuai dengan fungsi yang telah dijalankan. Berikut merupakan *State* Diagram Kelola DataTransaksi :

|  |
| --- |
| D:\1 SEMESTER 6\KP\LAPORAN\BAB 3\GAMBAR UML\State U3.jpg |
| Gambar 3.14 State Diagram Kelola Data Transaksi |

Keterangan :

Dimulai halaman\_utama() maka akan menampilkan halaman utama dan selesai, kemudian tampil\_penjualan() maka akan menampilkan halaman kelola data transaksi penjualan dan selesai, kemudian tampil\_pembelian() maka akan menampilkan halaman kelola data transaksi pembelian dan selesai, kemudian tombol\_transaksi() maka akan menampilkan tombol transaksi dan selesai, kemudian tambah\_transaksi() maka akan menampilkan data transaksi berhasil ditambahkan dan selesai, kemudian ubah\_transaksi() maka akan menampilkan data transaksi berhasil diperbarui dan selesai, kemudian hapus\_transaksi() maka akan menampilkan data transaksi berhasil dihapus dan selesai, kemudian cari\_transaksi() maka akan menampilkan data transaksi berhasil ditemukan dan selesai.

### ***State* Diagram Kelola Stok Barang**

Terdapat tujuh fungsi ketika sistem mengakses objek *Login*. Fungsi tersebut memicu sistem untuk melakukan *event* sesuai dengan fungsi yang telah dijalankan. Berikut merupakan *State* Diagram Kelola Stok Barang:

|  |
| --- |
|  |
| Gambar 3.15 State Diagram Kelola Stok Barang |

Keterangan :

Dimulai halaman\_utama() maka akan menampilkan halaman utama dan selesai, kemudian tampil\_stok\_seluruh() maka akan menampilkan halaman kelola stok barang keseluruhan dan selesai, kemudian tampil\_stok\_habis() maka akan menampilkan halaman kelola stok barang habis dan selesai, kemudian tombol\_stok() maka akan menampilkan tombol stok dan selesai, kemudian cari\_stok() maka akan menampilkan stok barang berhasil disortir dan selesai, kemudian cetak\_stok() maka stok barang tercetak dan selesai.

### ***State* Digram Kelola Laporan**

Terdapat tujuh fungsi ketika sistem mengakses objek *Login*. Fungsi tersebut memicu sistem untuk melakukan *event* sesuai dengan fungsi yang telah dijalankan. Berikut merupakan *State* Diagram Kelola Laporan:

|  |
| --- |
| D:\1 SEMESTER 6\KP\LAPORAN\BAB 3\GAMBAR UML\BARU.jpg\Kelola Laporan.jpg |
| Gambar 3.16 State Diagram Kelola Laporan |

Keterangan :

Dimulai dari halaman\_utama() maka akan menampilkan halaman utama dan selesai, kemudian tampil\_laporan() maka akan menampilkan halaman kelola laporan dan selesai, kemudian laporan\_penjualan() maka akan menampilkan laporan transaksi penjualan dan selesai, kemudian laporan\_pembelian() maka akan menampilkan laporan transaksi pembelian dan selesai, kemudian laporan\_stok() maka akan menampilkan laporan stok barang dan selesai, kemudian cetak\_laporan() maka laporan tercetak dan selesai.

### ***Sequence* Diagram**

*Sequence* Diagrammerupakan bentuk kedua dari *behavioral model*. *Sequence Diagram* menunjukan bagaimana sebuah kejadian *(event)* menyebabkan transisi dari objek ke objek. Berikut merupakan *Sequence* Diagramdari Aplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang di Nando Sport Centre Cianjur :

### ***Sequence* Diagram *Login***

*Sequence* Diagram *Login* merupakan rangkaian pesan yang terdapat didalam sistem pada saat pengguna menjalankan fungsi *Login* yang dilakukan dari aktor terhadap objek *Login*. Berikut merupakan *Sequence* Diagram *Login* dari Aplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang di Nando Sport Centre Cianjur :

|  |
| --- |
| D:\1 SEMESTER 6\KP\LAPORAN\BAB 3\GAMBAR UML\Sequence.jpg\U1 LOGIN.jpg |
| Gambar 3.17 Sequence Diagram Login |

Keterangan :

Dimulai dari sign\_in() maka sistem akan menampilkan halaman utama dan selesai.

### ***Sequence* DiagramKelola Master Data Barang**

*Sequence* DiagramKelola Master Data Barangmerupakan rangkaian pesan yang terdapat didalam sistem pada saat pengguna menjalankan fungsi Kelola Master Data Barang yang dilakukan oleh Admin. Berikut merupakan *Sequence* DiagramKelola Master Data Barangdari Aplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang di Nando Sport Centre Cianjur :

|  |
| --- |
|  |
| Gambar 3.18 Sequence Diagram Kelola Master Data Barang |

Keterangan :

Dimulai dari tambah\_barang() maka sistem akan menampilkan pesan data berhasil ditambahkan dan selesai, kemudian ubah\_barang() maka sistem akan menampilkan pesan data berhasil diperbarui dan selesai, kemudian hapus\_barang() maka sistem akan menampilkan pesan data berhasil dihapus dan selesai, kemudian cari\_barang() maka sistem akan menampilkan hasil pencarian dan selesai.

### ***Sequence Diagram* Kelola Data Transaksi (Penjualan)**

*Sequence* DiagramKelola Data Transaksimerupakan rangkaian pesan yang terdapat didalam sistem pada saat pengguna menjalankan fungsi Kelola Data Transaksi yang dilakukan oleh Admin. Berikut merupakan *Sequence* DiagramKelola Data Transaksidari Aplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang di Nando Sport Centre Cianjur :

|  |
| --- |
|  |
| Gambar 3.19 Sequence Diagram Kelola Data Transaksi (Penjualan) |

Keterangan :

Dimulai dari tambah\_transaksi() maka sistem akan menampilkan pesan data transaksi berhasil ditambahkan lalu cetak\_nota() maka sistem akan mencetak nota penjualan dan selesai, kemudian ubah\_transaksi() maka sistem akan menampilkan pesan data transaksi berhasil diperbarui dan selesai, kemudian hapus\_transaksi() maka sistem akan menampilkan pesan data transaksi berhasil dihapus dan selesai, kemudian cari\_transaksi() maka sistem akan menampilkan hasil pencarian dan selesai.

### ***Sequence* DiagramKelola DataTransaksi (Pembelian)**

*Sequence* DiagramKelola Data Transaksimerupakan rangkaian pesan yang terdapat didalam sistem pada saat pengguna menjalankan fungsi Kelola Data Transaksi yang dilakukan oleh Admin. Berikut merupakan *Sequence* DiagramKelola Data Transaksidari Aplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang di Nando Sport Centre Cianjur :

|  |
| --- |
|  |
| Gambar 3.20 Sequence Diagram Kelola Data Transaksi (Pembelian) |

Keterangan :

Dimulai dari tambah\_transaksi() maka sistem akan menampilkan pesan data transaksi berhasil ditambahkan dan selesai, kemudian ubah\_transaksi() maka sistem akan menampilkan pesan data transaksi berhasil diperbarui dan selesai, kemudian hapus\_transaksi() maka sistem akan menampilkan pesan data transaksi berhasil dihapus dan selesai, kemudian cari\_transaksi() maka sistem akan menampilkan hasil pencarian dan selesai.

### ***Sequence* DiagramKelola Stok Barang (Keseluruhan)**

*Sequence* DiagramKelola Stok Barangmerupakan rangkaian pesan yang terdapat didalam sistem pada saat pengguna menjalankan fungsi Kelola Stok Barang yang dilakukan oleh Admin. Berikut merupakan *Sequence* DiagramKelola Stok Barangdari Aplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang di Nando Sport Centre Cianjur :

|  |
| --- |
|  |
| Gambar 3.21 Sequence Diagram Kelola Stok Barang (Keseluruhan) |

Keterangan :

Dimulai dari cari\_stok() maka sistem akan menampilkan data hasil sortir dan selesai, kemudian cetak\_stok() maka sistem akan mencetak stok barang dan selesai.

### ***Sequence Diagram* Kelola Stok Barang (Habis dan Minim)**

*Sequence* DiagramKelola Stok Barangmerupakan rangkaian pesan yang terdapat didalam sistem pada saat pengguna menjalankan fungsi Kelola Stok Barang yang dilakukan oleh Admin. Berikut merupakan *Sequence* DiagramKelola Stok Barangdari Aplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang di Nando Sport Centre Cianjur :

|  |
| --- |
|  |
| Gambar 3.22 Sequence Diagram Kelola Stok Barang(Habis dan Minim) |

Keterangan :

Dimulai dari cari\_stok() maka sistem akan menampilkan data hasil sortir dan selesai, kemudian cetak\_stok() maka sistem akan mencetak stok barang dan selesai.

### ***Sequence Diagram* Kelola Laporan**

*Sequence* DiagramKelola Laporanmerupakan rangkaian pesan yang terdapat didalam sistem pada saat pengguna menjalankan fungsi Kelola Laporan yang dilakukan oleh Admin. Berikut merupakan *Sequence* DiagramKelola Stok Barang Laporandari Aplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang di Nando Sport Centre Cianjur :

|  |
| --- |
|  |
| Gambar 3.23 Sequence Diagram Kelola Laporan |

Keterangan :

Dimulai dari cetak\_laporan() maka sistem akan menampilkan form laporan yang akan dicetak dan selesai.

### ***Sequence* Diagram *Logout***

*Sequence* Diagram *Logout* merupakan rangkaian pesan yang terdapat didalam sistem pada saat pengguna menjalankan fungsi *Logout* yang dilakukan dari aktor terhadap objek *Logout*. Berikut merupakan *Sequence* Diagram *Logout* dari Aplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang di Nando Sport Centre Cianjur :

|  |
| --- |
| D:\1 SEMESTER 6\KP\LAPORAN\BAB 3\GAMBAR UML\Sequence.jpg\U6 LOGOUT.jpg |
| Gambar 3.24 Sequence Diagram Logout |

Keterangan :

Dimulai dari sign\_out() maka sistem akan menampilkan halaman login dan selesai.

## **Perancangan *Database* dan Struktur Tabel**

### **3.3.1 Model *Entity Relationship* Diagram**

*Entity Relationship* Diagram adalah suatu model yang digunakan untuk menggambarkan data dalam bentuk entitas (*entity*), atribut dan hubungan antar entitas (*relationship*). Suatu *entity* bersifat unik dan memiliki atribut sebagai pembeda dengan *entity* lainnya. Berikut merupakan *Entity Realtionship* Diagram dari Aplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang di Nando Sport Centre Cianjur :

|  |
| --- |
|  |
| Gambar 3.25 Entity Realtionship Diagram Aplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang di Nando Sport Centre Cianjur |

Keterangan :

Dalam *Entity Realtionship* Diagram di atas, diketahui terdapat 5 buah entitas utama. Terdapat entitas user, barang, djual, jual, dbeli, dan beli.

### **Strruktur Tabel**

Struktur tabel menggambarkan *detail* tabel yang berisi *field,* tipe data, panjang data dan keterangan lainnya. Berikut merupakan Struktur Tabel dari Aplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang di Nando Sport Centre Cianjur :

### **Tabel *User***

##### Tabel 3.11 Tabel *User*

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Panjang** | **Kunci** |
| *username* | varchar | 15 | *Primary key* |
| *password* | varchar | 10 |  |

### **Tabel Barang**

##### Tabel 3.12 Tabel Barang

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Panjang** | **Kunci** |
| id\_barang | varchar | 10 | *Primary key* |
| tgl | date |  |  |
| nama\_barang | varchar | 30 |  |
| jenis\_barang | varchar | 20 |  |
| kode\_jenis\_barang | varchar | 10 |  |
| nama\_merk | varchar | 20 |  |
| harga\_satuan | double | 10 |  |
| jumlah\_stok | double | 10 |  |
| status | varchar | 15 |  |
| keterangan | varchar | 80 |  |

### **Tabel DJual**

##### Tabel 3.13 Tabel DJual

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Panjang** | **Kunci** |
| no\_nota | varchar | 15 |  |
| tgl\_penjualan | date |  |  |
| jumlah\_jual | double | 15 |  |
| subtotal | double | 15 |  |
| id\_barang | varchar | 10 |  |
| nama\_barang | varchar | 30 |  |
| harga\_satuan | double | 10 |  |

### **Tabel Jual**

##### Tabel 3.14 Tabel Jual

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Panjang** | **Kunci** |
| no\_nota | varchar | 15 | *Primary key* |
| tgl\_penjualan | date |  |  |
| jumlah\_total | double | 15 |  |
| total | double | 15 |  |

### **Tabel DBeli**

##### Tabel 3.15 Tabel DBeli

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Panjang** | **Kunci** |
| no\_nota | varchar | 15 |  |
| tgl\_pembelian | date |  |  |
| jumlah\_barang | double | 10 |  |
| subtotal | double | 10 |  |
| id\_barang | varchar | 10 |  |
| nama\_barang | varchar | 30 |  |
| harga\_satuan | double | 10 |  |

### **Tabel Beli**

##### Tabel 3.16 Tabel Beli

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Panjang** | **Kunci** |
| no\_nota | varchar | 15 | *Primary key* |
| tgl\_pembelian | date |  |  |
| jumlah\_total | double | 10 |  |
| total | double | 10 |  |

## **Perancangan Struktur Menu**

Perancangan struktur menu merupakan gambaran jalur pemakaian aplikasi pengelolaan kas rutin dan penjadwalan piket anggota yang akan dibangun. Pada struktur menu aplikasi pengelolaan kas rutin dan penjadwalan piket anggota hanya terdapat satu bagian struktur menu. Berikut merupakan Struktur Menu dari Aplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang di Nando Sport Centre Cianjur :

### **Struktur Menu Admin**

|  |
| --- |
| Laporan Penjualan  Laporan Detail Penjualan  Laporan Pembelian  Logout  Laporan Detail Pembelian  Kelola Stok Barang  Kelola Laporan  Stok Barang Habis dan Minim  Stok Barang Keseluruhan  Kelola Pembelian  Kelola Penjualan  Kelola Data Transaksi  Kelola Master Data Barang  Halaman Utama Admin |
| Gambar 3.26 Struktur Menu Admin |

## **Perancangan Antar Muka**

Perancanagan antar muka menggambarkan bagaimana *software* berkomunikasi dengan sistem dan manusia yang menggunakannya. Antarmuka menggambarkan aliran informasi dan prilakunya. Berikut merupakan Perancangan Antar Muka dari Aplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang di Nando Sport Centre Cianjur :

### **Antar Muka Halaman *Login***

Halaman *Login* merupakan tampilan awal ketika Aplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang dijalankan. Admin menginputkan *username* dan *password* kemudian menekan tombol *Login* untuk masuk ke halaman utama.

|  |
| --- |
|  |
| Gambar 3.27 Antar Muka Halaman Login |

### **Antar Muka Halaman Utama (Admin)**

Halaman utama merupakan tampilan halaman utama ketika admin berhasil *login*.

|  |
| --- |
| D:\1 SEMESTER 6\KP\LAPORAN\Mockup Antar Muka Mentah\Halaman Utama.png |
| Gambar 3.28 Antar Muka Halaman Utama |

### **Antar Muka Halaman Kelola Master Data Barang**

Halaman Kelola Master Data Barang merupakan halaman yang digunakan untuk menampilkan informasi data barang. Selain menampilkan informasi, di halaman ini juga dapat melakukan tambah, ubah, hapus, dan cari data barang.

|  |
| --- |
|  |
| Gambar 3.29 Antar Muka Kelola Master Data Barang |

### **Antar Muka Halaman Tambah Data Barang**

Halaman Tambah Data Barang merupakan halaman yang digunakan untuk menginputkan data Barang baru.

|  |
| --- |
| D:\1 SEMESTER 6\KP\LAPORAN\Mockup Antar Muka Mentah\Tambah Data Barang.png |
| Gambar 3.30 Antar Muka Halaman Tambah Data Barang |

### **Antar Muka Halaman Ubah Data Barang**

Halaman Ubah Data Barang merupakan halaman yang digunakan untuk mengubah data Barang.

|  |
| --- |
| D:\1 SEMESTER 6\KP\LAPORAN\Mockup Antar Muka Mentah\Ubah Data Barang.png |
| Gambar 3.31 Antar Muka Halaman Ubah Data Barang |

### **Antar Muka Halaman Hapus Data Barang**

Halaman Hapus Data Barang merupakan halaman yang digunakan untuk menghapus data Barang.

|  |
| --- |
|  |
| Gambar 3.32 Antar Muka Halaman Hapus Data Barang |

### **Antar Muka Halaman Kelola Data Transaksi (Penjualan)**

Halaman Kelola Transaksi Penjualan merupakan halaman yang digunakan untuk menampilkan informasi data transaksi penjualan. Selain menampilkan informasi, di halaman ini juga dapat melakukan tambah, ubah, hapus, dan cari data transaksi penjualan.

|  |
| --- |
|  |
| Gambar 3.33 Antar Muka Kelola Data Transaksi Penjualan |

### **Antar Muka Halaman Tambah Data Transaksi (Penjualan)**

Halaman Tambah Data Transaksi Penjualan merupakan halaman yang digunakan untuk menginputkan data Transaksi Penjualan.

|  |
| --- |
|  |
| Gambar 3.34 Antar Muka Halaman Tambah Data Transaksi Penjualan |

### **Antar Muka Halaman Tombol Pilihan Data Barang**

Halaman Pilihan Data Barang untuk Penjualan dan Pembelian. Merupakan halaman yang digunakan untuk mengisi data barang otomatis di form tambah data transaksi

|  |
| --- |
|  |
| Gambar 3.35 Antar Muka Halaman Pilihan Penjualan/Pembelian |

### **Antar Muka Halaman Ubah Data Transaksi (Penjualan)**

Halaman Ubah Data Transaksi Penjualan merupakan halaman yang digunakan untuk mengubah data Transaksi Penjualan.

|  |
| --- |
|  |
| Gambar 3.36 Antar Muka Halaman Ubah Data Transaksi Penjualan |

### **Antar Muka Halaman Hapus Data Transaksi (Penjualan)**

Halaman Hapus Data Transaksi Penjualan merupakan halaman yang digunakan untuk menghapus data Transaksi Penjualan.

|  |
| --- |
|  |
| Gambar 3.37 Antar Muka Halaman Hapus Data Transaksi Penjualan |

### **Antar Muka Halaman Kelola Data Transaksi (Pembelian)**

Halaman Kelola Transaksi Pembelian merupakan halaman yang digunakan untuk menampilkan informasi detail data transaksi pembelian. Selain menampilkan informasi, di halaman ini juga dapat melakukan tambah, ubah, hapus, dan cari data transaksi pembelian.

|  |
| --- |
|  |
| Gambar 3.38 Antar Muka Kelola Data Transaksi Pembelian |

### **Antar Muka Halaman Tambah Data Transaksi (Pembelian)**

Halaman Tambah Data Transaksi Pembelian merupakan halaman yang digunakan untuk menginputkan data Transaksi Pembelian.

|  |
| --- |
|  |
| Gambar 3.39 Antar Muka Halaman Tambah Data Transaksi Pembelian |

### **Antar Muka Halaman Ubah Data Transaksi (Pembelian)**

Halaman Ubah Data Transaksi Pembelian merupakan halaman yang digunakan untuk mengubah data Transaksi Pembelian.

|  |
| --- |
|  |
| Gambar 3.40 Antar Muka Halaman Ubah Data Transaksi Pembelian |

### **Antar Muka Halaman Hapus Data Transaksi (Pembelian)**

Halaman Hapus Data Transaksi Pembelian merupakan halaman yang digunakan untuk menghapus data Transaksi Pembelian.

|  |
| --- |
|  |
| Gambar 3.41 Antar Muka Halaman Hapus Data Transaksi Pembelian |

### **Antar Muka Halaman Kelola Stok Barang (Keseluruhan)**

Halaman Kelola Stok Barang Keseluruhan merupakan halaman yang digunakan untuk menampilkan informasi stok barang keseluruhan. Selain menampilkan informasi, di halaman ini juga dapat melakukan cari dan cetak data stok barang.

|  |
| --- |
|  |
| Gambar 3.42 Antar Muka Kelola Stok Barang Keseluruhan |

### **Antar Muka Halaman Kelola Stok Barang (Habis dan Minim)**

Halaman Kelola Stok Barang Habis dan Minim merupakan halaman yang digunakan untuk menampilkan informasi stok barang habis dan minim. Selain menampilkan informasi, di halaman ini juga dapat melakukan cari dan cetak data stok barang.

|  |
| --- |
|  |
| Gambar 3.43 Antar Muka Kelola Stok Barang Habis dan Minim |

### **Antar Muka Halaman Kelola Laporan**

Halaman Kelola Laporan merupakan halaman yang digunakan untuk menampilkan pilihan laporan yang tersedia dan juga untuk mencetak laporan

|  |
| --- |
|  |
| Gambar 3.44 Antar Muka Kelola Laporan |

### **Antar Muka Halaman *Logout***

Halaman *Logout* merupakan halaman konfirmasi admin apakan ingin keluar dari aplikasi atau tidak.

|  |
| --- |
|  |
| Gambar 3.45 Antar Muka Halaman Logout |

## **Perancangan Hasil Cetak**

Berikut merupakan Perancangan Hasil Cetak dari Aplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang di Nando Sport Centre Cianjur :

### **Hasil Cetak Stok Barang (Keseluruhan)**

Hasil cetak Stok Barang Keseluruhan merupakan hasil cetak ketika admin menekan Cetak Stok di halaman Kelola Stok Barang Keseluruhan. Berisikan informasi mengenai stok barang keseluruhan.

|  |
| --- |
|  |
| Gambar 3.46 Hasil Cetak Stok Barang Keseluruhan |

### **Hasil Cetak Stok Barang (Habis dan Minim)**

Hasil cetak Stok Barang Habis dan Minim merupakan hasil cetak ketika admin menekan Cetak Stok di halaman Kelola Stok Barang Habis dan Minim. Berisikan informasi mengenai stok barang yang habis dan minim.

|  |
| --- |
|  |
| Gambar 3.47 Hasil Cetak Stok Barang Habis dan Minim |

### **Hasil Cetak Laporan Transaksi (Detail Penjualan)**

Hasil cetak Laporan Detail Transaksi Penjualan merupakan hasil cetak ketika admin menekan Cetak Laporan di halaman Kelola Laporan. Berisikan informasi mengenai pengelolaan detail data transaksi penjualan.

|  |
| --- |
|  |
| Gambar 3.48 Hasil Cetak Laporan Detail Transaksi Penjualan |

### **Hasil Cetak Laporan Transaksi (Penjualan)**

Hasil cetak Laporan Transaksi Penjualan merupakan hasil cetak ketika admin menekan Cetak Laporan di halaman Kelola Laporan. Berisikan informasi mengenai pengelolaan data transaksi penjualan.

|  |
| --- |
|  |
| Gambar 3.49 Hasil Cetak Laporan Transaksi Penjualan |

### **Hasil Cetak Laporan Transaksi (Detail Pembelian)**

Hasil cetak Laporan Detail Transaksi Pembelian merupakan hasil cetak ketika admin menekan Cetak Laporan di halaman Kelola Laporan. Berisikan informasi mengenai pengelolaan detail data transaksi pembelian.

|  |
| --- |
|  |
| Gambar 3.50 Hasil Cetak Laporan Detail Transaksi Pembelian |

### **Hasil Cetak Laporan Transaksi (Pembelian)**

Hasil cetak Laporan Transaksi Pembelian merupakan hasil cetak ketika admin menekan Cetak Laporan di halaman Kelola Laporan. Berisikan informasi mengenai pengelolaan data transaksi pembelian.

|  |
| --- |
|  |
| Gambar 3.51 Hasil Cetak Laporan Transaksi Pembelian |

### **Hasil Cetak Nota Transaksi Penjualan**

Hasil cetak Nota Transaksi Penjualan merupakan hasil cetak ketika admin selesai menyimpan data transkasi di halaman Kelola Transaksi Penjualan. Berisikan informasi mengenai detail transaksi penjualan.

|  |
| --- |
|  |
| Gambar 3.52 Hasil Cetak Nota Transaksi Penjualan |

## **Perancangan Pesan**

### **Pesan *Login* Berhasil**

Pesan *Login* Berhasil merupakan pesan yang ditampilkan ketika admin berhasil *Login* ke dalam aplikasi.

|  |
| --- |
|  |
| Gambar 3.53 Pesan Login Berhasil |

### **Pesan *Login* Gagal**

Pesan *Login* Gagal merupakan pesan yang ditampilkan ketika admin gagal *Login* ke dalam aplikasi diakibatkan oleh *username* atau *password* yang salah.

|  |
| --- |
|  |
| Gambar 3.54 Pesan Login Gagal |

### **Pesan Data Berhasil Di Tambahkan**

Pesan Data Berhasil Di Tambahkan merupakan pesan informasi untuk admin bahwa data yang diinputkan berhasil di simpan dan di tambahkan.

|  |
| --- |
|  |
| Gambar 3.55 Pesan Data Berhasil Di Tambahkan |

### **Pesan Data Berhasil Di Perbarui**

Pesan Data Berhasil Di Perbarui merupakan pesan informasi untuk admin bahwa data yang diubah berhasil di perbarui.

|  |
| --- |
|  |
| Gambar 3.56 Pesan Data Berhasil Di Perbarui |

### **Pesan Data Berhasil Di Hapus**

Pesan Data Berhasil Di Hapus merupakan pesan informasi untuk admin bahwa data yang di pilih berhasil di hapus.

|  |
| --- |
|  |
| Gambar 3.57 Pesan Data Berhasil Di Hapus |

### **Pesan Data *Field* Masih Kosong**

Pesan Data *Field* Masih Kosong merupakan pesan informasi untuk admin bahwa *form* yang diisi belum lengkap atau ada *field* yang masih kosong.

|  |
| --- |
|  |
| Gambar 3.58 Pesan Data Field Masih Kosong |

### **Pesan Data Konfirmasi Simpan**

Pesan Data Konfirmasi Simpan merupakan pesan konfirmasi untuk admin bahwa *form* akan disimpan atau tidak.

|  |
| --- |
|  |
| Gambar 3.59 Pesan Data Konfirmasi Simpan |

### **Pesan Data Konfirmasi Cetak Nota Penjualan**

Pesan Data Konfirmasi Cetak Nota Penjualan merupakan pesan konfirmasi untuk admin bahwanota transaksi penjualan akan di cetak atau tidak.

|  |
| --- |
|  |
| Gambar 3.60 Pesan Data Konfirmasi Simpan |

### **Pesan Data Yang Anda Cari Di Temukan**

Pesan Data Yang Anda Cari Di Temukan merupakan pesan informasi untuk admin bahwa pencarian atausortir yang di lakukan berhasil.

|  |
| --- |
|  |
| Gambar 3.61 Data Yang Anda Cari Di Temukan |

### **Pesan Data Tidak Di Temukan**

Pesan Data Yang Anda Cari Tidak Di Temukan merupakan pesan informasi untuk admin bahwapencarian atau sortir yang di lakukan gagal.

|  |
| --- |
|  |
| Gambar 3.62 Pesan Data Tidak Di Temukan |

# 

# **BAB IV IMPLEMENTASI DAN PENGUJIAN**

## **4.1 Implementasi**

Tahap implementasi dapat diterapkan jika sistem perangkat lunak dan perangkat kerasnya telah dipersiapkan. Kegiatan implementasi sistem ini meliputi kebutuhan perangkat lunak, perangkat keras, instalasi aplikasi dan implementasi antarmuka serta pengujian aplikasi sistem yang telah dirancang sesuai dengan kebutuhan organisasi.

### **Kebutuhan Perangkat Keras Pembangun**

Spesifikasi perangkat keras yang digunakan untuk membangunAplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang di Nando Sport Centre Cianjur diantaranya sebagai berikut :

##### Tabel 4.1 Daftar Perangkat Keras Pembangun

|  |  |
| --- | --- |
| **No.** | **Perangkat Keras** |
| 1. | Monitor LCD 14 Inchi |
| 2. | VGA NVIDIA GEFORCE 930M 2GB |
| 3. | Hardisk 500 GB |
| 4. | Processor Intel Core I3 |
| 5. | Memory RAM 2GB |
| 6. | Mouse |
| 7. | Keyboard |

### **Kebutuhan Perangkat Lunak Pembangun**

Perangkat lunak yang digunakan untuk membangun membangunAplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang di Nando Sport Centre Cianjur diantaranya sebagai berikut :

##### Tabel 4.2 Daftar Perangkat Lunak Pembangun

|  |  |
| --- | --- |
| **No.** | **Perangkat Lunak** |
| 1. | Microsoft Windows 7 32-bit |
| 2. | Delphi XE8 |
| 3. | Mysql Driver ODBC 32-bit |
| 4. | DBMS XAMPP |
| 5. | Balseamiq Mockups 3 |
| 6. | Visual Paradigm |

### **Implementasi Aplikasi**

### **Implementasi Halaman *Login***

|  |
| --- |
|  |
| Gambar 4.1 Implementasi Halaman Login |

### **Implementasi Halaman Utama (Admin)**

|  |
| --- |
|  |
| Gambar 4.2 Implementasi Halaman Utama Admin |

### **Implementasi Halaman Kelola Master Data Barang**

|  |
| --- |
|  |
| Gambar 4.3 Implementasi Halaman Kelola Master Data Barang |

### **Implementasi Halaman Tambah Master Data**

|  |
| --- |
|  |
| Gambar 4.4 Implementasi Halaman Tambah Master Data |

### **Implementasi Halaman Ubah Master Data**

|  |
| --- |
|  |
| Gambar 4.5 Implementasi Halaman Ubah Master Data |

### **Implementasi Hapus Master Data**

|  |
| --- |
|  |
| Gambar 4.6 Implementasi Hapus Master Data |

### **Implementasi Halaman Kelola Transaksi Penjualan**

|  |
| --- |
|  |
| Gambar 4.7 Implementasi Halaman Kelola Transaksi Penjualan |

### **Implementasi Halaman Tambah Transaksi Penjualan**

|  |
| --- |
|  |
| Gambar 4.8 Implementasi Halaman Tambah Transaksi Penjualan |

### **Implementasi Halaman Pilihan Barang untuk Transaksi**

|  |
| --- |
|  |
| Gambar 4.9 Implementasi Halaman Pilihan Barang untuk Transaksi |

### **Implementasi Halaman Ubah Transaksi Penjualan**

|  |
| --- |
|  |
| Gambar 4.10 Implementasi Halaman Ubah Transaksi Penjualan |

### **Implementasi Hapus Transaksi Penjualan**

|  |
| --- |
|  |
| Gambar 4.11 Implementasi HapusTransaksi Penjualan |

### **Implementasi Halaman Kelola Transaksi Pembelian**

|  |
| --- |
|  |
| Gambar 4.12 Implementasi Halaman Detail Transaksi Pembelian |

### **Implementasi Halaman Tambah Transaksi Pembelian**

|  |
| --- |
|  |
| Gambar 4.13 Implementasi Halaman Tambah Transaksi Pembelian |

### **Implementasi Halaman Ubah Transaksi Pembelian**

|  |
| --- |
|  |
| Gambar 4.14 Implementasi Halaman UbahTransaksi Pembelian |

### **Implementasi Hapus Transaksi Pembelian**

|  |
| --- |
|  |
| Gambar 4.15 Implementasi Hapus Transaksi Pembelian |

### **Implementasi Halaman Stok Barang Keseluruhan**

|  |
| --- |
|  |
| Gambar 4.16 Implementasi Halaman Stok Barang Keseluruhan |

### **Implementasi Halaman Stok Habis dan Minim**

|  |
| --- |
|  |
| Gambar 4.17 Implementasi Halaman Stok Habis dan Minim |

### **Implementasi Halaman Laporan**

|  |
| --- |
|  |
| Gambar 4.18 Implementasi Halaman Laporan |

### **Implementasi Halaman *Logout***

|  |
| --- |
|  |
| Gambar 4.19 Implementasi Halaman Logout |

### **Implementasi Hasil Cetak Laporan Stok Keseluruhan**

|  |
| --- |
|  |
| Gambar 4.20 Implementasi Hasil Cetak Laporan Stok Keseluruhan |

### **Implementasi Hasil Cetak Laporan Stok Habis dan Minin**

|  |
| --- |
|  |
| Gambar 4.21 Implementasi Hasil Cetak Laporan Habis dan Minim |

### **Implementasi Hasil Cetak Laporan Penjualan**

|  |
| --- |
|  |
| Gambar 4.22 Implementasi Hasil Cetak Laporan Penjualan |

### **Implementasi Hasil Cetak Laporan Detail Penjualan**

|  |
| --- |
|  |
| Gambar 4.23 Implementasi Hasil Cetak Laporan Detail Penjualan |

### **Implementasi Hasil Cetak Laporan Pembelian**

|  |
| --- |
|  |
| Gambar 4.24 Implementasi Hasil Cetak Laporan Pembelian |

### **Implementasi Hasil Cetak Laporan Detail Pembelian**

|  |
| --- |
|  |
| Gambar 4.25 Implementasi Hasil Cetak Laporan Detail Pembelian |

### **Implementasi Hasil Cetak Nota Penjualan**

|  |
| --- |
|  |
| Gambar 4.26 Implementasi Hasil Cetak Nota Penjualan |

### **Implementasi Pesan *Login* Berhasil**

|  |
| --- |
|  |
| Gambar 4.27 Implementasi Pesan Login Berhasil |

### **Implementasi Pesan *Login* Gagal**

|  |
| --- |
|  |
| Gambar 4.28 Implementasi Pesan Login Gagal |

### **Implementasi Pesan Data Berhasil Di Tambahkan**

|  |
| --- |
|  |
| Gambar 4.29 Implementasi Pesan Data Berhasil Di Tambahkan |

### **Implementasi Pesan Data Berhasil Di Perbarui**

|  |
| --- |
|  |
| Gambar 4.30 Implementasi Pesan Data Berhasil Di Perbarui |

### **Implementasi Pesan Data Berhasil Di Hapus**

|  |
| --- |
|  |
| Gambar 4.31 Implementasi Pesan Data Berhasil Di Hapus |

### **Implementasi Pesan Data Konfirmasi Simpan**

|  |
| --- |
|  |
| Gambar 4.32 Implementasi Pesan Data Konfirmasi Simpan |

### **Implementasi Pesan Data Konfirmasi Cetak Nota Penjualan**

|  |
| --- |
|  |
| Gambar 4.33 Implementasi Pesan Data Konfirmasi Cetak Nota Penjualan |

### **Implementasi Pesan Data Field Masih Kosong**

|  |
| --- |
|  |
| Gambar 4.34 Implementasi Pesan Data Field Masih Kosong |

### **Implementasi Pesan Data Yang Di Cari Di Temukan**

|  |
| --- |
|  |
| Gambar 4.35 Implementasi Pesan Data Yang Anda Cari Di Temukan |

### **Implementasi Pesan Data Tidak Di Temukan**

|  |
| --- |
|  |
| Gambar 4.36 Implementasi Pesan Data Tidak Di Temukan |

### ***Source CodePprogram***

Terlampir.

## **4.2 Pengujian**

Pengujian merupakan bagian yang penting dalam siklus pengembangan atau pembuatan perangkat lunak. Pengujian dilakukan untuk menjamin kualitas dan juga mengetahui kelemahan dari perangkat lunak. Tujuan dari pengujian ini adalah untuk menjamin bahwa perangkat lunak yang dibangun memiliki kualitas yang handal. Pengujian perangkat lunak ini menggunakan metode *black box.*Pengujian *black box* berfokus pada persyaratan fungsional perangkat lunak yang dibuat.

### **4.2.1 Kasus Dan Hasil Pengujian**

##### Tabel 4.3. Hasil Pengujian

| No. | Halaman | Pengujian | Hasil | Kesesuaian |
| --- | --- | --- | --- | --- |
| 1. | *Login* | *Login* sebagai Admin dan menekan Tombol *Login* | Menampilkan Halaman Utama | Sesuai |
| Menginput tidak sesuai *username* atau *password* | Menampilkan Pesan Kesalahan | Sesuai |
| Klik *Checkbox* Lihat *Password* | Menampilkan *Password* | Sesuai |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 2. | Halaman  Utama | Menekan Tombol Kelola Master Data Barang | Menampilkan Halaman Kelola Master Data Barang | Sesuai |
| Menekan Tombol Kelola Transaksi | Menampilkan Halaman Kelola Transaksi | Sesuai |
| Menekan Tombol Kelola Stok | Menampilkan Halaman Kelola Stok | Sesuai |
| Menekan Tombol Kelola Laporan | Menampilkan Halaman Kelola Laporan | Sesuai |
| Menekan Tombol *Logout* | Menampilkan Pesan Konfirmasi | Sesuai |
| 3. | Halaman Kelola Transaksi (Penjualan) | Menekan Tombol Tambah | Menampilkan *form* Tambah Transaksi Penjualan | Sesuai |
| Menekan Tombol Ubah | Menampilkan *form* Ubah Transaksi Penjualan | Sesuai |
| Menekan Tombol Hapus | Menampilkan Pesan Konfirmasi | Sesuai |
| Menekan Tombol Simpan | Menyimpan data yang ditambah | Sesuai |
| Menekan Tombol Kembali | Proses tambah atau ubah data dibatalkan | Sesuai |
| Menginput Data Kosong di Form Tambah Transaksi Penjualan | Menampilkan pesan kesalahan | Sesuai |
| Menginput Kata Kunci Pencarian yang sesuai dan Menekan Tombol Cari | Pencarian Berhasil dan Menampilkan Hasil Pencarian | Sesuai |
| Menekan Tombol *Refresh* | Data Tabel Berhasil Dimuat Ulang | Sesuai |
| Menekan Tombol Kembali | Kembali ke Halaman Utama | Sesuai |
| Menekan Konfirmasi Iya Cetak Nota Penjualan | Mencetak Nota Penjualan | Sesuai |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 4. | Halaman Kelola Transaksi (Detail Pembelian) | Menekan Tombol Tambah | Menampilkan *form* Tambah Transaksi Penjualan | Sesuai |
| Menekan Tombol Ubah | Menampilkan *form* Ubah Transaksi Penjualan | Sesuai |
| Menekan Tombol Hapus | Menampilkan Pesan Konfirmasi | Sesuai |
| Menekan Tombol Simpan | Menyimpan data yang ditambah | Sesuai |
| Menekan Tombol Kembali | Proses tambah atau ubah data dibatalkan | Sesuai |
| Menginput Data Kosong di Form Tambah Transaksi Penjualan | Menampilkan pesan kesalahan | Sesuai |
| Menginput Kata Kunci Pencarian yang sesuai dan Menekan Tombol Cari | Pencarian Berhasil dan Menampilkan Hasil Pencarian | Sesuai |
| Menekan Tombol *Refresh* | Data Tabel Berhasil Dimuat Ulang | Sesuai |
| Menekan Tombol Kembali | Kembali ke Halaman Utama | Sesuai |
| 5. | Halaman Stok Barang Keseluruhan | Menekan *Combobox* Cari | Menampilkan Pilihan Kategori Cari | Sesuai |
| Menginput Kata Kunci Pencarian yang sesuai dan Menekan Tombol Cari | Pencarian Berhasil dan Menampilkan Hasil Cari | Sesuai |
| Menekan Tombol *Refresh* | Data Tabel Berhasil Dimuat Ulang | Sesuai |
| Menginput Kata Kunci Kosong dan Menekan Tombol Cari | Menampilkan Pesan Kesalahan | Sesuai |
| Menekan Tombol Kembali | Kembali ke Halaman Utama | Sesuai |
| Menekan tombol Cetak Stok | Menampilkan halaman stok barang keseluruhan siap cetak | Sesuai |
| Menekan tombol *Print* | Mencetak stok barang keseluruhan | Sesuai |
| 6. | Halaman Stok Barang Habis dan Minim | Menekan *Combobox* Cari | Menampilkan Pilihan Kategori Cari | Sesuai |
| Menginput Kata Kunci Pencarian yang sesuai dan Menekan Tombol Cari | Pencarian Berhasil dan Menampilkan Hasil Cari | Sesuai |
| Menekan Tombol *Refresh* | Data Tabel Berhasil Dimuat Ulang | Sesuai |
| Menginput Kata Kunci Kosong dan Menekan Tombol Sortir | Menampilkan Pesan Kesalahan | Sesuai |
| Menekan Tombol Kembali | Kembali ke Halaman Utama | Sesuai |
| Menekan tombol Cetak Stok | Menampilkan halaman stok barang habis dan minim siap cetak | Sesuai |
| Menekan tombol *Print* | Mencetak stok barang habis dan minim | Sesuai |
| 7. | Laporan | Menekan *Combobox* Pilihan Laporan | Menampilkan Pilihan Laporan yang dapat dicetak | Sesuai |
| Melakukan Sortir Tanggal Laporan | Laporan berhasil di sortir sesuai tanggal yang dipilih | Sesuai |
| Menekan tombol Cetak Laporan | Menampilkan halaman Laporan siap cetak | Sesuai |
| Menekan tombol *Print* | Mencetak Laporan | Sesuai |
| Menekan tombol Kembali | Kembali ke Halaman Utama | Sesuai |
| 8. | *Logout* | Menekan tombol *Logout* | Menampilkan pesan konfirmasi | Sesuai |
| Menekan Tombol Ya | Keluar Dari Halaman Aplikasi | Sesuai |
| Menekan Tombol No | Kembali Ke Halaman Utama | Sesuai |

# **BAB V** **PENUTUP**

## **Simpulan**

Berdasarkan hasil analisis selama melaksanakan Kerja Praktek di Nando Sport Centre Cianjur yang berjudul Aplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang, dapat disimpulkan bahwa :

1. Telah dibuat sebuah aplikasi pengelolaan data transaksi penjualan dan stok barang di Nando Sport Centre Cianjur.
2. Telah memudahkan pengguna dalam mengelola data transaksi penjualan dan pembelian data barang di Nando Sport Centre Cianjur.
3. Telah mempermudah pengguna dalam mengelola stok barang menjadi terorganisir di Nando Sport Centre Cianjur.
4. Telah memudahkan pengguna melakukan pembuatan laporan transaksi penjualan dan pembelian dengan cepat dan efektif.

## **Saran**

Untuk meningkatkan kinerja Aplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang, maka penulis mengajukan beberapa saran kepada pihak yang akan mengembangkan sebagai pengembang selanjutnya, antara lain :

1. Akan lebih baik jika didalam sistem ini terdapat fungsi untuk mengelola pegawai, dan penggajian agar lebih lengkap.
2. Sitem ini berbasis desktop, jadi alangkah lebih baik jika aplikasi ini dikembangkan menjadi sebuah *website* yang mendukung penjualan barang secara *online* yang dapat mempermudah para pelanggan untuk dapat membeli barang.
3. Aplikasi Pengelolaan Data Transaksi Penjualan dan Stok Barang dapat dikembangkan untuk menunjang kinerja proses bisnis ke arah yang efektif dan efisien

# **DAFTAR PUSTAKA**

Al -Bahra Bin Ladjamudin. (2008). *Analisa dan Desain Sistem Informasi.* Yogyakarta: Graha Ilmu.

Bastian, I. (2007:27). *Sistem Akuntansi Sektor Publik.* Jakarta: Salemba Empat.

Begg, T. C. (2010). *Database System: A Practical Approach.* America: Pearson Education.

Fathansyah. (2002). *Basis Data.* Bandung: Informatika.

Hartono, J. (2004). *Pengenalan Komputer.* Yogyakarta: Andi.

Haviludin. (2011). *Memahami Penggunaan UML.* Samarinda: Vol 6 No 1.

Jogiyanto. (2005). *Analisis dan Desain Sistem Informasi.* Yogyakarta: Andi.

Joy, L. A. (2011). *Desktop Apps vs Web Apps: Which Should You Choose?*

Kani, Firmansyah, & Sufandi. (2010). *Pemrograman Database menggunakan Delphi.* Graha Ilmu: Jakarta.

Kenneth C. Laudon, J. P. (2018). *Sistem Informasi Manajemen.* Jakarta: Salemba4.

Konixbam. (2009). *Web Based Versus Desktop Based Application.*

Pramana, H. W. (2010). *Definisi Aplikasi.* Diambil kembali dari wikipedia.org/wiki/Aplikasi.

Pressman, R. S. (2010). *Pendekatan Praktisi Perangkat Lunak Edisi 7.* Yogyakarta: Andi.

Ristono. (2009). *Manajemen Persediaan.* Graha Ilmu.

Rosari, W. R. (2008). *Php dan MySQL untuk Pemula.* Yogyakarta: Andi.

Rouf, A. (2005). *Pengujian Perangkat Lunak Menggunakan Metode Blackbox dan Whitebox.*

Sibero, A. (2011). *Kitab Suci Web Programming.* Yogyakarta: Mediakom.

Sutarto, H. R., & Sutarto, I. M. (2009). *Mastering Java.* Jakarta: Elex Media Komputindo.

Turban, M. W. (1999). *Pengenalan Sistem Informasi.* Yogyakarta: Andi.

Winardi, J. (2005:26). *Manajemen Perubahan (The Management of Change).* Jakarta: Prenada Media.

Yasin, V. (2012). *Rekayasa Perangkat Lunak Berorientasi Objek.* Bogor: Mitrawacanamedia.

# **DAFTAR RIWAYAT HIDUP**



* 1. **Data Personal**

NPM : 5520116131

Nama : Mufti Akhmad Rilmansyah

Tempat / Tgl. Lahir : Cianjur, 09 Juni 1997

Jenis Kelamin : Laki-laki

Agama : Islam

Program Studi : Teknik Informatika

Alamat Rumah : Gg. Laksana IV Kelurahan Sayang Kecamatan Cianjur

Telp : 082246432163

* 1. **Pendidikan**

|  |  |  |  |
| --- | --- | --- | --- |
| **Jenjang** | **Nama Lembaga** | **Program Studi** | **Tahun Lulus** |
| SD | SDN Ibu Dewi 6 Cianjur |  | 2009 |
| SMP | SMPN 2 Cianjur |  | 2012 |
| SMA | SMAN 2 Cianjur | Ilmu Pengetahuan Alam | 2015 |
| PT | Universitas Suryakancana | Teknik Informatika |  |

Demikian daftar riwayat hidup ini dibuat dengan sebenarnya.

Cianjur, 27 April 2019

Mahasiswa Ybs.

**MUFTI AKHMAD R**

# 

# **DAFTAR LAMPIRAN**

* + 1. Lampiran Penilaian Kerja Praktek
    2. Lampiran *Source Code Program*

**LEMBAR PENILAIAN KERJA PRAKTEK**

**(OLEH PEMBIMBING LAPANGAN)**

Pembimbing Lapangan yang bertandatangan di bawah ini telah melaksanakan penilaian terhadap kegiatan Kerja Praktek untuk Mahasiswa Program Studi Teknik Informatika :

|  |  |  |
| --- | --- | --- |
| Nama  NPM  Program Studi  Judul | :  :  :  : | MUFTI AKHMAD RILMANSYAH  5520116131  TEKNIK INFORMATIKA  **APLIKASI PENGELOLAAN DATA TRANSAKSI PENJUALAN DAN STOK BARANG BERBASIS DESKTOP DI TOKO NANDO SPORT CENTRE CIANJUR** |

Dengan penilaian KP, sebagai berikut :

|  |  |  |
| --- | --- | --- |
| **NO.** | **PENILAIAN** | **NILAI** |
| 1. | Disiplin |  |
| 2. | Sikap |  |
| 3. | Pelaksanaan Tugas |  |
| 4. | Produk yang dihasilkan |  |
| **Rata-rata** | |  |

**Keterangan Nilai :**

1. Sangat baik = 80 s/d 100

2. Baik = 70 s/d 79

Cianjur, 27 April 2019

Pembimbing Lapangan,

AN. Nando Sport Centre Cianjur

**Vincensius Aditya**

3. Cukup = 55 s/d 69

4. Kurang = 40 s/d 54

5. Buruk = < 39