

**SISTEM INFORMASI EKSEKUTIF PENJUALAN KERAJINAN  
TAS ROTAN PADA BALINESE CLASSIC MENGGUNAKAN  
FRAMEWORK LARAVEL**

**TUGAS AKHIR**



MADE YOGI NUGRAHA (160030638)  
**INSTITUT TEKNOLOGI DAN BISNIS  
STIKOM BALI**

**INSTITUT TEKNOLOGI DAN BISNIS  
(ITB) STIKOM BALI**

**2019**



**PERSETUJUAN  
SEMINAR DAN SIDANG TUGAS AKHIR**

**SISTEM INFORMASI EKSEKUTIF PENJUALAN KERAJINAN  
TAS ROTAN PADA BALINESE CLASSIC MENGGUNAKAN  
*FRAMEWORK LARAVEL***

Oleh:

**MADE YOGI NUGRAHA (160030638)**

Dosen Pembimbing

Dian Pramana, S.Kom., M.Kom

Lilis Yuningsih, SH., M.Kom

Tanda Tangan

Tanggal

..... .....

..... .....

**INSTITUT TEKNOLOGI DAN BISNIS  
STIKOM BALI**

Denpasar, .....

Mengetahui,

Ketua Program Studi Sistem Informasi

Ricky Aurelius Nurtanto Diaz, S.Kom., M.T



## DAFTAR ISI

PERSETUJUAN SEMINAR DAN SIDANG TUGAS AKHIR .....	i
DAFTAR ISI.....	iii
DAFTAR TABEL .....	v
DAFTAR GAMBAR .....	ix
BAB I PENDAHULUAN.....	1
1.1    Latar Belakang .....	1
1.2    Rumusan Masalah .....	2
1.3    Tujuan Penelitian .....	2
1.4    Manfaat Penelitian .....	2
1.5    Ruang lingkup Penelitian .....	3
1.6    Sistematika Penulisan.....	4
BAB II TINJAUAN PUSTAKA .....	7
2.1 <i>State of The Art</i> .....	7
2.2 <i>Balinese Classic</i> .....	9
2.3    Sistem Informasi Eksekutif (SIE) .....	12
2.4    Tujuan Sistem Informasi Eksekutif .....	12
2.5 <i>Website</i> .....	13
2.6 <i>Data Flow Diagram (DFD)</i> .....	14
2.7 <i>Entity Relationship Diagram (ERD)</i> .....	16
2.8 <i>PHP</i> .....	17
2.9 <i>MariaDB</i> .....	18
2.10 <i>Laravel</i> .....	19
2.11 <i>Bootstrap</i> .....	21
2.12 <i>Blackbox Testing</i> .....	21
2.13 <i>Model Waterfall</i> .....	22
BAB III METODE PENELITIAN .....	25

3.1	Waktu dan Lokasi Penelitian .....	25
3.2	Alat dan bahan Penelitian .....	25
3.3	Metode Pengumpulan Data.....	26
3.4	Metode Perekayasaan.....	26
3.5	Penulisan Laporan.....	27
	BAB IV HASIL DAN PEMBAHASAN.....	29
4.1	Hasil Analisis .....	29
4.2	Perancangan Sistem .....	33
4.3	Perancangan Basis Data.....	44
4.4	Perancangan Antarmuka.....	50
4.4.1	Desain Antarmuka Admin.....	50
4.4.2	Desain Antarmuka Kasir.....	64
4.4.3	Desain Antarmuka Gudang .....	72
4.5	Implementasi Sistem .....	80
4.5.1	Tampilan Untuk Admin .....	80
4.5.2	Tampilan Untuk Kasir .....	94
4.5.3	Tampilan Untuk Gudang .....	101
4.6	Pengujian Black Box.....	108
4.6.1	Pengujian Halaman Admin.....	108
4.6.2	Pengujian Halaman Kasir .....	131
4.6.3	Pengujian Halama Gudang .....	141
	BAB V PENUTUP .....	157
5.1	Kesimpulan .....	157
5.2	Saran .....	157
	DAFTAR PUSTAKA .....	159

## DAFTAR TABEL

Tabel 2.1 State of The Art.....	7
Tabel 2.2 Simbol-simbol DFD.....	15
Tabel 2.3 Simbol-simbol ERD.....	16
Tabel 4.1 Analisa <i>User</i> .....	31
Tabel 4.2 <i>Mapping</i> Hak Akses.....	31
Tabel 4.3 Analisa Data.....	32
Tabel 4.4 Analisa Proses .....	33
Tabel 4.5 Struktur Tabel <i>Users</i> .....	46
Tabel 4.6 Struktur Tabel <i>Customers</i> .....	46
Tabel 4.7 Struktur Tabel <i>Suppliers</i> .....	47
Tabel 4.8 Struktur Tabel <i>Categories</i> .....	47
Tabel 4.9 Struktur Tabel <i>Product</i> .....	47
Tabel 4.10 Struktur Tabel <i>Stocks</i> .....	48
Tabel 4.11 Struktur Tabel Penjualan .....	48
Tabel 4.12 Struktur Tabel Penjualan_ <i>products</i> .....	49
Tabel 4.13 Struktur Tabel Pembelian .....	49
Tabel 4.14 Struktur Tabel Pembelian_ <i>product</i> .....	50
Tabel 4.15 Hasil Pengujian Halaman <i>Login</i> – Admin.....	109
Tabel 4.16 Hasil Pengujian pada Navigasi Menu – Admin .....	109
Tabel 4.17 Pengujian pada Halaman <i>Users</i> – Admin .....	111
Tabel 4.18 Pengujian <i>Form</i> Tambah pada Halaman <i>Users</i> – Admin .....	112
Tabel 4.19 Pengujian <i>Form</i> <i>Edit</i> pada Halaman <i>Users</i> – Admin .....	112
Tabel 4.20 Pengujian pada Halaman <i>Customers</i> – Admin .....	113
Tabel 4.21 Pengujian <i>Form</i> Tambah pada Halaman <i>Customers</i> – Admin .....	114
Tabel 4.22 Pengujian <i>Form</i> <i>Edit</i> pada Halaman <i>Customers</i> – Admin .....	115
Tabel 4.23 Pengujian pada Halaman <i>Suppliers</i> – Admin .....	115
Tabel 4.24 Pengujian <i>Form</i> Tambah pada Halaman <i>Suppliers</i> – Admin .....	116
Tabel 4.25 Pengujian <i>Form</i> <i>Edit</i> pada Halaman <i>Customers</i> – Admin .....	117
Tabel 4.26 Pengujian pada Halaman <i>Categories</i> – Admin .....	118
Tabel 4.27 Pengujian <i>Form</i> Tambah pada Halaman <i>Categories</i> – Admin .....	118
Tabel 4.28 Pengujian <i>Form</i> <i>Edit</i> pada Halaman <i>Categories</i> – Admin .....	119
Tabel 4.29 Pengujian pada Halaman <i>Products</i> – Admin .....	120

Tabel 4.30 Pengujian <i>Form Tambah</i> pada Halaman <i>Products – Admin</i> .....	121
Tabel 4.31 Pengujian <i>Form Edit</i> pada Halaman <i>Products – Admin</i> .....	121
Tabel 4.32 Pengujian pada Halaman <i>Stocks – Admin</i> .....	122
Tabel 4.33 Pengujian <i>Form Edit</i> pada Halaman <i>Stocks – Admin</i> .....	123
Tabel 4.34 Pengujian pada Halaman Penjualan – Admin.....	123
Tabel 4.35 Pengujian pada Halaman Tambah Transaksi Jual – Admin .....	124
Tabel 4.36 Pengujian pada Halaman <i>Edit Transaksi Jual – Admin</i> .....	125
Tabel 4.37 Pengujian pada Halaman Pembelian – Admin.....	125
Tabel 4.38 Pengujian pada Halaman Tambah Transaksi Beli – Admin.....	126
Tabel 4.39 Pengujian pada Halaman <i>Edit Transaksi Beli – Admin</i> .....	127
Tabel 4.40 Pengujian pada Halaman Laporan Penjualan – Admin .....	128
Tabel 4.41 Pengujian pada Halaman Laporan Pembelian – Admin .....	129
Tabel 4.42 Hasil Pengujian Halaman <i>Login – Kasir</i> .....	131
Tabel 4.43 Hasil Pengujian pada Navigasi Menu – Kasir .....	132
Tabel 4.44 Pengujian pada Halaman <i>Customers – Kasir</i> .....	132
Tabel 4.45 Pengujian <i>Form Tambah</i> pada Halaman <i>Customers – Kasir</i> .....	133
Tabel 4.46 Pengujian <i>Form Edit</i> pada Halaman <i>Customers – Kasir</i> .....	134
Tabel 4.47 Pengujian pada Halaman <i>Categories – Kasir</i> .....	135
Tabel 4.48 Pengujian <i>Form Tambah</i> pada Halaman <i>Categories – Kasir</i> .....	136
Tabel 4.49 Pengujian <i>Form Edit</i> pada Halaman <i>Categories – Kasir</i> .....	136
Tabel 4.50 Pengujian pada Halaman <i>Products – Kasir</i> .....	137
Tabel 4.51 Pengujian <i>Form Tambah</i> pada Halaman <i>Products – Kasir</i> .....	138
Tabel 4.52 Pengujian <i>Form Edit</i> pada Halaman <i>Products – Kasir</i> .....	139
Tabel 4.53 Pengujian pada Halaman Penjualan – Kasir .....	139
Tabel 4.54 Pengujian pada Halaman Tambah Transaksi Jual – Kasir .....	140
Tabel 4.55 Pengujian pada Halaman <i>Edit Transaksi Jual – Kasir</i> .....	141
Tabel 4.56 Hasil Pengujian Halaman <i>Login – Gudang</i> .....	142
Tabel 4.57 Hasil Pengujian pada Navigasi Menu – Gudang .....	142
Tabel 4.58 Pengujian pada Halaman <i>Suppliers – Gudang</i> .....	143
Tabel 4.59 Pengujian <i>Form Tambah</i> pada Halaman <i>Suppliers – Gudang</i> .....	144
Tabel 4.60 Pengujian <i>Form Edit</i> pada Halaman <i>Customers – Gudang</i> .....	145
Tabel 4.61 Pengujian pada Halaman <i>Categories – Gudang</i> .....	146
Tabel 4.62 Pengujian <i>Form Tambah</i> pada Halaman <i>Categories – Gudang</i> .....	146
Tabel 4.63 Pengujian <i>Form Edit</i> pada Halaman <i>Categories – Gudang</i> .....	147
Tabel 4.64 Pengujian pada Halaman <i>Products – Gudang</i> .....	148

Tabel 4.65 Pengujian <i>Form Tambah</i> pada Halaman <i>Products – Gudang</i> .....	149
Tabel 4.66 Pengujian <i>Form Edit</i> pada Halaman <i>Products – Gudang</i> .....	149
Tabel 4.68 Pengujian <i>Form Edit</i> pada Halaman <i>Stocks – Gudang</i> .....	151
Tabel 4.69 Pengujian pada Halaman Pembelian – Gudang.....	151
Tabel 4.70 Pengujian pada Halaman Tambah Transaksi Beli – Gudang.....	152
Tabel 4.71 Pengujian pada Halaman <i>Edit Transaksi Beli – Gudang</i> .....	153
Tabel 4.72 Pengujian pada Halaman Laporan Penjualan – Gudang .....	154
Tabel 4.73 Pengujian pada Halaman Laporan Pembelian – Gudang .....	155



INSTITUT TEKNOLOGI DAN BISNIS  
**STIKOM BALI**



## DAFTAR GAMBAR

Gambar 2.1 Gudang .....	9
Gambar 2.2 Akun <i>instagram</i> pertama .....	10
Gambar 2.3 Akun <i>instagram</i> kedua.....	10
Gambar 2.4 Grafik Penjualan .....	11
Gambar 2.5 MariaDB .....	18
Gambar 2.6 Laravel .....	20
Gambar 2.7 Bootstrap.....	21
Gambar 2.8 Model <i>Waterfall</i> .....	23
Gambar 4.1 Diagram Konteks .....	34
Gambar 4.2 Data <i>Flow</i> Diagram (DFD) Level 0 .....	35
Gambar 4.3 Data <i>Flow</i> Diagram (DFD) Level 1 Proses 1 <i>Login</i> .....	36
Gambar 4.4 Data <i>Flow</i> Diagram (DFD) Level 1 Proses 2 <i>Users</i> .....	37
Gambar 4.5 Data <i>Flow</i> Diagram (DFD) Level 1 Proses 3 <i>Customers</i> .....	37
Gambar 4.6 Data <i>Flow</i> Diagram (DFD) Level 1 Proses 4 <i>Suppliers</i> .....	38
Gambar 4.7 Data <i>Flow</i> Diagram (DFD) Level 1 Proses 5 <i>Categories</i> .....	39
Gambar 4.8 Data <i>Flow</i> Diagram (DFD) Level 1 Proses 6 <i>Product</i> .....	40
Gambar 4.9 Data <i>Flow</i> Diagram (DFD) Level 1 Proses 7 <i>Stock</i> .....	41
Gambar 4.10 Data <i>Flow</i> Diagram (DFD) Level 1 Proses 8 Transaksi Jual .....	42
Gambar 4.11 Data <i>Flow</i> Diagram (DFD) Level 1 Proses 9 Transaksi Beli.....	43
Gambar 4.12 Data <i>Flow</i> Diagram (DFD) Level 1 Proses 10 Laporan .....	43
Gambar 4.13 <i>Entity Relationship</i> Diagram .....	44
Gambar 4.14 Basis Data Konseptual .....	45
Gambar 4.15 Antarmuka <i>Login</i> – Admin .....	51
Gambar 4.16 Antarmuka <i>Dashboard</i> – Admin .....	51
Gambar 4.17 Antarmuka Data <i>Users</i> – Admin.....	52
Gambar 4.18 Antarmuka <i>Form Users</i> – Admin .....	52
Gambar 4.19 Antarmuka <i>detail users</i> – Admin .....	53
Gambar 4.20 Antarmuka Data <i>Customers</i> – Admin.....	53
Gambar 4.21 Antarmuka <i>Form Customers</i> – Admin.....	54
Gambar 4.22 Antarmuka <i>Detail Customers</i> – Admin .....	54
Gambar 4.23 Antarmuka Data <i>Suppliers</i> – Admin .....	55
Gambar 4.24 Antarmuka <i>Form Suppliers</i> – Admin .....	55

Gambar 4.25 Antarmuka Detail <i>Suppliers</i> – Admin .....	56
Gambar 4.26 Antarmuka Data <i>Categories</i> – Admin .....	56
Gambar 4.27 Antarmuka <i>Form Categories</i> – Admin .....	57
Gambar 4.28 Antarmuka Data <i>Products</i> – Admin .....	57
Gambar 4.29 Antarmuka <i>Form Products</i> – Admin.....	58
Gambar 4.30 Antarmuka Detail <i>Products</i> – Admin.....	59
Gambar 4.31 Antarmuka Data <i>Stock</i> – Admin.....	59
Gambar 4.32 Antarmuka <i>Form Stock</i> – Admin.....	60
Gambar 4.33 Antarmuka Data Penjualan – Admin .....	60
Gambar 4.34 Antarmuka Transaksi Jual – Admin .....	61
Gambar 4.35 Antarmuka Data Pembelian – Admin .....	61
Gambar 4.36 Antarmuka Transaksi Beli – Admin.....	62
Gambar 4.37 Antarmuka <i>Invoice</i> Penjualan – Admin.....	62
Gambar 4.38 Antarmuka <i>Invoice</i> Pembelian – Admin.....	63
Gambar 4.39 Antarmuka Laporan Penjualan – Admin.....	63
Gambar 4.40 Antarmuka Laporan Pembelian – Admin.....	64
Gambar 4.41 Antarmuka <i>Login</i> – Kasir.....	65
Gambar 4.42 Antarmuka <i>Dashboard</i> – Kasir.....	65
Gambar 4.43 Antarmuka <i>List Customer</i> – Kasir .....	66
Gambar 4.44 Antarmuka <i>Form Customer</i> – Kasir .....	67
Gambar 4.45 Antarmuka Detail <i>Customer</i> – Kasir .....	67
Gambar 4.46 Antarmuka <i>List Kategori</i> – Kasir .....	68
Gambar 4.47 Antarmuka <i>Form Kategori</i> – Kasir .....	68
Gambar 4.48 Antarmuka <i>List Product</i> – Kasir .....	69
Gambar 4.49 Antarmuka <i>Form Product</i> – Kasir .....	70
Gambar 4.50 Antarmuka Detail <i>Product</i> – Kasir.....	70
Gambar 4.51 Antarmuka <i>List Penjualan</i> – Kasir.....	71
Gambar 4.52 Antarmuka <i>Form Penjualan</i> – Kasir.....	71
Gambar 4.53 Antarmuka <i>Invoice</i> Penjualan – Kasir.....	72
Gambar 4.54 Antarmuka <i>Login</i> – Gudang .....	73
Gambar 4.55 Antarmuka <i>Dashboard</i> Kasir – Gudang.....	73
Gambar 4.56 Antarmuka <i>List Supplier</i> – Gudang.....	74
Gambar 4.57 Antarmuka <i>Form Supplier</i> – Gudang.....	74
Gambar 4.58 Antarmuka Detail <i>Supplier</i> – Gudang .....	75
Gambar 4.59 Antarmuka <i>List Kategori</i> – Gudang.....	75

Gambar 4.60 Antarmuka <i>Form Kategori</i> – Gudang .....	76
Gambar 4.61 Antarmuka <i>List Product</i> – Gudang .....	76
Gambar 4.62 Antarmuka <i>Form Product</i> – Gudang .....	77
Gambar 4.63 Antarmuka <i>Detail Product</i> – Gudang .....	77
Gambar 4.64 Antarmuka <i>List Stock</i> – Gudang.....	78
Gambar 4.65 Antarmuka <i>Form Stock</i> – Gudang.....	78
Gambar 4.66 Antarmuka <i>List Pembelian</i> – Gudang .....	79
Gambar 4.67 Antarmuka <i>Form Pembelian</i> – Gudang.....	79
Gambar 4.68 Tampilan Halaman <i>Login</i> – Admin .....	80
Gambar 4.69 Tampilan Halaman <i>Dashboard</i> – Admin .....	81
Gambar 4.70 Tampilan Halaman <i>List Users</i> – Admin.....	81
Gambar 4.71 Tampilan <i>Form User</i> – Admin .....	82
Gambar 4.72 Tampilan <i>Detail User</i> – Admin.....	82
Gambar 4.73 Tampilan Halaman <i>List Customers</i> – Admin.....	83
Gambar 4.74 Tampilan <i>Form Customer</i> – Admin .....	83
Gambar 4.75 Tampilan <i>Detail Customer</i> – Admin .....	84
Gambar 4.76 Tampilan Halaman <i>List Suppliers</i> – Admin .....	84
Gambar 4.77 Tampilan <i>Form Supplier</i> – Admin.....	85
Gambar 4.78 Tampilan <i>Detail Supplier</i> – Admin .....	85
Gambar 4.79 Tampilan Halaman <i>List Categories</i> – Admin.....	86
Gambar 4.80 Tampilan <i>Form Category</i> – Admin .....	86
Gambar 4.81 Tampilan Halaman <i>List Products</i> – Admin.....	87
Gambar 4.82 Tampilan <i>Form Product</i> – Admin.....	87
Gambar 4.83 Tampilan <i>Detail Product</i> – Admin.....	88
Gambar 4.84 Tampilan Halaman <i>List Stocks</i> – Admin .....	88
Gambar 4.85 Tampilan <i>Form Stock</i> – Admin .....	89
Gambar 4.86 Tampilan Halaman <i>List Penjualan</i> – Admin .....	89
Gambar 4.87 Tampilan <i>Form Penjualan</i> – Admin .....	90
Gambar 4.88 Tampilan Halaman <i>Invoice Penjualan</i> – Admin .....	90
Gambar 4.89 Tampilan Halaman <i>List Pembelian</i> – Admin .....	91
Gambar 4.90 Tampilan Halaman <i>Form Pembelian</i> – Admin .....	91
Gambar 4.91 Tampilan Halaman <i>Invoice Pembelian</i> – Admin .....	92
Gambar 4.92 Tampilan Halaman Laporan Penjualan – Admin .....	93
Gambar 4.93 Tampilan Halaman Laporan Pembelian – Admin .....	94
Gambar 4.94 Tampilan Halaman <i>Login</i> – Kasir .....	95

Gambar 4.95 Tampilan Halaman <i>Dashboard</i> – Kasir.....	95
Gambar 4.96 Tampilan Halaman <i>List Customers</i> – Kasir .....	96
Gambar 4.97 Tampilan <i>Form Customer</i> – Kasir.....	96
Gambar 4.98 Tampilan Detail <i>Customer</i> – Kasir .....	97
Gambar 4.99 Tampilan Halaman <i>List Categories</i> – Kasir .....	97
Gambar 4.100 Tampilan <i>Form Category</i> – Kasir.....	98
Gambar 4.101 Tampilan Halaman <i>List Products</i> – Kasir .....	98
Gambar 4.102 Tampilan <i>Form Product</i> – Kasir .....	99
Gambar 4.103 Tampilan Detail <i>Product</i> – Kasir .....	99
Gambar 4.104 Tampilan Halaman <i>List Penjualan</i> – Kasir .....	100
Gambar 4.105 Tampilan <i>Form Penjualan</i> – Kasir .....	100
Gambar 4.106 Tampilan Halaman <i>Invoice Penjualan</i> – Kasir.....	101
Gambar 4.107 Tampilan Halaman <i>Login</i> – Gudang.....	101
Gambar 4.108 Tampilan Halaman <i>Dashboard</i> – Gudang.....	102
Gambar 4.109 Tampilan Halaman <i>List Suppliers</i> – Gudang.....	102
Gambar 4.110 Tampilan <i>Form Supplier</i> – Gudang .....	103
Gambar 4.111 Tampilan Detail <i>Supplier</i> – Gudang.....	103
Gambar 4.112 Tampilan Halaman <i>List Categories</i> – Gudang .....	104
Gambar 4.113 Tampilan <i>Form Category</i> – Gudang .....	104
Gambar 4.114 Tampilan Halaman <i>List Products</i> – Gudang.....	105
Gambar 4.115 Tampilan <i>Form Product</i> – Gudang .....	105
Gambar 4.116 Tampilan Detail <i>Product</i> – Gudang .....	106
Gambar 4.117 Tampilan Halaman <i>List Stocks</i> – Gudang .....	106
Gambar 4.118 Tampilan <i>Form Stock</i> – Gudang.....	107
Gambar 4.119 Tampilan Halaman <i>List Pembelian</i> – Gudang.....	107
Gambar 4.120 Tampilan Halaman <i>Form Pembelian</i> – Gudang .....	108
Gambar 4.121 Tampilan Halaman <i>Invoice Pembelian</i> – Gudang .....	108

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Balinese *Classic* merupakan sebuah usaha yang bergerak pada penjualan kerajinan bali khususnya tas rotan. Produk dipasarkan ke berbagai negara seperti Malaysia, Filipina, Singapore dan masih banyak lagi melalui *supplier* masing-masing negara. Perusahaan yang bertempat di Jalan Kebo Iwa Perumahan Swamandala XIII No.3 Denpasar, Bali. Perusahaan ini baru merintis usahanya sejak Juli 2018 lalu. Balinese *Classic* melakukan pembelian tas dari beberapa *supplier* di pasar tradisional Bali kemudian barang akan disimpan di gudang dan akan dipasarkan kembali ke berbagai belahan dunia.

Selama ini perusahaan dalam melakukan kegiatan pengolahan data laporan dilakukan secara manual dengan menggunakan buku besar. Laporan tersebut berisi tentang *informasi* pembelian, penjualan, stok barang, jumlah barang yang dikirim ke masing-masing *supplier*. Laporan tersebut berasal dari hasil pencatatan dari nota-nota pembelian maupun penjualan setiap minggunya. Selain itu untuk menghitung laba rugi juga masih menggunakan alat manual seperti kalkulator. Dari proses tersebut, maka lambat laun akan menimbulkan kendala untuk operasional pada Balinese *Classic*.

Sebelumnya sudah dilakukan penelitian terdahulu yang memiliki keterkaitan dengan sistem *informasi* penjualan. Penelitian pertama diteliti oleh Yazid Maulana dengan judul "Sistem Informasi Penjualan Kayu Ulin Pada UD Karya Perdana Berbasis Web Menggunakan Framework Codeigniter" [1]. Dalam penelitian tersebut dijabarkan bagaimana cara bertransaksi menggunakan sarana aplikasi berbasis *web* yang bertujuan memudahkan penjualan kayu ulin pada UD Karya Perdana. Penelitian kedua dengan judul "Sistem Informasi Eksekutif Penjualan Jajanan Sagu pada Bagia Suli menggunakan Framework Laravel" yang diteliti oleh I Kadek Adi Adnyana [2]. Pada penelitian tersebut dihasilkan sebuah aplikasi eksekutif penjualan yang memiliki manfaat memudahkan penjualan jajanan sagu juga mempermudah dalam pembuatan laporan penjualan. Keterkaitan yang ada pada penelitian tersebut dengan penelitian yang akan dibuat ialah sama-sama sistem *informasi* eksekutif penjualan dan menggunakan *Framework* laravel.

Berdasarkan uraian sebelumnya dan referensi dari penelitian terdahulu maka pada penelitian ini penulis akan membuat Sistem Informasi Eksekutif penjualan kerajinan tas rotan pada Balinese Classic menggunakan *Framework* Laravel. Sistem Informasi Eksekutif (SIE) adalah salah satu jenis manajemen sistem informasi untuk memudahkan dan mendukung keterangan dan pembuatan keputusan yang dibutuhkan eksekutif senior dengan menyediakan kemudahan akses terhadap informasi baik dari dalam maupun luar. Sistem ini juga didukung dengan *Framework* Laravel yaitu suatu pengembangan website berbasis *Model View Controller* (MVC) yang ditulis dalam *PHP* yang dirancang untuk meningkatkan kualitas perangkat lunak dengan mengurangi biaya pengembangan awal dan biaya pemeliharaan, dan untuk meningkatkan pengalaman bekerja dengan aplikasi dengan menyediakan kode yang ekspresif, jelas dan menghemat waktu. Dengan adanya sistem informasi eksekutif ini dapat membantu mengelola proses transaksi seperti mencatat, mengolah serta menampilkan dalam bentuk grafik data transaksi. Keunggulan yang ditawarkan sistem ini adalah memudahkan kinerja perusahaan dalam proses perhitungan, pencatatan data-data perusahaan, serta arsip data perusahaan. Dalam hal ini sistem akan membantu pihak perusahaan dalam perencanaan stok barang dan perencanaan penjualan.

### **1.2 Rumusan Masalah**

Adapun rumusan masalah pada penelitian ini adalah bagaimana hasil rancang bangun Sistem Informasi Eksekutif penjualan kerajinan tas rotan pada Balinese Classic menggunakan *Framework* Laravel?

### **1.3 Tujuan Penelitian**

Adapun tujuan dari penelitian ini adalah untuk merancang bangun Sistem Informasi Eksekutif penjualan kerajinan tas rotan pada Balinese Classic menggunakan *Framework* Laravel.

### **1.4 Manfaat Penelitian**

- Adapun Manfaat yang didapat dari penelitian ini adalah sebagai berikut:
1. Memudahkan pengecekan dan pencarian data *customer*, stok barang dan transaksi yang masuk.
  2. Dapat melakukan *review* laporan penjualan per hari, per bulan, dan per tahun.
  3. Efisiensi kerja, waktu, dan tenaga.

4. Menyajikan laporan penjualan dengan cepat dan tepat waktu dalam bentuk grafik atau diagram.
5. Mempermudah dalam memantau perkembangan perusahaan.
6. Menghindari kesalahan yang diakibatkan menggunakan sistem manual.

### **1.5 Ruang lingkup Penelitian**

Adapun ruang lingkup pada penelitian ini adalah sebagai berikut:

1. *User* dari sistem *informasi eksekutif penjualan* ini adalah:
  - a. *Admin* adalah pengguna yang menjadi pengelola sistem *informasi* yang mendapat hak akses penuh terhadap sistem ini.
  - b. Kasir adalah pengguna yang memiliki tugas melakukan *input* penjualan.
  - c. Gudang adalah pengguna yang memiliki tugas melakukan *input* pembelian.
2. Sebelum masuk ke sistem, pengguna diharuskan melakukan *login* terlebih dahulu.
3. Untuk *login Top Level Management* dan Operator, Administrator harus mendaftarkan dulu pada halaman khusus administrator.
4. Data yang digunakan pada sistem ini adalah data *user*, data pembelian, data barang, data kategori, data pembeli, data *supplier* serta data penjualan.
5. Sistem memiliki fitur *dashboard* di mana berisi menu – menu tersendiri sesuai dengan siapa yang *login*, apakah itu *Top Level Management*, *Admin* atau Operator.
6. Admin dapat mengelola keseluruhan yang ada pada sistem.
7. Operator hanya dapat mengelola data transaksi, pembelian, barang dan *supplier*.
8. *Top Level Management* mendapatkan *informasi* berupa laporan seperti:
  - a. Laporan data penjualan baik itu per-hari, mingguan maupun tahunan.
  - b. Laporan produk yang paling banyak dijual.
  - c. Laporan pembelian barang.
  - d. Data barang.
9. Sistem ini dibangun dalam bentuk *web* menggunakan bahasa pemrograman *PHP* dan menggunakan *Framework Laravel* serta *Ajax*. *Database* yang digunakan dalam sistem ini menggunakan *MySQL* dengan *XAMPP* sebagai *software* untuk pengolahannya.

10. Perancangan sistem ini menggunakan *Data Flow Diagram (DFD)* dan untuk perancangan basis data digunakan *Entity Relationship Diagram (ERD)*.
11. *Output* yang dihasilkan adalah Struk transaksi, Laporan penjualan dan Laporan pembelian.
12. Pengujian sistem yang dilakukan menggunakan metode pengujian *Blackbox testing*.

### **1.6 Sistematika Penulisan**

Agar dalam penulisan laporan ini lebih tersusun rapi, maka sistematika penulisan laporan ini dibagi menjadi beberapa bab dengan masing – masing sub babnya yang saling berkaitan. Berikut adalah penulisan masing – masing bab yang terdapat pada laporan ini:

#### **BAB I PENDAHULUAN**

Bab I menjelaskan tentang *informasi umum* yaitu latar belakang penelitian, perumusan masalah, tujuan penelitian, manfaat penelitian, dan ruang lingkup penelitian.

#### **BAB II TINJAUAN PUSTAKA**

Bab II berisi tentang kajian-kajian teori yang menunjang penelitian selama penelitian berlangsung dengan topik yang akan digunakan yaitu sistem *informasi*, basis data, sistem *informasi eksekutif*, *PHP*, *Laravel*, *Ajax*, *Bootstrap*, *MariaDB*, *Flowchart*, *DFD*, *ERD*, Basis data konseptual.

#### **BAB III METODE PENELITIAN**

Bab III ini menjelaskan tentang lokasi dan waktu penelitian, alat dan bahan penelitian, metode pengumpulan data, dan metode perekayasaan pada sistem ini.

#### **BAB IV HASIL DAN PEMBAHASAN**

Bab IV menjelaskan tentang analisa dan perancangan sistem yang dimulai dari perancangan basis data, *data Flow diagram*, *entity relationship diagram*, basis

data konseptual, struktur *table*, perancangan antarmuka, implementasi sistem serta pengujian sistem.

## **BAB V PENUTUP**

Bab V sekaligus bab terakhir menjelaskan kesimpulan dan saran dari keseluruhan penelitian yang telah dilakukan, di mana kesimpulan dan saran ini dibuat sebagai tolak ukur apabila akan dilakukan pengembangan aplikasi selanjutnya untuk sistem ini.



INSTITUT TEKNOLOGI DAN BISNIS  
**STIKOM BALI**



## BAB II

### TINJAUAN PUSTAKA

#### 2.1 *State of The Art*

Penelitian tentang sistem *informasi penjualan* telah dilakukan oleh para peneliti sebelumnya. Penelitian-penelitian yang terkait dengan penelitian yang akan dilakukan oleh penulis terdapat pada Tabel 2.1.

Tabel 2.1 *State of The Art*

No	Nama	Judul	Deskripsi
1	Yazid Maulana [1]	Sistem Informasi Penjualan Kayu Ulin Pada UD Karya Perdana Berbasis Web Menggunakan Framework Codeigniter (2018).	<ul style="list-style-type: none"><li>• Sistem informasi ini dibangun pada <i>platform website</i> dan bisa diakses secara <i>online</i></li><li>• Dari segi pembangunan sistem informasi ini menggunakan Bahasa pemrograman <i>web</i> yaitu <i>PHP</i> dengan menggunakan <i>Framework</i> Codeigniter dan Bootstrap.</li><li>• Sistem informasi ini dapat melakukan <i>custom order</i> sesuai dengan permintaan <i>customer</i>.</li></ul>
2	I Kadek Adi Adnyana [2]	Sistem Informasi Eksekutif Penjualan Jajanan Sagu Pada Bagia Suli Menggunakan Framework Laravel (2018).	<ul style="list-style-type: none"><li>• Sebuah sistem informasi eksekutif yang dapat membantu dalam kegiatan penjualan.</li><li>• Sebuah sistem informasi eksekutif yang dapat menghemat waktu dalam proses pengumpulan informasi laporan</li></ul>

No	Nama	Judul	Deskripsi
			<p>penjualan yang dilakukan oleh pegawai administrasi.</p> <ul style="list-style-type: none"> <li>• <i>Output</i> pada sistem ini berupa laporan penjualan per hari, per bulan, per tahun.</li> </ul>
3	Ketut Oka Edi Artawan [3]	Sistem Informasi Penyewaan Busana Pada Denata Kebaya Menggunakan Framework Laravel (2019).	<ul style="list-style-type: none"> <li>• Dalam pengembangan sistem ini menggunakan metode <i>waterfall</i>.</li> <li>• Sistem ini dibangun atas <i>platform website</i> sehingga dalam operasional sistem ini dapat dijalankan dengan menggunakan internet dan browser.</li> <li>• Hasil dari penelitian telah menghasilkan sistem yang dapat mengelola data dan informasi penyewaan busana Denata Kebaya.</li> </ul>
4	I Putu Kawayana 2019 [4]	Sistem Informasi Eksekutif Pengadaan Barang Pada Saturday Cloth Menggunakan Framework Laravel (2019).	<ul style="list-style-type: none"> <li>• Sistem ini menggunakan Bahasa pemrograman <i>PHP</i> dengan <i>Framework</i> Laravel dan basis data Mysql.</li> <li>• Dari segi perancangan sistem informasi penyewaan ini menggunakan <i>DFD</i> dan <i>ERD</i>.</li> <li>• Sistem ini memiliki beberapa fitur yaitu</li> </ul>

No	Nama	Judul	Deskripsi
			pengelolaan barang, laporan berbentuk grafik.

Berdasarkan refensi penelitian terdahulu yang terdapat pada *Table 2.1*, dapat disimpulkan bahwa perekayasaan yang akan dibuat memiliki kemiripan dari segi metode pengembangan sistem yaitu dengan menggunakan metode *waterfall*, dari segi perancangan sistem menggunakan *ERD* dan *DFD*, dan dari segi *platform* yang digunakan untuk mengoperasikan sistem yaitu dengan menggunakan *platform website*. Berdasarkan beberapa penelitian tersebut dapat digunakan sebagai acuan dalam mengembangkan sistem pada perekayasaan ini. Selain melihat dari kemiripan adapun pengembangan dari segi teknologi yang akan digunakan pada perekayasaan sistem yaitu dibangun dengan menggunakan *Framework Laravel*.

## 2.2 Balinese Classic

Balinese Classic merupakan sebuah usaha yang bergerak pada penjualan kerajinan bali khususnya tas rotan. Produk dipasarkan ke berbagai negara seperti Malaysia, Filipina, Singapore dan masih banyak lagi melalui *supplier* masing-masing negara. Perusahaan yang bertempat di Jalan Kebo Iwa Perumahan Swamandala XIII No.3 Denpasar, Bali. Perusahaan ini baru merintis usahanya sejak Juli 2018 lalu. Balinese Classic melakukan pembelian tas dari beberapa *supplier* di pasar tradisional Bali kemudian barang akan disimpan di gudang dan akan dipasarkan kembali ke berbagai belahan dunia. Saat ini dalam menyimpan barang, untuk sementara balinese classic menggunakan kamar sebagai media penyimpanan dan juga untuk mencatat stock masih dilakukan secara manual melalui buku besar yang bisa dilihat pada Gambar 2.1



Gambar 2.1 Gudang

## 1. Sejarah Usaha

Awal mulanya usaha penjualan tas ini didirikan secara tidak sengaja oleh luthvi dan yogi karena sedang ada *trend* tas rotan yang sedang maraknya dicari-cari oleh wisatawan asing. Langkah awal yang dilakukan luthvi dan yogi yaitu mencari pengrajin dari tas rotan ini, setelah berhasil mendapatkan beberapa pengrajin dan membeli beberapa *stock*, Kami mencoba memasarkannya melalui *instagram* dengan membuat 2 akun *instagram*, berikut merupakan halaman *instagram* dari usaha kecil ini yang bisa dilihat pada Gambar 2.1 dan 2.2



Gambar 2.2 Akun *instagram* pertama



Gambar 2.3 Akun *instagram* kedua

Akun di atas dikelola oleh luthvi dan yogi sendiri yang bertugas juga sebagai admin.

## 2. Visi dan Misi Balinese Classic

Adapun visi dan misi dari Balinese Classic meliputi :

### Visi

- Selalu menjadi perusahaan tas unik yang memiliki konsep kuat dan memiliki ciri khas tersendiri dibandingkan dengan tas lain.

- b. Memiliki nilai jual tinggi dengan adanya pembaruan-pembaruan yang diterapkan pada produk sesuai tren yang berjalan.
- c. Mampu menciptakan produk *original* agar perusahaan dapat menciptakan rasa bangga atas kepuasan dari konsumen Balinese Classic.

#### **Misi**

- a. Membuat produk yang berkualitas dengan selalu memantau kualitas di setiap produksinya.
- b. Membuat desain baru yang belum pernah dibuat orang lain.
- c. Membuat ciri khas tersendiri yang menjadi kekuatan perusahaan Balinese Classic.
- d. *Research* segala jenis *mode* yang berkembang dan dikembangkan ulang dengan konsep Balinese Classic.

### 3. Penjualan

Saat ini Balinese Classic dalam melakukan pengecekan laporan penjualan tidak dalam bentuk grafik melainkan masih dalam bentuk tabel yang dicatat pada buku besar, dari permasalahan tersebut maka penulis membuat contoh grafik penjualan yang bisa dilihat pada Gambar 2.4 berikut:



Gambar 2.4 Grafik Penjualan

Berikut penjelasan dari grafik penjualan pada Gambar 2.4 :

- a. Pada bulan januari, penjualan tas sebanyak 300 $pcs$  yang meliputi tas rotan sebanyak 200 $pcs$  dan tas ate sebanyak 100 $pcs$ .
- b. Pada bulan februari, penjualan tas sebanyak 550 $pcs$  yang meliputi tas rotan sebanyak 300 $pcs$  dan tas ate sebanyak 250 $pcs$ .

- c. Pada bulan januari, penjualan tas sebanyak 650 $pcs$  yang meliputi tas rotan sebanyak 450 $pcs$  dan tas ate sebanyak 200 $pcs$ .
- d. Pada bulan januari, penjualan tas sebanyak 650 $pcs$  yang meliputi tas rotan sebanyak 500 $pcs$  dan tas ate sebanyak 150 $pcs$ .
- e. Pada bulan januari, penjualan tas sebanyak 320 $pcs$  yang meliputi tas rotan sebanyak 200 $pcs$  dan tas ate sebanyak 120 $pcs$ .

Dengan adanya data penjualan berbentuk grafik seperti Gambar 2.4, dapat memudahkan dalam memantau perkembangan perusahaan, kapan penjualan menurun/meningkat bisa dilihat dengan jelas.

### 2.3 Sistem Informasi Eksekutif (SIE)

Sistem informasi eksekutif didefinisikan sebagai sebuah sistem terkomputerisasi yang menyediakan *informasi* internal dan eksternal bagi para eksekutif yang relevan terhadap faktor penentu keberhasilan dengan akses yang sangat mudah. Sistem informasi eksekutif dirancang untuk membantu eksekutif mencari *informasi* yang diperlukan pada saat mereka membutuhkannya dan dalam bentuk apapun yang paling bermanfaat [5].

### 2.4 Tujuan Sistem Informasi Eksekutif

Tujuan dari perancangan sistem aplikasi eksekutif adalah untuk membantu para eksekutif dalam mempelajari tentang sebuah organisasi, bagaimana proses-proses bekerja, interaksi antara lingkungan luar dan lingkungan dalam satu organisasi untuk mendukung dalam membuat laporan berdasarkan data yang akurat. Secara global sistem informasi eksekutif merupakan sistem yang digunakan oleh *top level management* dalam mengetahui perkembangan yang terjadi pada perusahaan.

Sistem Informasi Eksekutif (SIE) dikembangkan untuk bermacam-macam alasan. Kajian-kajian memperlihatkan bahwa keuntungan-keuntungan yang paling umum dari SIE adalah perbaikan dalam kualitas dan kuantitas yang ada bagi para eksekutif.

1. Lebih banyak *informasi* yang tepat waktu.
2. Akses ke data operasional lebih besar.
3. Akses ke *database* perusahaan lebih besar.
4. Lebih banyak *informasi* yang bersaing.
5. Terdapat *informasi* baru atau tambahan.

6. Mengurangi biaya kertas.
7. Informasi tentang lingkungan luar lebih banyak [6].

## 2.5 **Website**

*WEB (Word Wide Web)* atau situs dapat diartikan sebagai kumpulan halaman-halaman yang digunakan untuk menampilkkan informasi, teks gambar diam atau bergerak, animasi suara dana atau gabungan dari semua itu. Baik yang bersifat statis maupun dinamis yang membentuk suatu rangkaian bangunan yang saling berkaitan di mana masing-masing dihubungkan dengan jaringan-jaringan halaman (*hyperlink*). Untuk menyediakan keberadaan sebuah website. maka harus disediakan unsur-unsur penunjang seperti:

1. **Nama Domain**

Nama *domain* atau *domain name* atau URL (*Uniform Resource Located*) adalah alamat unik di dunia internet yang digunakan untuk mengidentifikasi sebuah *website* atau digunakan untuk menemukan sebuah *website* pada dunia internet.

2. **Tempat Website (Web Hosting)**

*Web Hosting* dapat diartikan sebagai ruangan yang terdapat dalam *hardisk* tempat penyimpanan berbagai data, *file*, gambar. dan sebagainya yang akan ditampilkan di *website*.

3. **Bahasa Program (Script Program)**

Bahasa program merupakan bahasa yang digunakan untuk menerjemahkan setiap perintah dalam website pada saat diakses.

4. **Desain Website**

Setelah melakukan pembelian nama *domain* dan *web hosting* serta penguasaan program, maka harus memiliki desain *website*. Unsur *website* yang penting adalah desain, karena desain *website* sangat menentukan kualitas dan keindahan dari suatu *website*.

5. **Publikasi website**

Keberadaan sebuah *website* tidak ada gunanya dibangun apabila tidak dikunjungi atau dikenal oleh pengguna internet. Maka untuk mengenalkan situs *website* kepada masyarakat memerlukan adanya publikasi atau promosi.

6. **Pemeliharaan website**

Untuk mendukung kelanjutan dari situs *website* diperlukan pemeliharaan setiap waktu sesuai dengan keinginan seperti penambahan konten *informasi*,

berita, *link*, artikel, gambar maupun informasi terbaru lainnya. Tanpa pemeliharaan yang baik situs akan terkesan membosankan dan monoton, sehingga akan mulai ditinggalkan oleh pengunjung internet [7].

## 2.6 Data Flow Diagram (DFD)

*Data Flow Diagram (DFD)* adalah suatu *network* yang menggambarkan suatu Sistem komputer, manual, atau gabungan dari keduanya yang penggambarannya disusun dalam bentuk kumpulan komponen sistem yang saling berhubungan sesuai dengan aturan mainnya. Keuntungan dari *DFD* adalah memungkinkan untuk menggambarkan sistem dari *level* yang paling tinggi kemudian mengurakannya menjadi *level* yang lebih rendah. Sedangkan kekurangan *DFD* adalah tidak menunjukkan proses pengulangan, proses keputusan dan proses penghitungan.

Adapun langkah - langkah dalam membuat Data Flow Diagram dibagi menjadi 3 (tiga) tahap atau tingkat kontruksi *DFD*, yaitu:

### 1. Diagram Konteks

Diagram ini dibuat untuk menggambarkan sumber serta tujuan data yang akan diproses atau dengan kata lain diagram tersebut digunakan untuk menggambarkan sistem secara umum/global dari keseluruhan sistem yang ada.

### 2. Diagram Level Nol

Diagram ini dibuat untuk menggambarkan tahapan yang ada di dalam diagram konteks, yang penjabarannya lebih terperinci.

### 3. Diagram Detail

Diagram ini dibuat untuk menggambarkan arus data lebih mendetail lagi dan tahapan proses yang ada di dalam diagram *level nol* [8].

Menurut Jogiyanto HM dalam buku yang berjudul Analisis & Desain Sistem Informasi simbol yang digunakan *Data Flow Diagram (DFD)* yaitu:

#### a. *External Entity* (kesatuan luar) atau *Boundary* (batas sistem)

setiap sistem pasti mempunyai batas sistem (*boundary*) yang memisahkan suatu sistem dengan lingkungan luarnya. Sistem akan menerima *input* dan menghasilkan *output* kepada lingkungan luarnya. Kesatuan luar (*external entity*) merupakan kesatuan (*entity*) di lingkungan luar sistem yang dapat berupa orang, organisasi atau sistem lainnya yang berada dilingkungan luarnya yang akan memberikan *input* atau menerima *output* dari sistem.

#### b. *Data Flow* (arus data)

*Data Flow* (arus data) di DFD diberi simbol suatu anak panah. Arus data ini mengalir di antara proses (*process*), simpanan data (*data store*) dan kesatuan luar (*external entity*). Arus data ini menunjukkan arus dari data yang dapat berupa masukkan untuk sistem atau hasil dari proses sistem.

c. *Process (proses)*

Suatu proses adalah kegiatan atau kerja yang dilakukan oleh orang, mesin atau komputer dari hasil suatu arus data yang masuk ke dalam proses untuk dihasilkan arus data yang akan keluar dari proses. Untuk *physical data Flow diagram* (PDFD), proses dapat dilakukan oleh orang, mesin atau komputer, sedangkan untuk *logical data Flow diagram* (LDFD), suatu proses hanya menunjukkan proses dari komputer. Suatu proses dapat ditunjukkan dengan simbol lingkaran atau empat persegi panjang tegak dengan sudut-sudutnya tumpul.

d. *Data store (simpanan data)*

Simpanan data (*data store*) merupakan simpanan dari data yang dapat berupa:

- Suatu *file* atau *database* di sistem komputer
- Suatu arsip atau catatan manual
- Suatu kotak tempat data di meja seseorang
- Suatu tabel acuan manual
- Suatu agenda atau buku.

Dari uraian di atas dapat disimpulkan bahwa *DFD* adalah diagram yang menggunakan notasi-notasi ini untuk menggambarkan arus dari data sistem.

Adapun beberapa simbol yang digunakan dalam *Data Flow Diagram* yang dilukiskan pada *Table 2.2* berikut:

Tabel 2.2 Simbol-simbol *DFD*

No	Simbol	Nama	Keterangan
1		Entitas Luar	Suatu entitas eksternal adalah objek di luar sistem yang berkomunikasi dengan sistem sebagai sumber maupun pemakai informasi

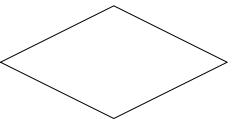
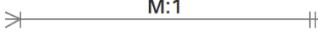
2		Proses	Suatu proses dalam sistem yang mengolah data masukkan menjadi informasi
3		Penyimpanan Data	Merupakan tempat penyimpanan data yang perlu disimpan untuk digunakan oleh satu proses atau lebih
4		Garis Alir	Untuk menunjukkan arah dari aliran data pada suatu proses

## 2.7 Entity Relationship Diagram (ERD)

*Entity Relationship Diagram (ERD)* adalah suatu diagram untuk menggambarkan desain konseptual dari model konseptual suatu basis data relasional. *ERD* juga merupakan gambaran yang menghubungkan antara objek satu dengan objek yang lain dalam dunia nyata. Sebagai contoh, jika akan membuat *ERD* dari sistem penjualan suatu perusahaan, maka bahan sebagai objek *ERD* bisa berupa barang, pembeli, *supplier*, dan lain sebagainya. Pemodelan basis data dalam bentuk *Entity Relationship Diagram* hanyalah sebagai pendekatan dan perkiraan deskripsi data pada *database* yang akan dibangun, dibangun melalui evaluasi yang sangat subjektif dari informasi yang terkumpul selama menganalisis kebutuhan sistem. Dalam pembuatan *Entity Relationship Diagram* perlu diperhatikan penentuan sesuatu konsep apakah merupakan *entity*, *attribute*, atau *relationship* [9]. Beberapa elemen-elemen *ERD* adalah sebagai berikut:

Tabel 2.3 Simbol-simbol *ERD*

No	Simbol	Nama	Keterangan
1		Entitas	Jenis entitas dapat berupa suatu elemen lingkungan, sumber daya atau transaksi yang <i>field-fieldnya</i> dipergunakan dalam aplikasi program

2		Hubungan atau Relasi	Menunjukkan nama relasi antar satu entitas dengan entitas lainnya
3		Atribut	Atribut adalah karakteristik dari sebuah entitas
4		One to One	Digunakan untuk menghubungkan antara dua tabel yang memungkinkan satu tabel berelasi dengan satu data saja pada tabel lainnya.
5		One to many	Digunakan menghubungkan antara dua tabel yang memungkinkan beberapa data tabel berelasi dengan beberapa data yang ada di tabel lain.
6		Many to many	Digunakan menghubungkan antara dua tabel yang memungkinkan beberapa data tabel berelasi dengan beberapa tabel lainnya.

## 2.8 PHP

PHP singkatan dari *PHP Hypertext Processor* yang digunakan sebagai bahasa *script server-side* dalam pengembangan *web* yang disisipkan pada dokumen *HTML*. Penggunaan *PHP* memungkinkan *web* dapat dibuat dinamis sehingga *maintenance* situs *web* tersebut menjadi lebih mudah dan efisien. *PHP* merupakan *software open source* yang disebarluaskan dan dilisensikan secara gratis serta dapat diunduh secara bebas dari situs resminya <http://www.php.net>. *PHP* ditulis dengan menggunakan bahasa C.

*PHP* diciptakan pertama kali oleh Rasmus Lerdorf pada tahun 1994. Awalnya, *PHP* digunakan untuk mencatat jumlah serta untuk mengetahui siapa saja pengunjung *homepage*-nya. Rasmus Lerdorf adalah seorang pendukung *open source*. Oleh karena itu, ia mengeluarkan Personal Home Page *Tools* versi 1.0 secara gratis, kemudian menambah kemampuan *PHP* 1.0 dan meluncurkan *PHP* 2.0. Pada tahun 1996, telah banyak digunakan dalam *website* di dunia.

Sebuah kelompok pengembang *software* yang terdiri dari Rasmus, Zeew Suraski, Andi Gutman, Stig Bakken Shane Caraveo, dan Jim Wistead bekerja sama untuk menyempurnakan *PHP* 2.0. Akhirnya, pada tahun 1998, *PHP* 3.0 diluncurkan. Penyempurnaan terus dilakukan sehingga pada tahun 2000 dikeluarkan *PHP* 4.0. Tidak sampai disitu, kemampuan *PHP* terus ditambah, dan saat ini versi terbaru yang telah dikeluarkan adalah *PHP* 7.2.x [10].

## 2.9 MariaDB

*Database* adalah sekumpulan data yang dapat diolah atau dapat dimanipulasi menggunakan suatu perangkat lunak. *Database* merupakan aspek yang sangat penting dalam sebuah *website* karena berfungsi sebagai gudang penyimpanan data yang akan diolah lebih lanjut karena dapat mengorganisasi data, menghindari duplikasi data, menghindari hubungan antar data yang tidak jelas dan juga melakukan *update* data yang rumit.

MariaDB merupakan versi pengembangan terbuka dan mandiri dari MySQL. Sejak diakuisisinya MySQL oleh Oracle pada September 2010, Monty Program sebagai penulis awal kode sumber MySQL memisahkan diri dari pengembangan dan membuat versi yang lebih mandiri yakni MariaDB [11].



Gambar 2.5 MariaDB

- Semua kemampuan MySQL dimiliki pula oleh MariaDB yakni:
1. Portabilitas. MariaDB dapat berjalan stabil pada berbagai sistem operasi seperti Windows, Linux, FreeBSD, Mac Os X Server, Solaris, Amiga, dan masih banyak lagi.
  2. Perangkat lunak sumber terbuka. MariaDB didistribusikan sebagai perangkat lunak sumber terbuka, di bawah lisensi GPL sehingga dapat digunakan secara gratis.
  3. *Multi-user*. MariaDB dapat digunakan oleh beberapa pengguna dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.

4. *Performance tuning*, MariaDB memiliki kecepatan yang menakjubkan dalam menangani *query* sederhana, dengan kata lain dapat memproses lebih banyak SQL per satuan waktu.
5. Ragam tipe data. MariaDB memiliki ragam tipe data yang sangat kaya, seperti *signed / unsigned integer, float, double, char, text, date, timestamp*, dan lain-lain.
6. Perintah dan fungsi. MariaDB memiliki operator dan fungsi secara penuh yang mendukung perintah *Select* dan *Where* dalam perintah (*query*).
7. Keamanan. MariaDB memiliki beberapa lapisan keamanan seperti *level subnetmask*, nama host, dan izin akses *user* dengan sistem perizinan yang mendetail serta sandi terenkripsi.
8. Skalabilitas dan Pembatasan. MariaDB mampu menangani basis data dalam skala besar, dengan jumlah rekaman (*records*) lebih dari 50 juta dan 60 ribu tabel serta 5 miliar baris. Selain itu batas indeks yang dapat ditampung mencapai 32 indeks pada tiap tabelnya.
9. Konektivitas. MariaDB dapat melakukan koneksi dengan klien menggunakan protokol TCP/IP, *Unix socket* (UNIX), atau *Named Pipes* (NT).
10. Pelokalan bahasa. MariaDB dapat mendeteksi pesan kesalahan pada klien dengan menggunakan lebih dari dua puluh bahasa. Meski pun demikian, bahasa Indonesia belum termasuk di dalamnya.
11. Antarmuka. MariaDB memiliki antar muka (*interface*) terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (*Application Programming Interface*).
12. Klien dan peralatan. MariaDB dilengkapi dengan berbagai peralatan (*tool*) yang dapat digunakan untuk administrasi basis data, dan pada setiap peralatan yang ada disertakan petunjuk *online*.
13. Struktur tabel. MariaDB memiliki struktur tabel yang lebih fleksibel dalam menangani *ALTER TABLE*, dibandingkan basis data lainnya semacam *PostgreSQL* ataupun *Oracle*.

## 2.10 Laravel

Laravel adalah sebuah *Framework PHP* yang dirilis di bawah lisensi MIT, dibangun dengan konsep MVC (*model view controller*). Laravel adalah pengembangan *website* berbasis MVP yang ditulis dalam *PHP* yang dirancang

untuk meningkatkan kualitas perangkat lunak dengan mengurangi biaya pengembangan awal dan biaya pemeliharaan, dan untuk meningkatkan pengalaman bekerja dengan aplikasi dengan menyediakan sintaks yang ekspresif, jelas dan menghemat waktu [12].

MVC adalah sebuah pendekatan perangkat lunak yang memisahkan aplikasi logika dari presentasi. MVC memisahkan aplikasi berdasarkan komponen-komponen aplikasi, seperti: manipulasi data, *controller*, dan *user interface*.

1. *Model*, *Model* mewakili struktur data. Biasanya *model* berisi fungsi-fungsi yang membantu seseorang dalam pengelolaan basis data seperti memasukkan data ke basis data, pembaruan data dan lain-lain.
2. *View*, *View* adalah bagian yang mengatur tampilan ke pengguna. Bisa dikatakan berupa halaman *web*.
3. *Controller*, *Controller* merupakan bagian yang menjembatani *model* dan *view*.



Gambar 2.6 Laravel

Beberapa fitur yang terdapat di Laravel:

1. *Bundles*, yaitu sebuah fitur dengan sistem pengemasan modular dan tersedia beragam di aplikasi.
2. *Eloquent ORM*, merupakan penerapan PHP lanjutan menyediakan metode internal dari pola “active record” yang menagatasi masalah pada hubungan objek *database*.
3. *Application Logic*, merupakan bagian dari aplikasi, menggunakan *controller* atau bagian *Route*.
4. *Reverse Routing*, mendefinisikan relasi atau hubungan antara *Link* dan *Route*.
5. *Restful controllers*, memisahkan logika dalam melayani *HTTP GET* and *POST*.
6. *Class Auto Loading*, menyediakan *loading* otomatis untuk *class PHP*.

7. *View Composer*, adalah kode unit logikal yang dapat dieksekusi ketika *view* sedang *loading*.
8. *IoC Container*, memungkin obyek baru dihasilkan dengan pembalikan *controller*.
9. *Migration*, menyediakan sistem kontrol untuk skema *database*.
10. *Unit Testing*, banyak tes untuk mendeteksi dan mencegah regresi.
11. *Automatic Pagination*, menyederhanakan tugas dari penerapan halaman.

## 2.11 Bootstrap

Bootstrap adalah *open-source front-end toolkit* yang dikembangkan oleh Mark Otto dan Jacob Thornton dari perusahaan yang berkecimpung dalam bidang sosial media yaitu Twitter, dirilis pada Agustus 2011. Bootstrap dikembangkan dengan tujuan untuk membantu desainer dan pengembang dalam membangun *front-end* sebuah *website* secara cepat dan efisien. Twitter Bootstrap secara cepat menjadi proyek yang paling dilihat pada GitHub dengan lebih dari 33.000 pengguna GitHub [14].



Gambar 2.7 Bootstrap

Framework Bootstrap membagi ukuran resolusi layar ke dalam 4 kategori, yaitu:

1. *Large Devices* (dengan ukuran resolusi layar 1200px).
2. *Medium Devices* (dengan ukuran resolusi layar 992px).
3. *Small Devices* (dengan ukuran resolusi layar 768px).
4. *Extra Small Devices* (dengan ukuran resolusi layar < 768px)

## 2.12 Blackbox Testing

Pengujian disisi pengembang dilakukan oleh *programmer* dari aplikasi bersangkutan atau mereka yang mengerti dan terlibat didalam pengembangan sistem tersebut. Pengujian disisi pengembang ini umum disebut dengan *Blackbox*

*Testing*, pengujian ini menekankan pada sejauh mana fungsionalitas sistem *informasi* berjalan dengan baik sesuai dengan yang diinginkan oleh pengguna dan Pengembangan [15]. Terdapat empat buah jenis pengujian pada pengujian disisi pengembangan (*Blackbox*) ini antara lain:

1. Pengujian *Interface* (Antarmuka) Aplikasi

Pengujian *interface* (antarmuka) aplikasi sistem *informasi* bertujuan untuk mengetahui fungsionalitas dari setiap elemen interface yang ada disetiap halaman aplikasi sistem *informasi* di mana setiap elemen yang ditampilkan harus sesuai dengan fungsionalitas yang dimiliki.

2. Pengujian Fungsi Dasar Sistem

Pengujian fungsi dasar sistem bertujuan untuk mengetahui sejauh mana kinerja dari setiap fungsi dasar sistem yang ada di dalam aplikasi sistem *informasi*.

3. Pengujian *Form Handle* Sistem

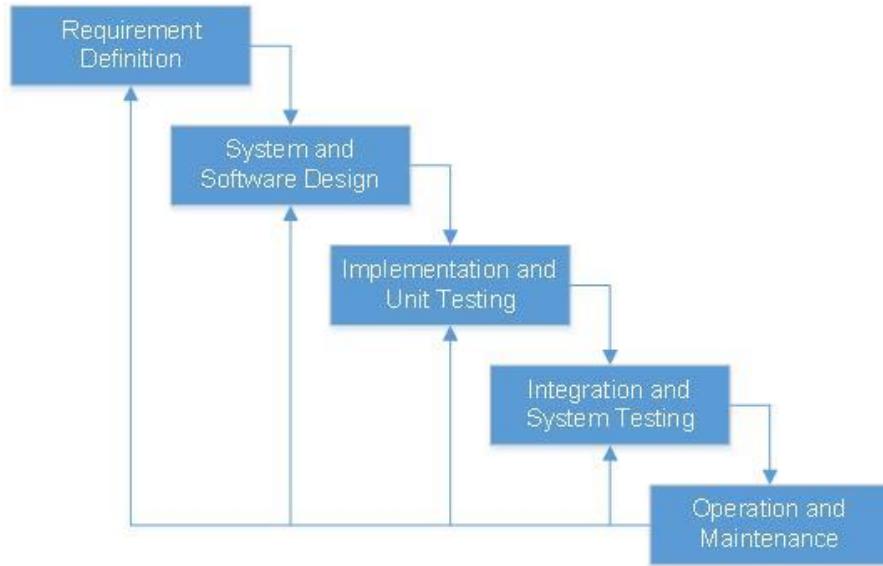
Pengujian *Form Handle* Sistem bertujuan untuk mengetahui seperti apa dan sejauh mana respon oleh sistem *informasi* terhadap masukan yang diberikan oleh pengguna di mana apabila terjadi kesalahan, sistem harus mampu memberikan *respon* agar pengguna dapat memperbaiki kesalahan yang dilakukan.

4. Pengujian Keamanan Sistem

Pengujian Keamanan Sistem bertujuan untuk mengetahui sejauh mana tingkat keamanan yang dimiliki oleh sistem *informasi* untuk dapat memberikan kenyamanan kepada para pengguna.

### 2.13 Model *Waterfall*

Prosedur pengembangan yang digunakan pada sistem *informasi* penjualan ini adalah menggunakan prosedur pengembangan model *Waterfall*. Model ini melakukan pendekatan secara sistematis dan urut mulai dari level kebutuhan sistem lalu menuju ke tahap analisis, desain, *coding*, *testing / verification*, dan *maintenance*. Disebut dengan *waterfall* karena tahap demi tahap yang dilalui harus menunggu selesainya tahap sebelumnya dan berjalan berurutan. Sebagai contoh tahap desain harus menunggu selesainya tahap sebelumnya yaitu tahap *requirement*. Secara umum tahapan pada model *waterfall* dapat dilihat pada Gambar 2.5 berikut:



Gambar 2.8 Model Waterfall

Gambar di atas adalah tahapan umum dari model proses ini. Akan tetapi Pressman memecah model ini menjadi 6 tahapan meskipun secara garis besar sama dengan tahapan-tahapan model *waterfall* pada umumnya. Berikut adalah penjelasan dari tahap-tahap yang dilakukan di dalam model ini menurut Pressman:

1. *Requirements definition*

Proses pencarian kebutuhan diintensifkan dan difokuskan pada *software*. Untuk mengetahui sifat dari program yang akan dibuat, maka para *software engineer* harus mengerti tentang domain informasi dari *software* misalnya fungsi yang dibutuhkan, *user interface*. Dari 2 aktivitas tersebut (pencarian kebutuhan sistem dan *software*) harus didokumentasikan dan ditunjukkan kepada pelanggan.

2. *Sistem and Software Design*

Proses ini digunakan untuk mengubah kebutuhan-kebutuhan diatas menjadi representasi ke dalam bentuk "blueprint" *software* sebelum *coding* dimulai. Desain harus dapat mengimplementasikan kebutuhan yang telah disebutkan pada tahap sebelumnya. Seperti 2 aktivitas sebelumnya, maka proses ini juga harus didokumentasikan sebagai konfigurasi dari *software*.

3. *Implementation and Unit Testing*

Untuk dapat dimengerti oleh mesin, dalam hal ini adalah komputer, maka desain tadi harus diubah bentuknya menjadi bentuk yang dapat dimengerti oleh mesin, yaitu ke dalam bahasa pemrograman melalui proses *coding*. Tahap ini

merupakan implementasi dari tahap *design* yang secara teknis nantinya dikerjakan oleh *programmer*.

4. *Integration and System Testing*

Sesuatu yang dibuat haruslah diujicobakan. Demikian juga dengan *software*. Semua fungsi-fungsi *software* harus diujicobakan, agar *software* bebas dari *error*, dan hasilnya harus benar - benar sesuai dengan kebutuhan yang sudah didefinisikan sebelumnya.

5. *Operation and Maintenance*

Pemeliharaan suatu *software* diperlukan, termasuk di dalamnya adalah pengembangan, karena *software* yang dibuat tidak selamanya hanya seperti itu. Ketika dijalankan mungkin saja masih ada *error* kecil yang tidak ditemukan sebelumnya, atau ada penambahan fitur-fitur yang belum ada pada *software* tersebut. Pengembangan diperlukan ketika adanya perubahan dari eksternal perusahaan seperti ketika ada pergantian sistem operasi, atau perangkat lainnya [16].



## **BAB III**

### **METODE PENELITIAN**

Metode penelitian yang digunakan dalam pembuatan “Sistem Informasi Eksekutif Penjualan Kerajinan Tas Rotan pada Balinese Classic Menggunakan Framework Laravel” adalah sebagai berikut:

#### **3.1 Waktu dan Lokasi Penelitian**

Penelitian ini dilakukan pada bulan Mei 2019 sampai Agustus 2019. Lokasi penelitian di Jalan Kebo Iwa Perumahan Swamandala XIII No.3 Denpasar, Bali. Lokasi ini merupakan kantor dan gudang dari Balinese Classic.

#### **3.2 Alat dan bahan Penelitian**

Alat dan bahan penelitian yang digunakan dalam mengumpulkan informasi di Balinese Classic menggunakan alat penelitian berupa perangkat keras dan perangkat lunak yaitu:

##### **1. Perangkat Keras**

Perangkat keras yang digunakan dalam penelitian ini adalah sebagai berikut:

- a. Laptop Asus A45a
- b. Ram 4GB
- c. Processor intel inside™ Core™ i3
- d. Hardisk 500GB

##### **2. Perangkat Lunak**

Perangkat lunak yang digunakan dalam penelitian ini adalah sebagai berikut:

- a. Windows 10
- b. Visual Studio Code
- c. Mozilla Firefox
- d. Google Chrome
- e. Xampp
- f. Ms Word
- g. Visual Paradigm

### 3.3 Metode Pengumpulan Data

Pada perekayasaan ini digunakan teknik pengumpulan data sebagai berikut:

1. *Observasi*

Melakukan pengamatan langsung pada Balinese Classic dengan menggunakan alat penunjang yaitu buku, pulpen, dan lain-lain. Melihat langsung mengenai kegiatan operasional perusahaan seperti kegiatan transaksi, proses pencatatan hingga pembelian barang. Serta melihat studi kasus apa yang sering terjadi dalam pengambilan keputusan di perusahaan tersebut.

2. *Studi Literatur*

Pada tahap ini dilakukan pengumpulan data dan *informasi* dengan cara membaca buku-buku dan artikel ilmiah yang berhubungan dengan objek perekayasaan, seperti: *website*, *PHP*, *Framework Laravel* dan lainnya.

3. *Wawancara*

Wawancara merupakan teknik pengumpulan data yang dilakukan melalui tatap muka dan Tanya jawab langsung antara peneliti terhadap narasumber atau sumber data. Dalam tahap ini, penulis melakukan wawancara dengan anggota dari Balinese Classic mengenai apa saja kesulitan dan kebutuhan yang diperlukan sehingga dapat dijadikan pedoman untuk membuat sistem ini nantinya.

### 3.4 Metode Perekayasaan

Metode perekayasaan yang digunakan pada penelitian ini adalah sebagai berikut:

1. *Analisis Sistem*

Setelah *informasi* dikumpulkan maka dilakukan sebuah analisa kebutuhan terhadap sistem yang akan dibangun. Pada tahap ini akan dilakukan klasifikasi tipe pengguna dan batasannya, alur kerja sistem, kebutuhan penyimpanan dan desain yang disusun secara sistematis, selain itu juga akan dilakukan hipotesis untuk menentukan jawaban sementara terhadap rumusan masalah pada perekayasaan.

2. *Perancangan Sistem*

Tahap perancangan merupakan tahap untuk menciptakan sesuatu konsep kerja terpadu antara manusia dan mesin sehingga dapat menghasilkan suatu *informasi* yang akurat. Perancangan sistem akan dibuat berdasarkan dari analisa

kebutuhan sehingga dapat dirancang *DFD*, *ERD*, dan Konseptual *database* yang sesuai.

### 3. Pembuatan Program

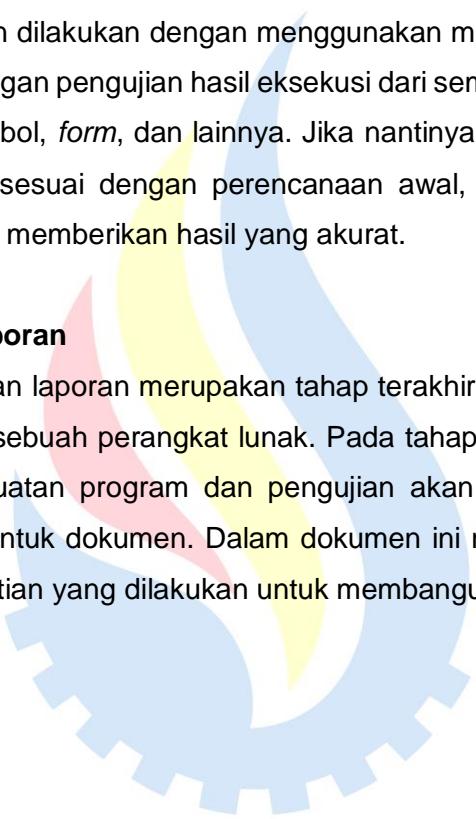
Proses pembuatan program berbasis *web* ini dari sebuah desain ke dalam program menggunakan Bahasa pemrograman *PHP* menggunakan *Framework* Laravel. Perangkat Lunak *code editor* yang akan digunakan yaitu Visual Code Studio serta pengolahan *database* menggunakan *PhpMyAdmin*.

### 4. Pengujian Sistem

Pengujian akan dilakukan dengan menggunakan metode pengujian *Black Box Testing*, Yaitu dengan pengujian hasil eksekusi dari semua fungsi-fungsi yang digunakan seperti tombol, *form*, dan lainnya. Jika nantinya hasil eksekusi fungsi-fungsi tersebut tidak sesuai dengan perencanaan awal, maka akan langsung diperbarui sehingga memberikan hasil yang akurat.

### 3.5 Penulisan Laporan

Tahap penulisan laporan merupakan tahap terakhir dari proses penelitian terhadap pembuatan sebuah perangkat lunak. Pada tahap ini, hasil dari analisa, desain sistem, pembuatan program dan pengujian akan dirangkum ke dalam sebuah laporan berbentuk dokumen. Dalam dokumen ini nantinya akan tercatat semua aktivitas penelitian yang dilakukan untuk membangun aplikasi ini.



INSTITUT TEKNOLOGI DAN BISNIS  
**STIKOM BALI**



## **BAB IV**

### **HASIL DAN PEMBAHASAN**

#### **4.1 Hasil Analisis**

Analisis sistem merupakan proses penganalisaan sebuah sistem yang bertujuan untuk menganalisa atau merancang suatu sistem. Tahapan ini merupakan tahapan yang terpenting karena jika terjadi kesalahan pada tahap ini, maka akan menyebabkan kesalahan pada tahapan berikutnya.

Adapun analisis kebutuhan sistem terdiri dari:

##### **1. Kebutuhan Fungsional**

Kebutuhan fungsional adalah kebutuhan yang berisi proses-proses apa saja yang dilakukan oleh sistem. Analisa kebutuhan fungsional terdiri dari:

- a. Sistem dapat menyediakan *login* untuk admin
  - 1) Admin dapat *login* ke dalam halaman sistem.
  - 2) Admin dapat mengelola dan mengakses data dari aplikasi meliputi data *users*, data *customers*, data *suppliers*, data *categories*, data *products*, data *stock*, data penjualan, data pembelian, dan data laporan.
  - 3) Admin dapat menambah semua data ke dalam sistem, seperti tambah data *users*, tambah data *customers*, tambah data *suppliers*, tambah data *categories*, tambah data *products*, tambah data *stock*, tambah data penjualan, tambah data pembelian.
  - 4) Admin dapat mengedit semua data di dalam sistem, seperti *edit* data *users*, *edit* data *customers*, *edit* data *suppliers*, *edit* data *categories*, *edit* data *products*, *edit* data *stock*, *edit* data penjualan, *edit* data pembelian.
  - 5) Admin dapat melihat laporan dan mengekspor data di dalam sistem.
- b. Sistem dapat menyediakan *login* untuk kasir
  - 1) Kasir dapat *login* ke dalam halaman sistem.
  - 2) Kasir dapat mengelola dan mengakses data dari aplikasi meliputi data *customers*, data *categories*, data *products*, data penjualan.
  - 3) Kasir dapat menambah data ke dalam sistem, seperti tambah data *customers*, tambah data *categories*, tambah data *products*, tambah data penjualan.

- 4) Kasir dapat mengedit data di dalam sistem, seperti *edit* data *customers*, *edit* data *categories*, *edit* data *products*, *edit* data penjualan.
- c. Sistem dapat menyediakan *login* untuk Gudang
  - 1) Gudang dapat *login* ke dalam halaman sistem.
  - 2) Gudang dapat mengelola dan mengakses data dari aplikasi meliputi data *suppliers*, data *categories*, data *products*, data *stock* dan data pembelian.
  - 3) Gudang dapat menambah data ke dalam sistem, seperti tambah data *suppliers*, data *categories*, data *products*, data *stock* dan data pembelian.
  - 4) Gudang dapat mengedit data di dalam sistem, seperti *edit* data *suppliers*, data *categories*, data *products*, data *stock* dan data pembelian.

## 2. Kebutuhan Non Fungsional

Kebutuhan non fungsional ini merupakan kebutuhan yang memenuhi dalam pembuatan sistem. Adapun kebutuhan non fungsional pada sistem yaitu:

### a. Perangkat keras(*hardware*)

Untuk merancang dan membuat sistem *informasi* dibutuhkan perangkat keras agar sistem yang dibuat dapat berjalan dengan baik. Spesifikasi *hardware* yang digunakan adalah sebagai berikut:

- 1) Laptop : Asus X441U Series
- 2) Processor : Intel® Core™ i3-6006U CPU @ 2.00GHz 1.99 GHz
- 3) Memory : 4.00 GB
- 4) Sistem Type : 64-bit Operating Sistem

### b. Perangkat Lunak(*Software*)

Untuk mendukung dan merancang pembuatan sistem dibutuhkan perangkat lunak agar sistem yang dibuat dapat berjalan dengan baik. Perangkat lunak yang digunakan adalah sebagai berikut:

- 1) Sistem operasi : Windows 10
- 2) Bahasa Pemrograman : PHP, Ajax
- 3) DBMS : MariaDB
- 4) Framework : Laravel, Bootstrap
- 5) Code Editor : Visual Code Studio

6) Design UX : Balsamiq

3. Analisis User

Dari hasil analisa, terdapat 3 (tiga) tipe *user* yang dapat mengakses sistem.

Berikut adalah penjabaran *user* pada *table 4.1 analisa users*:

Tabel 4.1 Analisa User

No	Users	Admin
1	Admin	Admin adalah tipe <i>user</i> yang memiliki otoritas paling tinggi, dapat mengelola data dan informasi yang terdapat pada sistem yaitu mengelola data <i>user</i> , data-data transaksi, data kategori, data produk, data stok, data <i>customer</i> , data <i>supplier</i> dan melihat laporan.
2	Kasir	Kasir adalah tipe <i>user</i> yang dapat mengelola data pada bagian kasir, seperti data transaksi jual, data kategori, data produk, dan data <i>customer</i> .
3	Gudang	Gudang merupakan tipe <i>user</i> yang dapat mengelola data pada bagian Gudang, seperti data transaksi beli, data kategori, data produk, data stok, dan data <i>supplier</i> .

Berikut merupakan *mapping* dari aplikasi ini pada *table 4.2*:

Tabel 4.2 Mapping Hak Akses

Proses	Admin	Kasir	Gudang
Dashboard	✓	✗	✗
User	✓	✗	✗
Penjualan	✓	✓	✗
Pembelian	✓	✗	✓
Kategori	✓	✓	✓
Produk	✓	✓	✓
Stok	✓	✗	✓
Customer	✓	✓	✗
Supplier	✓	✗	✓

Laporan	V	X	X
---------	---	---	---

#### 4. Analisa Data

Pada sistem yang akan dibangun dapat diidentifikasi beberapa data yang diperlukan agar sistem dapat dijalankan sesuai dengan yang direncanakan. Hasil Analisa data ini selanjutnya akan dijadikan acuan dalam pembuatan data *store* pada perancangan sistem dan *table* pada basis data. Adapun data yang dimaksud dapat dilihat pada *table* 4.3 di bawah ini:

Tabel 4.3 Analisa Data

No	Data	Keterangan
1	User	Adalah data yang berisikan informasi mengenai data pengguna dan data yang digunakan untuk <i>login</i> ke sistem.
2	Penjualan	Adalah data yang berisikan informasi mengenai transaksi penjualan
3	Penjualan Produk	Adalah data yang berisikan informasi produk dari transaksi penjualan
4	Pembelian	Adalah data yang berisikan informasi mengenai transaksi pembelian
5	Pembelian Produk	Adalah data yang berisikan informasi produk dari transaksi pembelian
6	Kategori	Adalah data yang berisikan informasi kategori dari produk
7	Produk	Adalah data yang berisikan informasi mengenai produk
8	Stok	Adalah data yang berisikan jumlah stok dari produk
9	Customer	Adalah data yang berisikan informasi pelanggan
10	Supplier	Adalah data yang berisikan informasi pemasok

## 5. Analisis Proses

Terdapat 4 proses dalam sistem ini. Proses ini merupakan gambaran umum dari fungsi-fungsi yang dimiliki sistem dalam pengelolaan terhadap suatu data yang ada. Pada Tabel 4.4 berikut akan dijelaskan proses yang ada:

Tabel 4.4 Analisa Proses

No	Proses	Keterangan
1	<i>Login</i>	Merupakan proses awal yang harus dilakukan oleh semua <i>Users</i> agar dapat mengoperasikan sistem. <i>Login user</i> dibedakan berdasarkan level guna memberikan hak akses yang berbeda kepada setiap pengguna aplikasi ini
2	<i>Users</i>	Merupakan proses pengolahan data yang dilakukan pada data <i>user</i> , data <i>customer</i> , data <i>supplier</i>
3	Transaksi	Merupakan proses pengolahan data utama dalam sistem ini. Proses yang dilakukan adalah mengelola data transaksi penjualan dan transaksi pembelian
4	Produk	Merupakan proses pengolahan data-data produk seperti data kategori, data produk dan data stok produk
5	Laporan	Merupakan proses mencetak laporan – laporan data transaksi

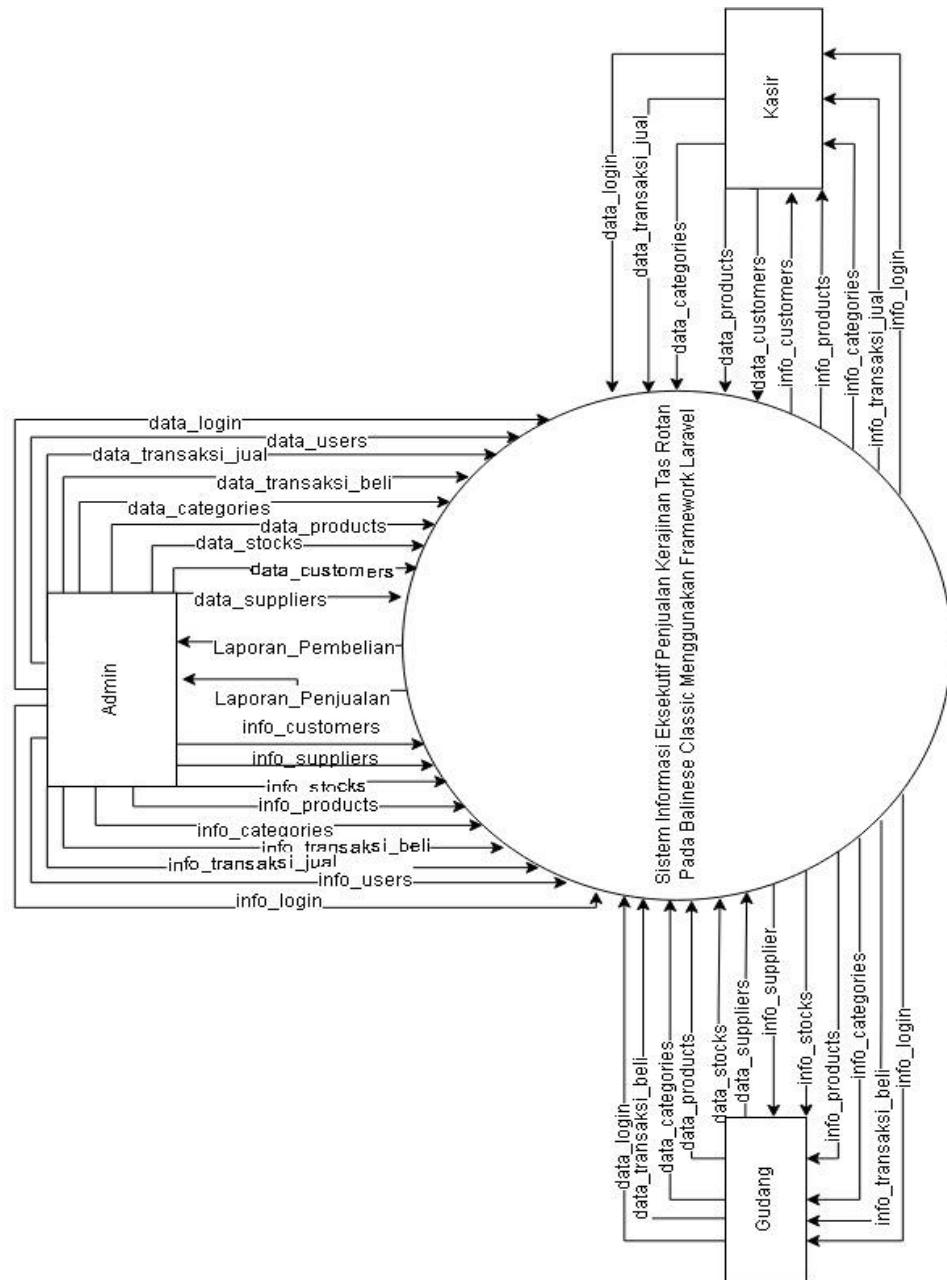
## 4.2 Perancangan Sistem

Pada tahapan perancangan sistem ini digunakan Data Flow Diagram (DFD), untuk menggambarkan aliran data yang terjadi dalam proses sistem. Perancangan sistem ini memiliki beberapa tahap, dimulai dari diagram konteks, DFD level 0 dan DFD level 1 yang lebih detail. Berikut ini adalah hasil perancangan sistem menggunakan DFD:

### 1. Diagram Konteks

Diagram Konteks merupakan gambaran secara umum bagaimana sistem ini bekerja. Gambar 4.1 di bawah ini merupakan diagram konteks dari Sistem Informasi Eksekutif Penjualan Kerajinan Tas Rotan Pada Balinese Classic

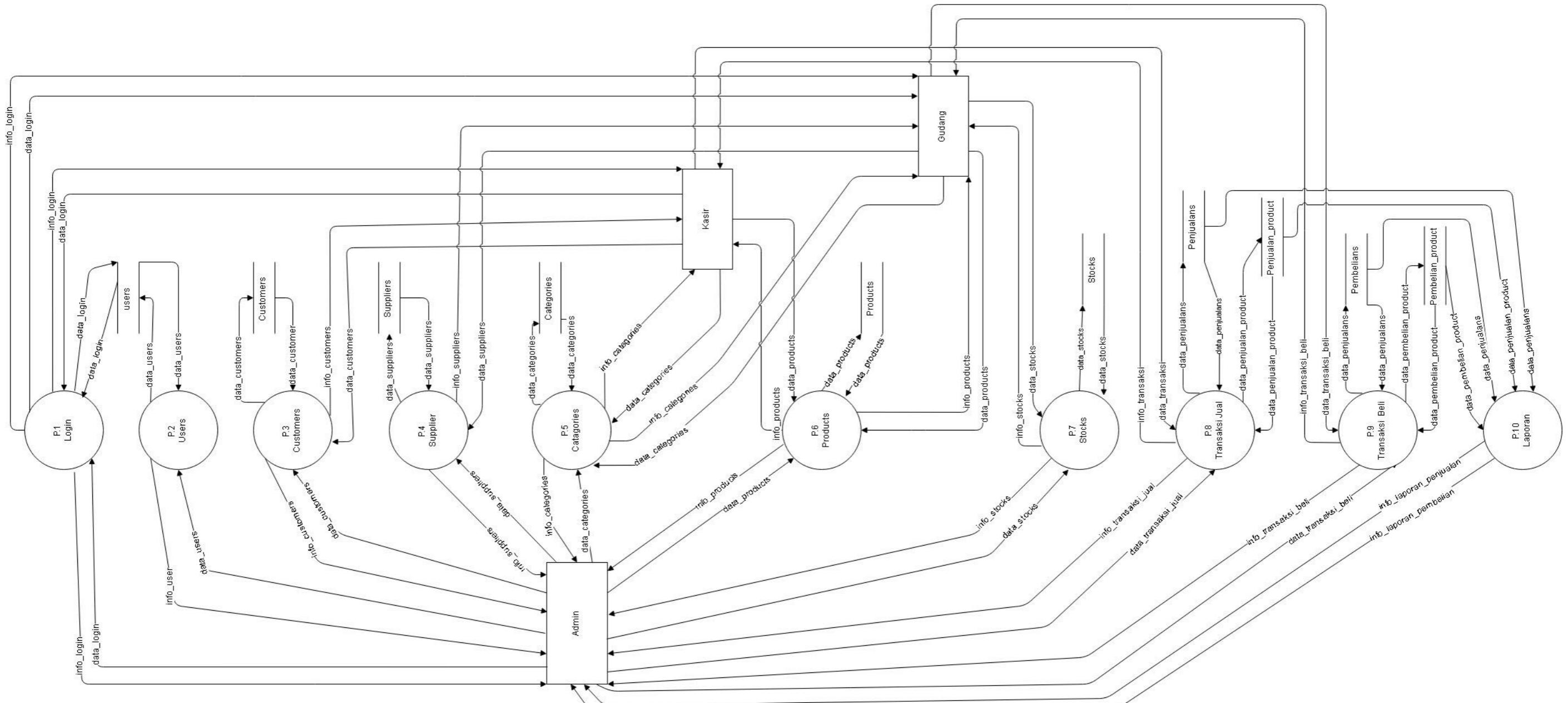
Menggunakan *Framework Laravel* yang merupakan dasar yang digunakan dalam penyusunan sistem ke level berikutnya. Terdapat 3 *user* yang dapat mengakses sistem ini yaitu, Admin, Kasir dan Gudang.



Gambar 4.1 Diagram Konteks

## 2. Data Flow Diagram (DFD) Level 0

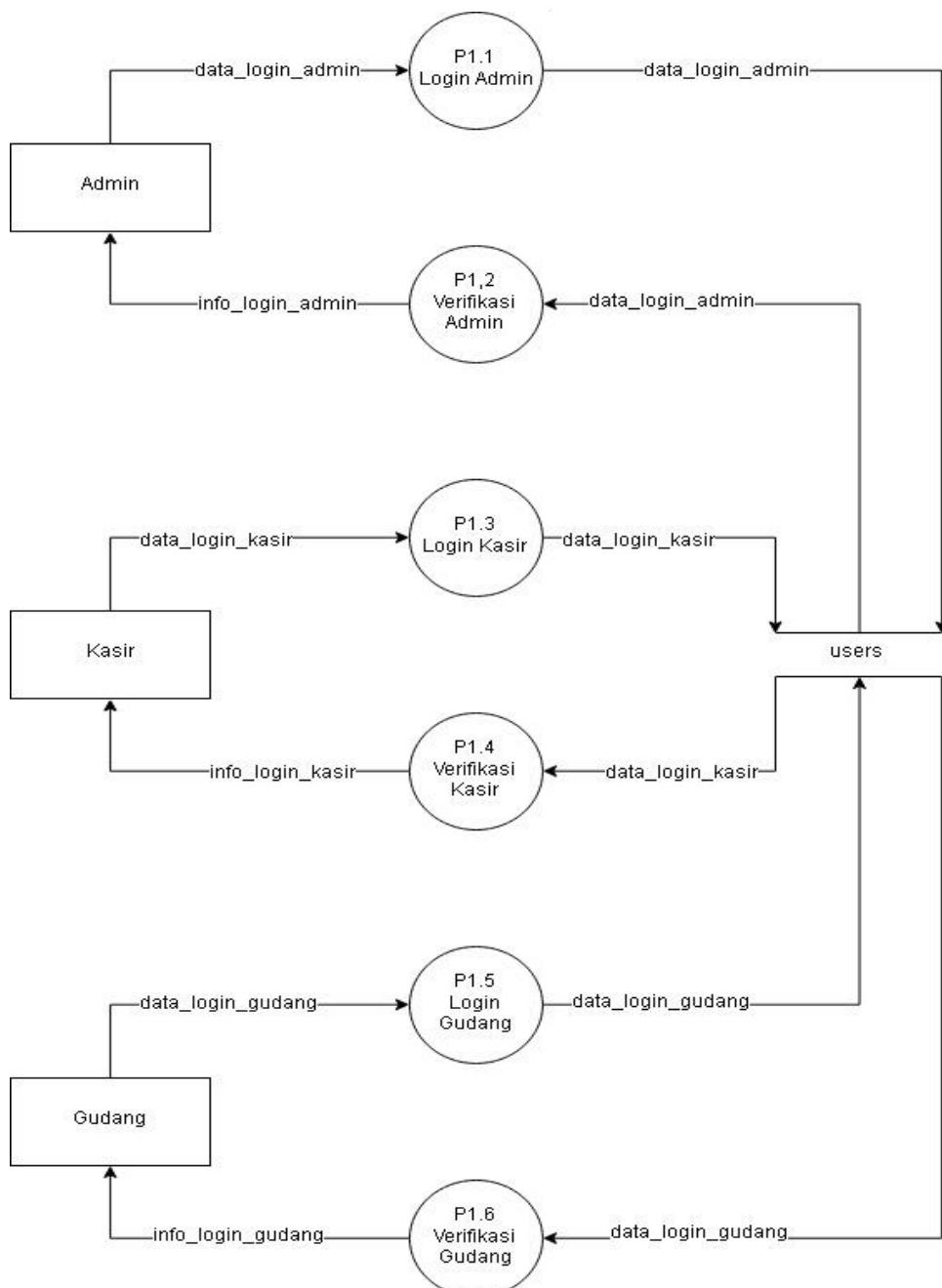
Data Flow Diagram Level 0 menggambarkan arus data yang lebih jelas dan detail yang dikelompokkan ke dalam proses\_proses yang lebih kecil beserta dengan data store yang terlibat. Berikut ini adalah gambar DFD Level 0 dari sistem ini:



Gambar 4.2 Data Flow Diagram (DFD) Level 0

3. Data Flow Diagram (DFD) Level 1 Proses 1 *Login*

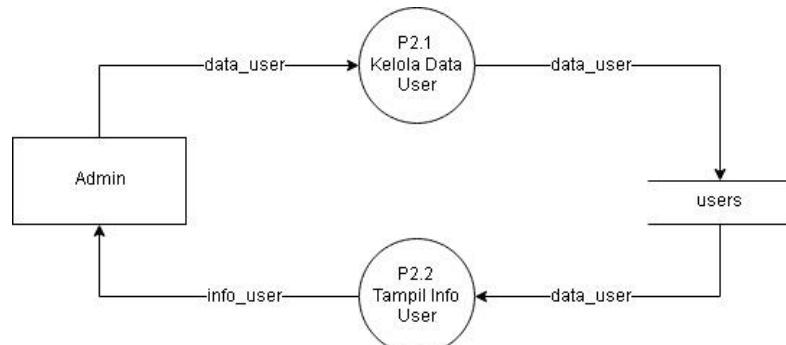
DFD Level 1 *Login* menggambarkan arus data yang lebih jelas dan detail di dalam sistem pada proses *login*. Proses 1.1(*Login Admin*) merupakan proses data *login* admin yang diverifikasi ke *database users*. Proses 1.3(*Login Kasir*) merupakan proses data *login* pegawai kasir yang diverifikasi ke *database users*. Proses 1.5(*Login Gudang*) merupakan proses data *login* pegawai Gudang yang diverifikasi ke *database users*. Berikut adalah gambar 4. Data Flow Diagram (DFD) Level 1 Proses *Login*:



Gambar 4.3 Data Flow Diagram (DFD) Level 1 Proses 1 *Login*

4. Data Flow Diagram (DFD) Level 1 Proses 2 Users

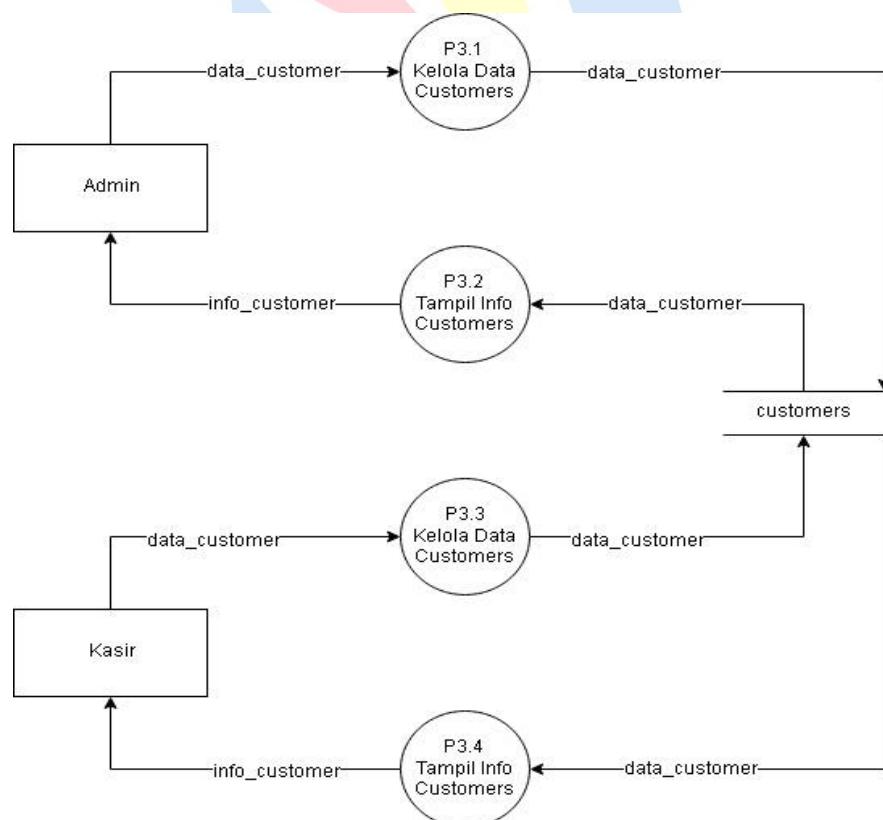
Data Flow Diagram (DFD) Level 1 Proses *Users* menggambarkan arus data pada proses kelola data *users*. Proses kelola data *users* ini dilakukan oleh admin saja. Berikut ini adalah DFD Level 1 Proses *Users* dari sistem ini:



Gambar 4.4 Data Flow Diagram (DFD) Level 1 Proses 2 *Users*

5. Data Flow Diagram (DFD) Level 1 Proses 3 Customers

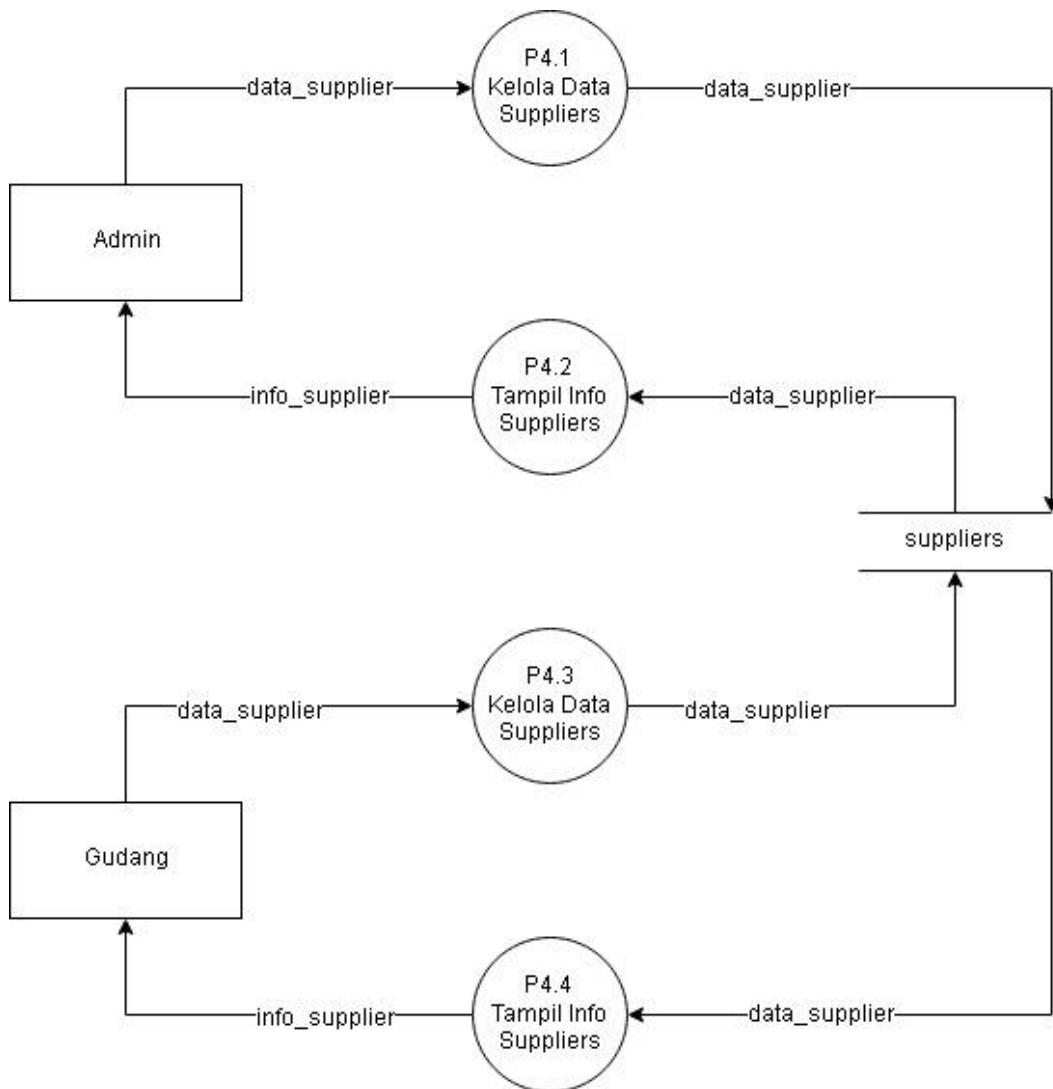
Data Flow Diagram (DFD) Level 1 Proses 3 *Customers* menggambarkan arus data pada proses kelola data *customers*. Proses kelola data *customers* ini dilakukan oleh admin dan kasir. Berikut ini adalah DFD Level 1 Proses *Customers* dari sistem ini:



Gambar 4.5 Data Flow Diagram (DFD) Level 1 Proses 3 *Customers*

6. Data Flow Diagram (DFD) Level 1 Proses 4 Suppliers

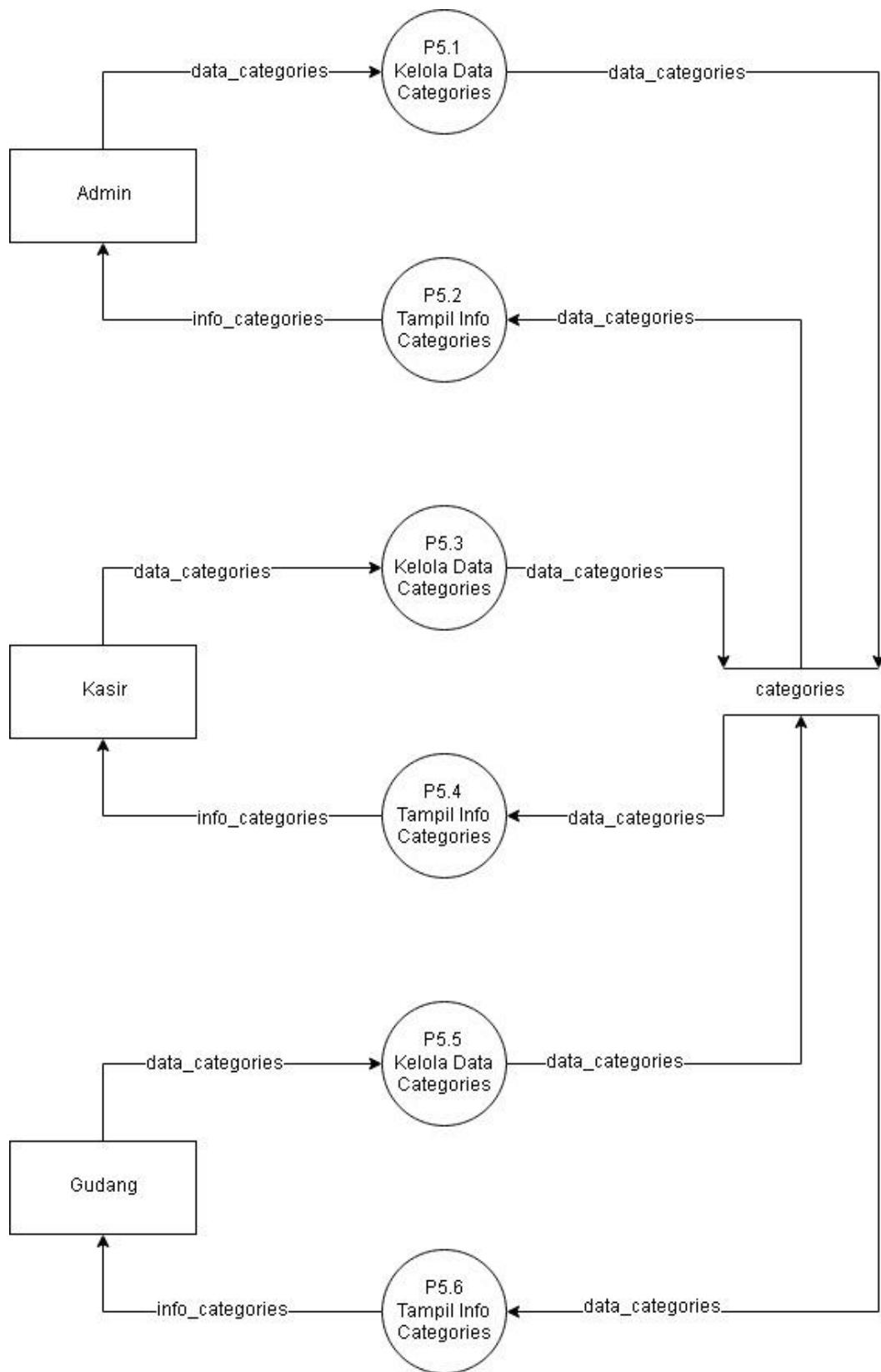
Data Flow Diagram (DFD) Level 1 Proses 4 Suppliers menggambarkan arus data pada proses kelola data *customers*. Proses kelola data *suppliers* ini dilakukan oleh admin dan gudang. Berikut ini adalah DFD Level 1 Proses *Suppliers* dari sistem ini:



Gambar 4.6 Data Flow Diagram (DFD) Level 1 Proses 4 Suppliers

7. Data Flow Diagram (DFD) Level 1 Proses 5 Categories

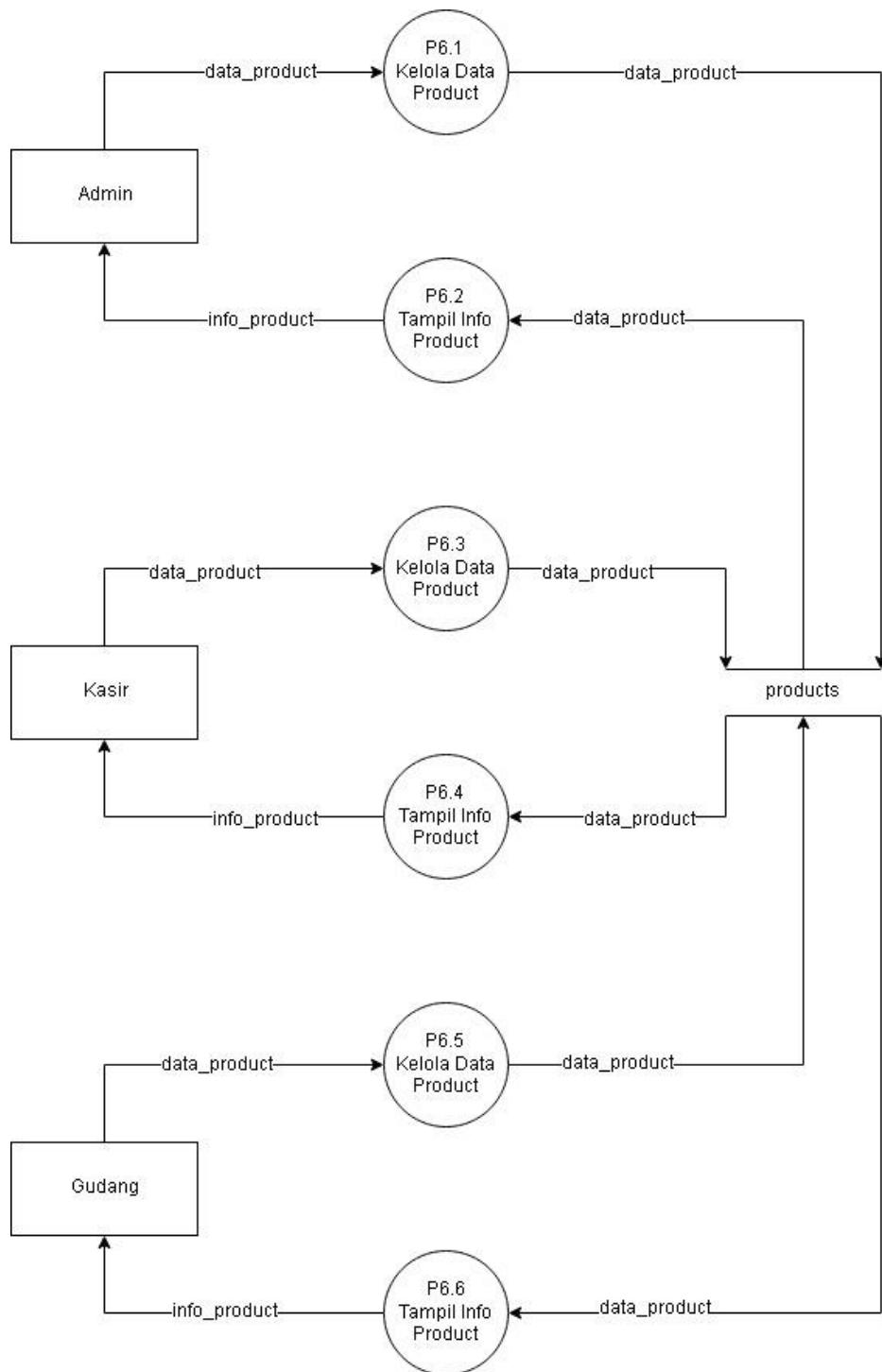
Data Flow Diagram (DFD) Level 1 Proses 5 Categories menggambarkan arus data pada proses kelola data *categories*. Proses kelola data *categories* ini dilakukan oleh admin, kasir dan gudang. Berikut ini adalah DFD Level 1 Proses 5 *Categories* dari sistem ini:



Gambar 4.7 Data Flow Diagram (DFD) Level 1 Proses 5 Categories

8. Data Flow Diagram (DFD) Level 1 Proses 6 Products

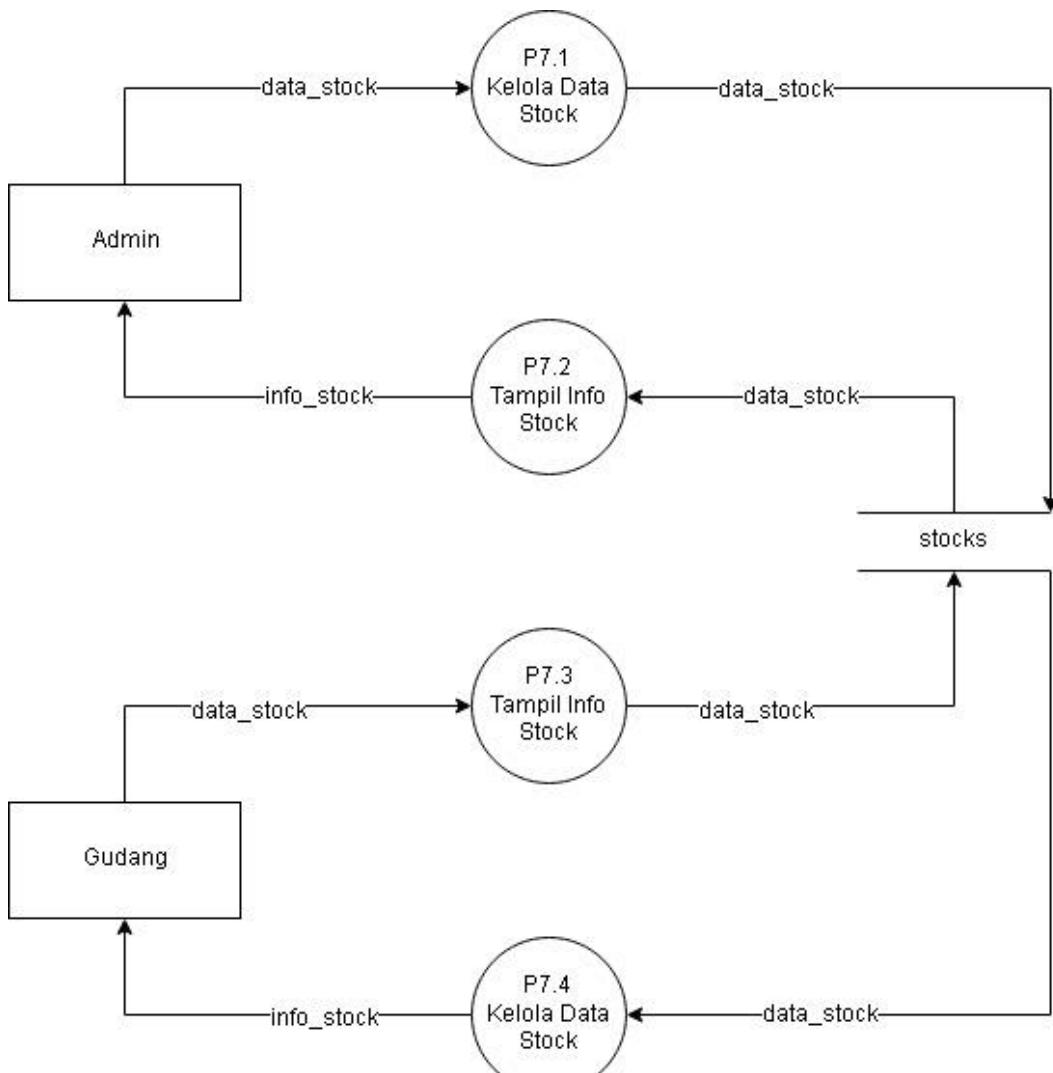
Data Flow Diagram (DFD) Level 1 Proses 6 Products menggambarkan arus data pada proses kelola data *product*. Proses kelola data *products* ini dilakukan oleh admin, kasir dan gudang. Berikut ini adalah DFD Level 1 Proses 6 Products dari sistem ini:



Gambar 4.8 Data Flow Diagram (DFD) Level 1 Proses 6 Product

9. Data Flow Diagram (DFD) Level 1 Proses 7 Stocks

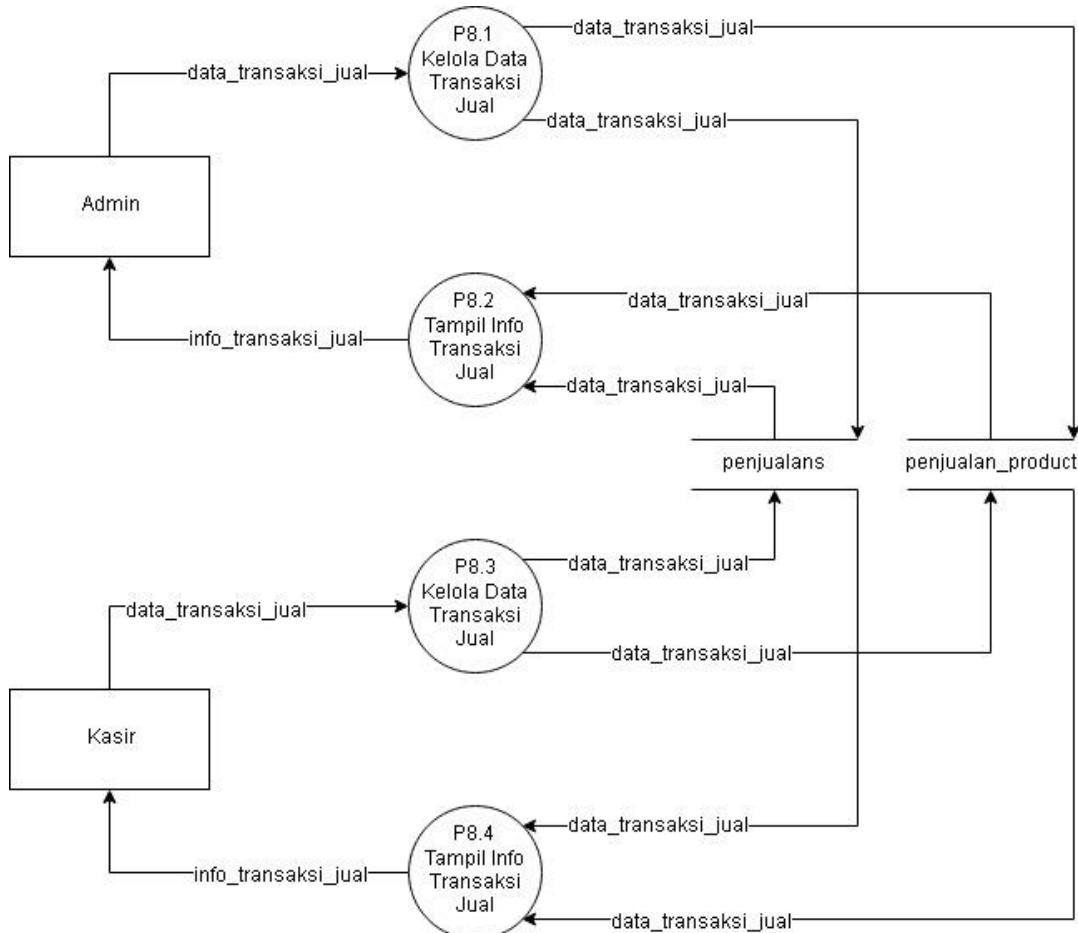
Data Flow Diagram (DFD) Level 1 Proses 7 Stocks menggambarkan arus data pada proses kelola data stocks. Proses kelola data stock ini dilakukan oleh admin dan gudang. Berikut ini adalah DFD Level 1 Proses 7 Stocks dari sistem ini:



Gambar 4.9 Data Flow Diagram (DFD) Level 1 Proses 7 Stock

10. Data Flow Diagram (DFD) Level 1 Proses 8 Transaksi Jual

Data Flow Diagram (DFD) Level 1 Proses 8 Transaksi Jual menggambarkan arus data pada proses kelola data transaksi jual. Proses kelola data transaksi jual ini dilakukan oleh admin dan kasir. Berikut ini adalah DFD Level 1 Proses 8 Transaksi Jual dari sistem ini:

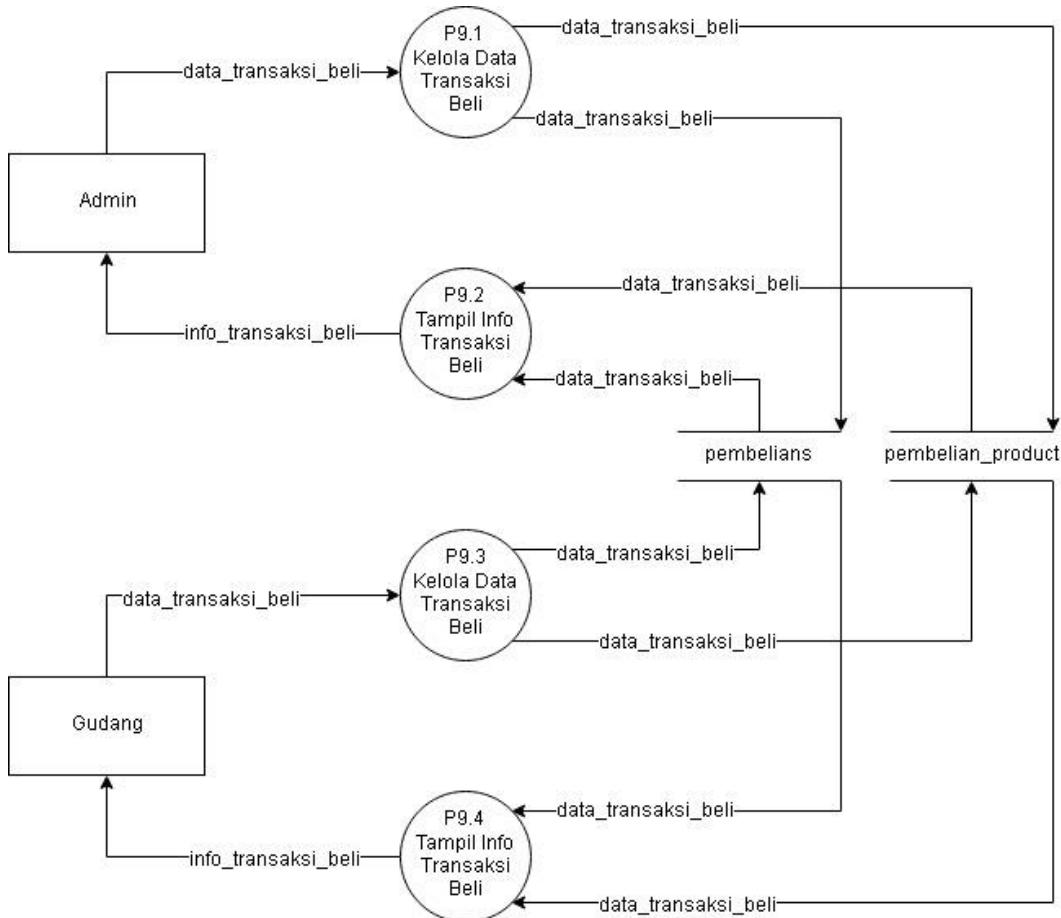


Gambar 4.10 Data Flow Diagram (DFD) Level 1 Proses 8 Transaksi Jual

#### 11. Data Flow Diagram (DFD) Level 1 Proses 9 Transaksi Beli

Data Flow Diagram (DFD) Level 1 Proses 9 Transaksi Beli menggambarkan arus data pada proses kelola data transaksi beli. Proses kelola data transaksi beli ini dilakukan oleh admin dan gudang. Berikut ini adalah DFD Level 1 Proses 9 Transaksi Beli dari sistem ini:

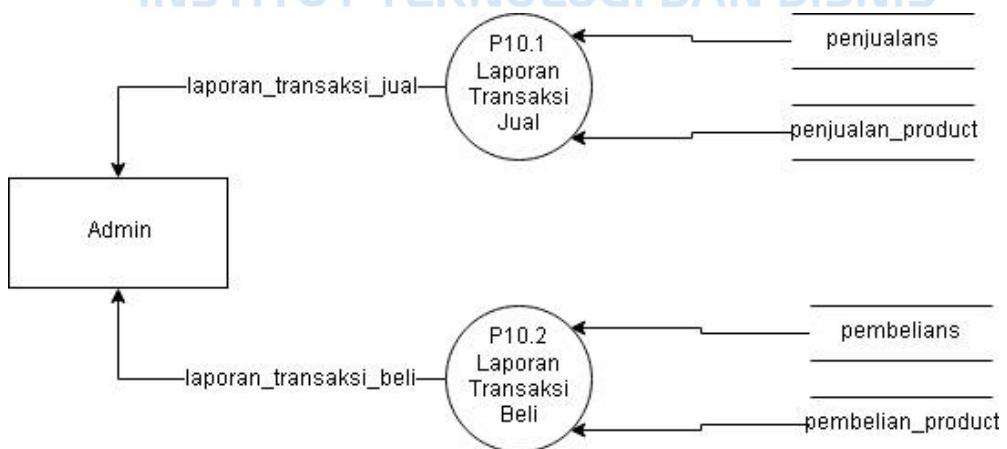
**STIKOM BALI**



Gambar 4.11 Data Flow Diagram (DFD) Level 1 Proses 9 Transaksi Beli

#### 12. Data Flow Diagram (DFD) Level 1 Proses 10 Laporan

Data Flow Diagram (DFD) Level 1 Proses 10. Laporan. Proses 10.1 (laporan transaksi jual) dan Proses 10.2 (laporan transaksi beli) merupakan gambaran data-data yang sudah masuk dari transaksi yang telah dilakukan oleh sistem ini. Berikut ini adalah gambar 4. DFD Level 1 Proses 10 Laporan:



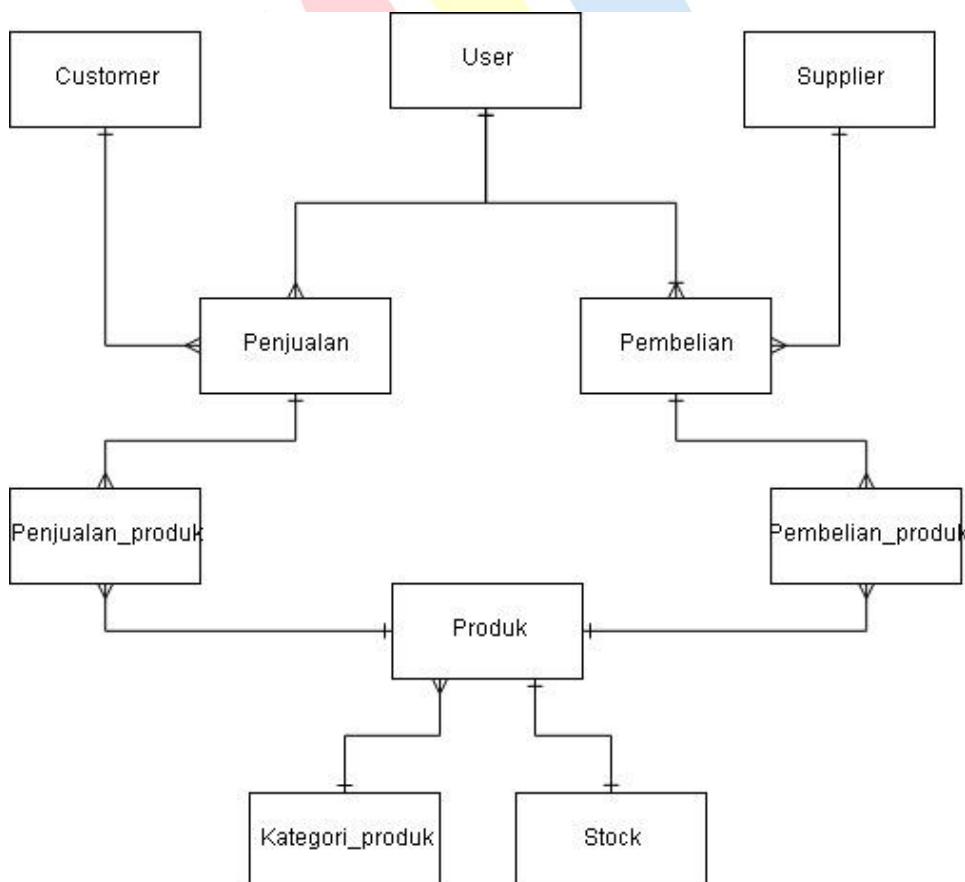
Gambar 4.12 Data Flow Diagram (DFD) Level 1 Proses 10 Laporan

### 4.3 Perancangan Basis Data

Pada tahapan ini dilakukan perancangan basis data dari Sistem Informasi Eksekutif Penjualan Tas Rotan Pada Balinese Classic Menggunakan Framework Laravel yang terdiri dari *Entity Relationship Diagram* (ERD), Basis Data Konseptual dan Struktur Tabel, di antaranya sebagai berikut:

#### 1. Entity Relationship Diagram

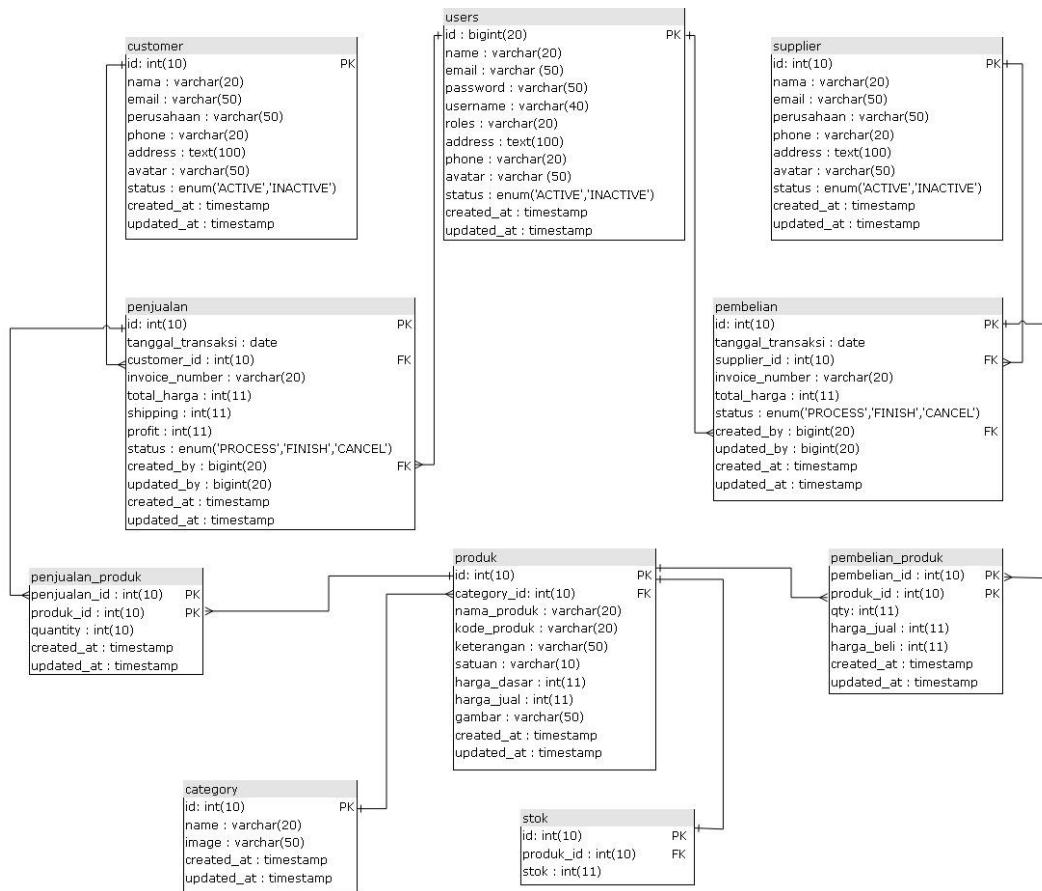
*Entity Relationship Diagram* adalah sebuah diagram yang secara konseptual memetakan hubungan antar penyimpanan pada diagram DFD. Pada perancangan ini terdapat ERD (*Entity Relationship Diagram*) yang berisikan 10 buah *table*. *User* memiliki relasi *one to many* dengan penjualan dan pembelian, *customer* memiliki relasi *one to many* dengan penjualan, *supplier* memiliki relasi *one to many* dengan pembelian, penjualan memiliki relasi *one to many* dengan penjualan\_produk, pembelian memiliki relasi *one to many* dengan pembelian\_produk, penjualan\_produk memiliki relasi *many to one* dengan produk, pembelian\_produk memiliki relasi *one to many* dengan produk, produk memiliki relasi *many to one* dengan kategori\_produk, produk memiliki relasi *one to one* dengan stock. Berikut gambar 4. Merupakan ERD dari sistem yang dibangun:



Gambar 4.13 *Entity Relationship Diagram*

## 2. Basis Data Konseptual

Basis Data Konseptual pada Sistem Informasi Eksekutif Penjualan Kerajinan Tas Rotan Pada Balinese Classic Menggunakan Framework Laravel ini terdiri dari 7 tabel seperti pada Gambar 4.14 Berikut:



Gambar 4.14 Basis Data Konseptual

## 3. Struktur Tabel

Pada penyimpanan basis data diperlukan kumpulan *table* - *table* yang digunakan untuk menyimpan data. Data yang dikelola akan disimpan pada *field-field* terkait sesuai dengan *table* yang digunakan. Berikut ini adalah struktur *table* dari sistem yang dibuat.

### a. Struktur Tabel User

Digunakan untuk menyimpan seluruh data *user* pada Sistem Informasi Eksekutif Penjualan Kerajinan Tas Rotan Pada Balinese Classic Menggunakan Framework Laravel. Berikut struktur *table user* pada Tabel 4.5.

Tabel 4.5 Struktur Tabel *Users*

No	Nama Atribut	Tipe Data	Ukuran	Key Constraint
1	<i>id</i>	<i>bigint</i>	20	<i>Primary Key</i>
2	<i>name</i>	<i>varchar</i>	20	
4	<i>password</i>	<i>varchar</i>	50	
5	<i>username</i>	<i>varchar</i>	40	
6	<i>roles</i>	<i>varchar</i>	20	
7	<i>address</i>	<i>text</i>	100	
8	<i>phone</i>	<i>varchar</i>	20	
9	<i>avatar</i>	<i>varchar</i>	50	
10	<i>status</i>	<i>enum</i>	ACTIVE,INACTIVE	
11	<i>created_at</i>	<i>timestamp</i>		
12	<i>updated_at</i>	<i>timestamp</i>		

b. Struktur Tabel *Customer*

Digunakan untuk menyimpan seluruh data *customer* pada Sistem Informasi Eksekutif Penjualan Kerajinan Tas Rotan Pada Balinese Classic Menggunakan Framework Laravel. Berikut struktur *table customer* pada Tabel 4.6.

Tabel 4.6 Struktur Tabel *Customers*

No	Nama Atribut	Tipe Data	Ukuran	Key Constraint
1	<i>id</i>	<i>int</i>	10	<i>Primary Key</i>
2	<i>nama</i>	<i>varchar</i>	20	
3	<i>email</i>	<i>varchar</i>	50	
4	<i>perusahaan</i>	<i>varchar</i>	50	
5	<i>phone</i>	<i>varchar</i>	20	
6	<i>address</i>	<i>text</i>	100	
7	<i>avatar</i>	<i>varchar</i>	50	
8	<i>status</i>	<i>enum</i>	ACTIVE,INACTIVE	
9	<i>created_at</i>	<i>timestamp</i>		
10	<i>updated_at</i>	<i>timestamp</i>		

c. Struktur Tabel *Supplier*

Digunakan untuk menyimpan seluruh data *supplier* pada Sistem Informasi Eksekutif Penjualan Kerajinan Tas Rotan Pada Balinese Classic Menggunakan Framework Laravel. Berikut struktur *table supplier* pada Tabel 4.7.

Tabel 4.7 Struktur Tabel *Suppliers*

No	Nama Atribut	Tipe Data	Ukuran	Key Constraint
1	<i>id</i>	<i>int</i>	10	<i>Primary Key</i>
2	nama	<i>varchar</i>	20	
3	email	<i>varchar</i>	50	
4	perusahaan	<i>varchar</i>	50	
5	<i>phone</i>	<i>varchar</i>	20	
6	<i>address</i>	<i>text</i>	100	
7	<i>avatar</i>	<i>varchar</i>	50	
8	status	<i>enum</i>	ACTIVE,INACTIVE	
9	<i>created_at</i>	<i>timestamp</i>		
10	<i>updated_at</i>	<i>timestamp</i>		

d. Struktur Tabel *Category*

Digunakan untuk menyimpan seluruh data kategori pada Sistem Informasi Eksekutif Penjualan Kerajinan Tas Rotan Pada Balinese Classic Menggunakan Framework Laravel. Berikut struktur *table* kategori pada Tabel 4.8:

Tabel 4.8 Struktur Tabel *Categories*

No	Nama Atribut	Tipe Data	Ukuran	Key Constraint
1	<i>id</i>	<i>int</i>	10	<i>Primary Key</i>
2	<i>name</i>	<i>varchar</i>	20	
3	<i>image</i>	<i>varchar</i>	50	
4	<i>created_at</i>	<i>timestamp</i>		
5	<i>updated_at</i>	<i>timestamp</i>		

e. Struktur Tabel *Product*

Digunakan untuk menyimpan seluruh data *product* pada Sistem Informasi Eksekutif Penjualan Kerajinan Tas Rotan Pada Balinese Classic Menggunakan Framework Laravel. Berikut struktur *table product* pada Tabel 4.9:

Tabel 4.9 Struktur Tabel *Product*

No	Nama Atribut	Tipe Data	Ukuran	Key Constraint
1	<i>id</i>	<i>int</i>	10	<i>Primary Key</i>
2	<i>category_id</i>	<i>int</i>	10	<i>Foreign Key</i>
3	nama_produk	<i>varchar</i>	20	

No	Nama Atribut	Tipe Data	Ukuran	Key Constraint
4	kode_produk	varchar	20	
5	keterangan	text	50	
6	satuan	varchar	10	
7	harga_dasar	int	11	
8	harga_jual	int	11	
9	gambar	varchar	50	
10	created_at	timestamp		
11	updated_at	timestamp		

f. Struktur Tabel Stock

Digunakan untuk menyimpan seluruh data stok pada Sistem Informasi Eksekutif Penjualan Kerajinan Tas Rotan Pada Balinese Classic Menggunakan Framework Laravel. Berikut struktur *table* stok pada Tabel 4.10:

Tabel 4.10 Struktur Tabel Stocks

No	Nama Atribut	Tipe Data	Ukuran	Key Constraint
1	<i>id</i>	int	10	Primary Key
2	<i>product_id</i>	int	10	Foreign Key
3	stok	int	11	

g. Struktur Tabel Penjualan

Digunakan untuk menyimpan seluruh data penjualan pada Sistem Informasi Eksekutif Penjualan Kerajinan Tas Rotan Pada Balinese Classic Menggunakan Framework Laravel. Berikut struktur *table* penjualan pada Tabel 4.11:

Tabel 4.11 Struktur Tabel Penjualan

No	Nama Atribut	Tipe Data	Ukuran	Key Constraint
1	<i>id</i>	int	10	Primary Key
2	tanggal_transaksi	date		
3	<i>customer_id</i>	int	10	Foreign Key
4	<i>invoice_number</i>	varchar	20	
5	total_harga	int	11	
6	<i>shipping</i>	varchar	11	
7	profit	int	11	

No	Nama Atribut	Tipe Data	Ukuran	Key Constraint
8	status	enum		
9	created_by	bigint	20	Foreign Key
10	Updated_by	bigint	20	
11	created_at	timestamp		
12	updated_at	timestamp		

h. Struktur Tabel Penjualan\_product

Digunakan untuk menyimpan seluruh data detail penjualan pada Sistem Informasi Eksekutif Penjualan Kerajinan Tas Rotan Pada Balinese Classic Menggunakan Framework Laravel. Berikut struktur *table* penjualan\_product pada Tabel 4.12:

Tabel 4.12 Struktur Tabel Penjualan\_products

No	Nama Atribut	Tipe Data	Ukuran	Key Constraint
1	penjualan_id	int	10	Primary Key
2	product_id	int	10	Primary Key
3	qty	int	11	
4	created_at	timestamp		
5	updated_at	timestamp		

i. Struktur Tabel Pembelian

Digunakan untuk menyimpan seluruh data pembelian pada Sistem Informasi Eksekutif Penjualan Kerajinan Tas Rotan Pada Balinese Classic Menggunakan Framework Laravel. Berikut struktur *table* user pada Tabel 4.13:

Tabel 4.13 Struktur Tabel Pembelian

No	Nama Atribut	Tipe Data	Ukuran	Key Constraint
1	<i>id</i>	int	10	PK
2	tanggal_transaksi	date		
3	supplier_id	int	10	Foreign Key
4	invoice_number	varchar	20	
5	total_harga	int	11	
6	status	enum		
7	created_by	bigint	20	Foreign Key
8	Updated_by	bigint	20	

9	<i>created_at</i>	timestamp		
10	<i>updated_at</i>	timestamp		

j. Struktur Tabel *Pembelian\_product*

Digunakan untuk menyimpan seluruh data detail pembelian pada Sistem Informasi Eksekutif Penjualan Kerajinan Tas Rotan Pada Balinese Classic Menggunakan Framework Laravel. Berikut struktur *table* *pembelian\_product* pada Tabel 4.14:

Tabel 4.14 Struktur Tabel *Pembelian\_product*

No	Nama Atribut	Tipe Data	Ukuran	Key Constraint
1	<i>pembelian_id</i>	<i>int</i>	10	PK
2	<i>product_id</i>	<i>int</i>	10	PK
3	<i>qty</i>	<i>int</i>	11	
4	<i>created_at</i>	timestamp		
5	<i>updated_at</i>	timestamp		

#### 4.4 Perancangan Antarmuka

Perancangan antarmuka akan menjelaskan rancangan antarmuka dari sistem yang dirancang. Perancangan antarmuka ini akan memberikan gambaran tentang bagaimana bentuk dari tampilan sistem yang selanjutnya akan diimplementasikan menjadi Sistem Informasi Eksekutif Penjualan Kerajinan Tas Rotan Pada Balinese Classic Menggunakan Framework Laravel.

##### 4.4.1 Desain Antarmuka Admin

Bagian ini akan menjelaskan tentang rancangan antarmuka untuk *roles* admin. Rancangan antarmuka tersebut antaranya sebagai berikut:

1. Desain Antarmuka *Login Admin*

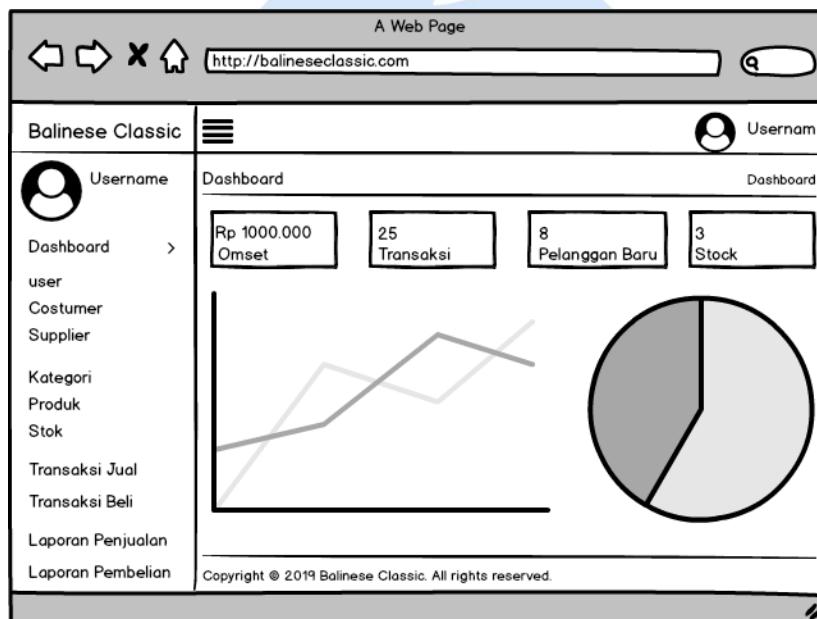
Halaman *login* adalah halaman pertama di mana admin diminta untuk memasukkan email dan *password* agar bisa masuk ke dalam sistem. Pada tahap ini sistem akan menentukan tampilan selanjutnya pada halaman *dashboard* berdasarkan *roles* admin tersebut. Desain antarmuka *login* dapat dilihat pada Gambar 4.15 berikut:



Gambar 4.15 Antarmuka *Login* – Admin

## 2. Desain Antarmuka *Dashboard* Admin

Halaman *Dashboard* Admin adalah halaman setelah admin memasukkan data *login* yang memiliki *roles* admin. Halaman ini menampilkan jumlah omset, jumlah transaksi, jumlah pelanggan baru, dan jumlah *stock* serta terdapat menu *sidebar* di sebelah kiri seperti, menu *dashboard*, menu *users*, menu *customer*, menu *supplier*, menu kategori, menu produk, menu *stock*, menu transaksi jual, menu transaksi beli, menu laporan penjualan, menu laporan pembelian. Desain antarmuka halaman *dashboard* admin dapat dilihat pada Gambar 4.16 berikut:



Gambar 4.16 Antarmuka *Dashboard* – Admin

### 3. Desain Antarmuka Kelola Data Users

Halaman Kelola Data *Users* menampilkan sidebar menu dan daftar data *users* yang telah disimpan di dalam sistem. Desain antarmuka kelola data *users* dapat dilihat pada gambar berikut:

Gambar 4.17 Antarmuka Data *Users* – Admin

### 4. Desain Form Data *Users*

*Form* kelola data *users* adalah *form* yang digunakan untuk memasukkan data *users* dan mengedit data *user*. Desain antarmuka *form* kelola data *users* dapat dilihat pada gambar berikut:

Gambar 4.18 Antarmuka *Form Users* – Admin

## 5. Desain Show Detail Data *Users*

Show detail data *user* adalah tampilan yang berisikan data lengkap dari *user* tersebut. Desain antarmuka *show detail data user* dapat dilihat pada gambar berikut:

The wireframe shows a form titled "Info Data User". It contains several input fields and one placeholder image for an "Avatar". The fields are labeled as follows:

- Tanggal Daftar
- Role
- Phone
- Status
- Address
- Avatar (Placeholder image with a large X)
- Nama
- Username
- Email

Gambar 4.19 Antarmuka detail *users* – Admin

## 6. Desain Antarmuka Kelola Data *Customers*

Halaman Kelola data *customer* menampilkan sidebar menu dan daftar data *customer* yang telah disimpan di dalam sistem. Desain antarmuka kelola data *customer* dapat dilihat pada gambar berikut:

The screenshot shows a web application interface for managing customer data. The top navigation bar includes standard browser controls and a URL bar showing <http://balineseclassic.com>. The header features the brand name "Balinese Classic" and a user profile icon with the text "Username".

The left sidebar contains a vertical navigation menu with the following items:

- Dashboard
- user
- Costumer >
- Supplier
- Kategori Produk
- Stok
- Transaksi Jual
- Transaksi Beli
- Laporan Penjualan
- Laporan Pembelian

The main content area is titled "Customer List" and displays a table with the following columns:

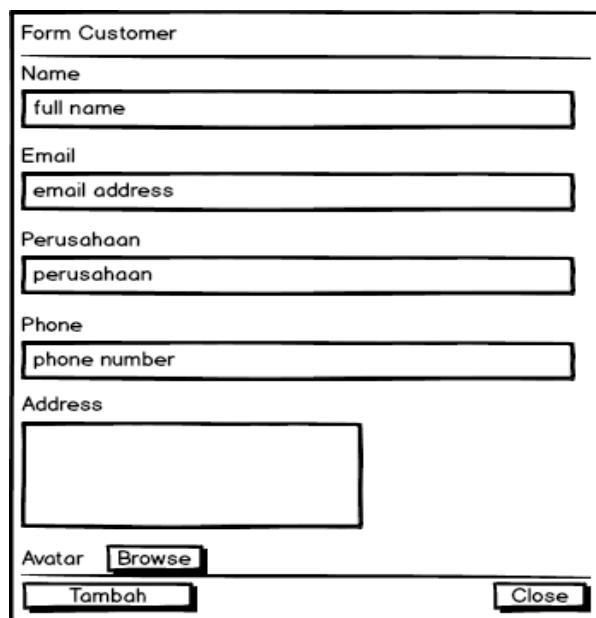
No^	Name	Email	Perusahaan	Telepon	Terdaftar	Action
1						
2						
3						
4						
5						
6						
7						
8						
9						
10						

At the bottom of the page, there is a copyright notice: "Copyright © 2019 Balinese Classic. All rights reserved."

Gambar 4.20 Antarmuka Data *Customers* – Admin

7. Desain *Form Data Customers*

*Form* kelola data *customer* adalah *form* yang digunakan untuk memasukkan data *customer* dan mengedit data *customer*. Desain antarmuka *form* kelola data *customer* dapat dilihat pada gambar berikut:



Form Customer

Name  
full name

Email  
email address

Perusahaan  
perusahaan

Phone  
phone number

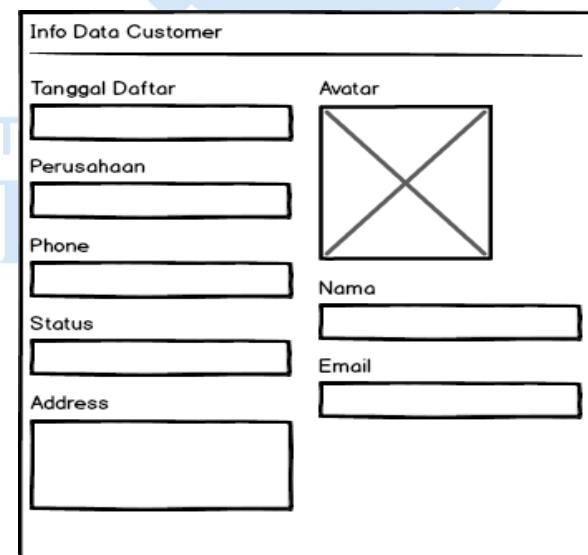
Address

Avatar

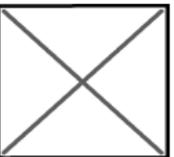
Gambar 4.21 Antarmuka *Form Customers – Admin*

8. Desain *Show Detail Data Customers*

*Show detail data user* adalah tampilan yang berisikan data lengkap dari *customer* tersebut. Desain antarmuka *show detail data customer* dapat dilihat pada gambar berikut:



Info Data Customer

Tanggal Daftar	Avatar
Perusahaan	
Phone	Nama
Status	Email
Address	

Gambar 4.22 Antarmuka *Detail Customers – Admin*

## 9. Desain Antarmuka Kelola Data *Suppliers*

Halaman Kelola data *supplier* menampilkan sidebar menu dan daftar data *supplier* yang telah disimpan didalam sistem. Desain antarmuka kelola data *supplier* dapat dilihat pada gambar berikut:

Gambar 4.23 Antarmuka Data *Suppliers* – Admin

## 10. Desain Form Data *Suppliers*

Form kelola data *supplier* adalah form yang digunakan untuk memasukkan data *supplier* dan mengedit data *supplier*. Desain antarmuka form kelola data *supplier* dapat dilihat pada gambar berikut:

Gambar 4.24 Antarmuka Form *Suppliers* – Admin

## 11. Desain Show Detail Data Suppliers

Show detail data *supplier* adalah tampilan yang berisikan data lengkap dari *supplier* tersebut. Desain antarmuka *show detail data supplier* dapat dilihat pada gambar berikut:

The wireframe shows a form titled "Info Data Supplier". It includes fields for "Tanggal Daftar" (Date Registered), "Perusahaan" (Company), "Phone", "Status", and "Address". On the right side, there is an "Avatar" placeholder with a large "X" and fields for "Nama" (Name) and "Email".

Gambar 4.25 Antarmuka Detail Suppliers – Admin

## 12. Desain Antarmuka Kelola Data Categories

Halaman Kelola data *category* menampilkan sidebar menu dan daftar data *category* yang telah disimpan didalam sistem. Desain antarmuka kelola data *category* dapat dilihat pada gambar berikut:

The screenshot shows a web browser window with the URL "http://balineseclassic.com". The page title is "Balinese Classic". The left sidebar has links for "Dashboard", "user", "Costumer", "Supplier", "Kategori", "Produk", "Stok", "Transaksi Jual", "Transaksi Beli", "Laporan Penjualan", and "Laporan Pembelian". The main content area shows a "Kategori List" table with columns "No^", "Nama Kategori", "Gambar", and "Action". A "Tambah" button is located at the top right of the table. The footer contains the copyright notice "Copyright © 2019 Balinese Classic. All rights reserved."

Gambar 4.26 Antarmuka Data Categories – Admin

13. Desain Form Data Categories

Form kelola data category adalah form yang digunakan untuk memasukkan data category dan mengedit data category. Desain antarmuka form kelola data category dapat dilihat pada gambar berikut:

Gambar 4.27 Antarmuka Form Categories – Admin

14. Desain Antarmuka Kelola Data Products

Halaman Kelola data product menampilkan sidebar menu dan daftar data product yang telah disimpan di dalam sistem. Desain antarmuka kelola data product dapat dilihat pada gambar berikut:

Gambar 4.28 Antarmuka Data Products – Admin

15. Desain *Form Data Products*

*Form* kelola data *products* adalah *form* yang digunakan untuk memasukkan data *products* dan mengedit data *products*. Desain antarmuka *form* kelola data *products* dapat dilihat pada gambar berikut:

Form Product

Kode Produk  
kode produk

Nama Produk  
nama produk

Kategori  
kategori

Keterangan

Satuan  
satuan

Harga Dasar  
harga dasar

Harga Jual  
harga jual

gambar

Gambar 4.29 Antarmuka *Form Products* – Admin

16. Desain *Show Detail Data Products*

*Show detail data products* adalah tampilan yang berisikan data lengkap dari *products* tersebut. Desain antarmuka *show detail data products* dapat dilihat pada gambar berikut:

**Info Data Product**

Kode Produk	Gambar
<input type="text"/>	
Nama	<input type="text"/>
Kategori	<input type="text"/>
Satuan	<input type="text"/>
Keterangan	<input type="text"/>
	Harga Dasar
	<input type="text"/>
	Harga Jual
	<input type="text"/>
	Stock
	<input type="text"/>

Gambar 4.30 Antarmuka Detail *Products* – Admin

#### 17. Desain Antarmuka Kelola Data Stock

Halaman Kelola data stock menampilkan sidebar menu dan daftar data stock yang telah disimpan didalam sistem. Desain antarmuka kelola data stock dapat dilihat pada gambar berikut:

A Web Page

The screenshot shows a web browser window with the following details:

- Header:** A Web Page, back, forward, search bar (http://balineseclassic.com), and a magnifying glass icon.
- Top Bar:** Balinese Classic logo, Username input field, and a three-line menu icon.
- Left Sidebar (Username):**
  - Dashboard
  - user
  - Costumer
  - Supplier
  - Kategori
  - Produk
  - Stok >
  - Transaksi Jual
  - Transaksi Beli
  - Laporan Penjualan
  - Laporan Pembelian
- Current Page:** Dashboard > Stock
- Content Area:**
  - Stock:** A table with columns: No^, Nama, Stock, Action. The 'Action' column contains a pencil icon.
  - Buttons:** Home, Stock, and a 'Tambah' button.
- Footer:** Copyright © 2019 Balinese Classic. All rights reserved.

Gambar 4.31 Antarmuka Data Stock – Admin

## 18. Desain Form Data Stock

*Form* kelola data stock adalah *form* yang digunakan untuk memasukkan data stock dan mengedit data stock. Desain antarmuka *form* kelola data stock dapat dilihat pada gambar berikut:

Form Stock

---

Nama Produk

Jumlah Stock



---

Edit
Close

Gambar 4.32 Antarmuka *Form Stock – Admin*

## 19. Desain Antarmuka Kelola Data Penjualan

Halaman Kelola data penjualan menampilkan sidebar menu dan daftar data penjualan yang telah disimpan di dalam sistem. Desain antarmuka kelola data penjualan dapat dilihat pada gambar berikut:

A Web Page

<http://balineseclassic.com>

Balinese Classic		 Username					
							
 Username  <a href="#">Dashboard</a> <a href="#">user</a> <a href="#">Costumer</a> <a href="#">Supplier</a>  <a href="#">Kategori</a> <a href="#">Produk</a> <a href="#">Stok</a>  <a href="#">Transaksi Jual &gt;</a> <a href="#">Transaksi Beli</a>  <a href="#">Laporan Penjualan</a> <a href="#">Laporan Pembelian</a>	Dashboard						
	<a href="#">Home</a> > Penjualan						
	Penjualan List						<a href="#">Tambah</a>
	No^v	Invoice Number^v	Tanggal Pesan^v	Customer^v	Product	Status	Action
							 

Copyright © 2019 Balinese Classic. All rights reserved.

Gambar 4. 33 Antarmuka Data Penjualan – Admin

## 20. Desain Antarmuka Transaksi Penjualan

Pada halaman transaksi penjualan ini berisi *form* yang berfungsi untuk menampung data transaksi penjualan. Desain antarmuka transaksi penjualan dapat dilihat pada gambar berikut:

The screenshot shows a web browser window titled 'A Web Page' with the URL 'http://balineseclassic.com'. The page has a header with a user icon and 'Username'. On the left is a sidebar menu with items like 'Dashboard', 'user', 'Costumer', 'Supplier', 'Kategori', 'Produk', 'Stok', 'Transaksi Jual' (with a dropdown arrow), 'Transaksi Beli', 'Laporan Penjualan', and 'Laporan Pembelian'. The main content area is titled 'Dashboard' and shows a breadcrumb path: 'Home > Penjualan > Transaksi'. Below this is a 'Form Transaksi' section containing fields for 'Date', 'Customer', 'Shipping', 'Status', 'Produk', 'QTY', and a 'Submit' button. There is also a 'Tambah Produk' (Add Product) button and a delete icon (X). At the bottom of the content area is a copyright notice: 'Copyright © 2019 Balinese Classic. All rights reserved.'

Gambar 4.34 Antarmuka Transaksi Jual – Admin

## 21. Desain Antarmuka Kelola Data Pembelian

Halaman Kelola data pembelian menampilkan *sidebar* menu dan daftar data pembelian yang telah disimpan di dalam sistem. Desain antarmuka kelola data pembelian dapat dilihat pada gambar berikut:

The screenshot shows a web browser window titled 'A Web Page' with the URL 'http://balineseclassic.com'. The page has a header with a user icon and 'Username'. On the left is a sidebar menu with items like 'Dashboard', 'user', 'Costumer', 'Supplier', 'Kategori', 'Produk', 'Stok', 'Transaksi Jual', 'Transaksi Beli' (with a dropdown arrow), 'Laporan Penjualan', and 'Laporan Pembelian'. The main content area is titled 'Dashboard' and shows a breadcrumb path: 'Home > Pembelian'. Below this is a 'Pembelian List' section with a 'Tambah' (Add) button. A table displays columns for 'No^v', 'Invoice Number^v', 'Tanggal Pesan^v', 'Customer^v', 'Product', 'Status', and 'Action'. The table has several rows of data. At the bottom of the content area is a copyright notice: 'Copyright © 2019 Balinese Classic. All rights reserved.'

Gambar 4.35 Antarmuka Data Pembelian – Admin

22. Desain Antarmuka Transaksi Pembelian

Pada halaman transaksi pembelian ini berisi *form* yang berfungsi untuk menampung data transaksi pembelian. Desain antarmuka transaksi pembelian dapat dilihat pada gambar berikut:

Gambar 4.36 Antarmuka Transaksi Beli – Admin

23. Desain Antarmuka *Invoice* Penjualan

Halaman ini menampilkan nota dari transaksi penjualan yang sudah ada di dalam sistem. Desain antarmuka *invoice* penjualan dapat dilihat pada gambar berikut:

Product	Qty	Price	Total Price
Tas ate	40	1000	40000
Tas ate	40	1000	40000

Sub\_total = RP 80.000  
 Shipping = Rp 20.000  
 Ppn(10%) = Rp 8.000  
 Total = Rp 108.000

Gambar 4.37 Antarmuka *Invoice* Penjualan – Admin

#### 24. Desain Antarmuka *Invoice* Pembelian

Halaman ini menampilkan nota dari transaksi pembelian yang sudah ada di dalam sistem. Desain antarmuka *invoice* pembelian dapat dilihat pada gambar berikut:

Balinese Classic

From  
**Balinese Classic**  
Jalan Kebo Iwa Perum  
Swamandala XIII no 3  
Phone: (+62) 0895342574617

To  
**Supplier**  
Jalan Kambjo No.5, Jawa Barat  
Phone: (+62) 0895342574617  
Email: supplier@amail.com

Invoice #20198089648

Product	Qty	Price	Total Price
Tas ate	40	1000	40000
Tas ate	40	1000	40000

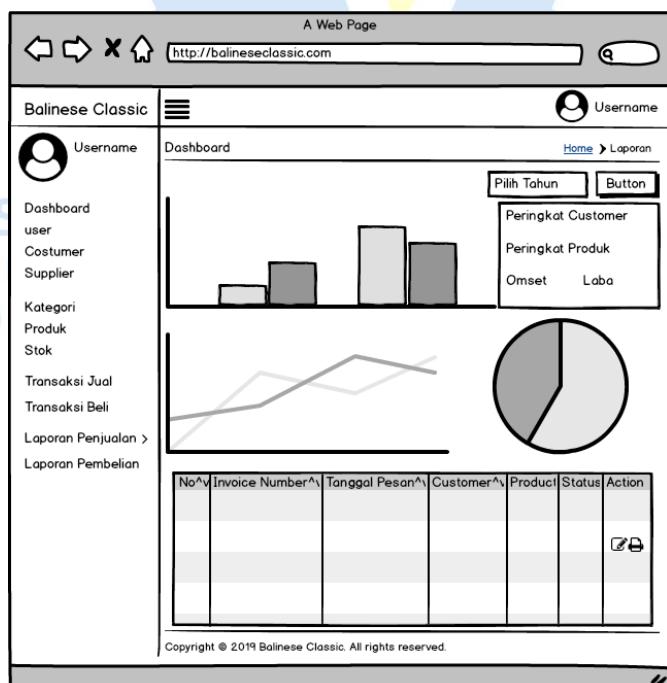
Sub\_total = RP 80.000  
Shipping = Rp 20.000  
Ppn(10%) = Rp 8.000  
Total = Rp 108.000

Print

Gambar 4.38 Antarmuka *Invoice* Pembelian – Admin

#### 25. Desain Antarmuka Laporan Penjualan

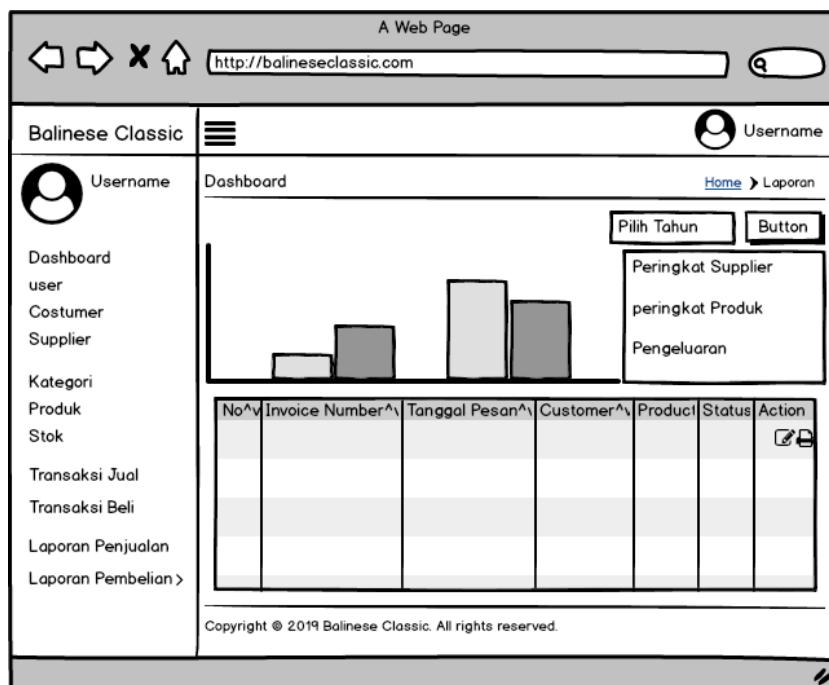
Halaman ini menampilkan data-data penjualan berupa grafik dan *grid* di mana *user* diminta untuk memasukkan tahun untuk memfilter data tersebut. Desain antarmuka laporan penjualan dapat dilihat pada gambar berikut:



Gambar 4.39 Antarmuka Laporan Penjualan – Admin

## 26. Desain Antarmuka Laporan Pembelian

Halaman ini menampilkan data-data pembelian berupa grafik dan *grid* di mana *user* diminta untuk memasukkan tahun untuk memfilter data tersebut. Desain antarmuka laporan pembelian dapat dilihat pada gambar berikut:



Gambar 4.40 Antarmuka Laporan Pembelian – Admin

### 4.4.2 Desain Antarmuka Kasir

Bagian ini akan menjelaskan tentang rancangan antarmuka untuk *roles* Kasir. Rancangan antarmuka tersebut antaranya sebagai berikut:

#### 1. Desain Antarmuka *Login* Kasir

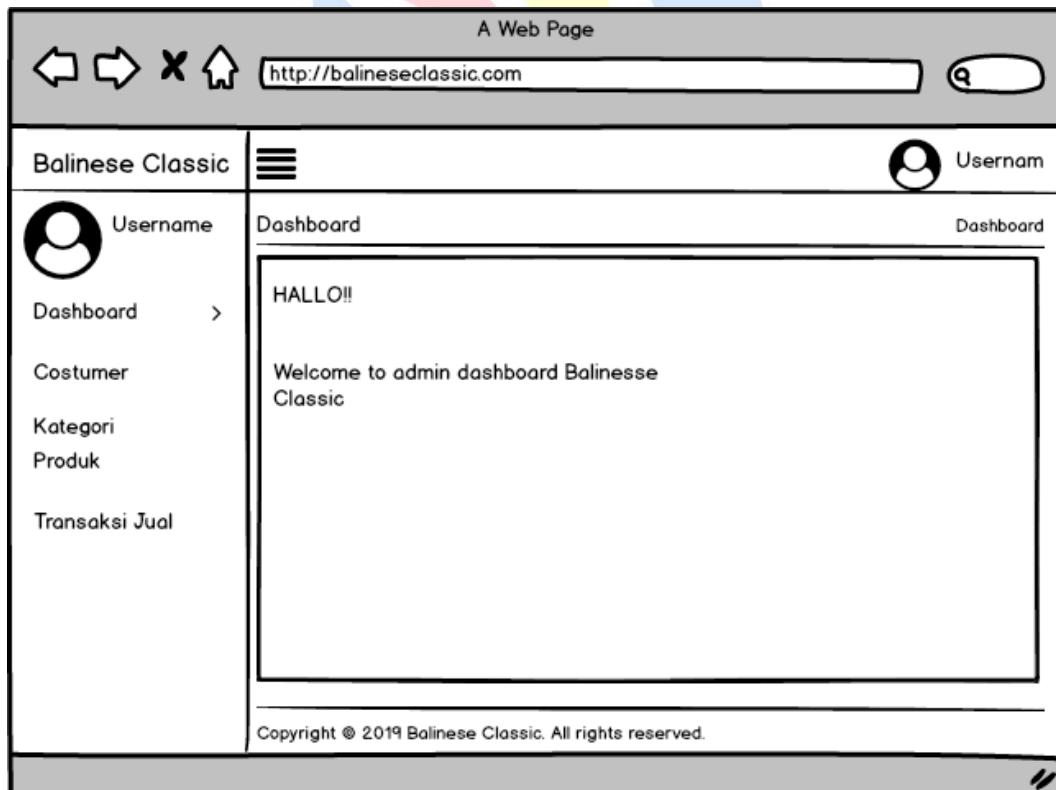
Halaman *login* adalah halaman pertama di mana kasir diminta untuk memasukkan email dan *password* agar bisa masuk ke dalam sistem. Pada tahap ini sistem akan menentukan tampilan selanjutnya pada halaman *dashboard* berdasarkan *roles* kasir tersebut. Desain antarmuka *login* dapat dilihat pada gambar berikut:



Gambar 4.41 Antarmuka *Login* – Kasir

## 2. Desain Antarmuka *Dashboard* Kasir

Halaman *Dashboard* Kasir merupakan halaman di mana kasir setelah berhasil melakukan *login*. Menu yang terdapat pada *dashboard* kasir meliputi, *Dashboard*, *Customer*, Kategori, Produk, dan Transaksi Jual.



Gambar 4.42 Antarmuka *Dashboard* – Kasir

### 3. Desain Antarmuka *List Customer*

Halaman ini menampilkan data-data *customer* yang ada di *database*, juga terdapat tombol untuk menuju *form customer* dan detail *customer*. Berikut dapat dilihat pada gambar di bawah ini:

A Web Page

<http://balineseclassic.com>

Balinese Classic

Username

Dashboard

Costumer >

Kategori Produk

Transaksi Jual

Dashboard

Customer List

Home > Customer

Tambah

No^	Name	Email	Perusahaan	Telepon	Terdaftar	Action
1						
2						
3						
4						
5						
6						
7						
8						
9						
10						

Copyright © 2019 Balinese Classic. All rights reserved.

Gambar 4.43 Antarmuka *List Customer* – Kasir

### 4. Desain *Form Customer*

*Form* kelola data *customer* adalah *form* yang digunakan untuk memasukkan data *customer* dan mengedit data *customer*. Berikut dapat dilihat pada gambar di bawah ini:



**Form Customer**

---

Name

Email

Perusahaan

Phone

Address

Avatar

Gambar 4.44 Antarmuka *Form Customer* – Kasir

### 5. Desain Detail *Customer*

*Detail customer* adalah *pop up* yang menampilkan data *customer* secara detail/lengkap. Berikut dapat dilihat pada gambar di bawah ini:

**Info Data Customer**

---

Tanggal Daftar <input type="text"/>	Avatar 
Perusahaan <input type="text"/>	Nama <input type="text"/>
Phone <input type="text"/>	Email <input type="text"/>
Status <input type="text"/>	
Address <input type="text"/>	

Gambar 4.45 Antarmuka *Detail Customer* – Kasir

## 6. Desain Antarmuka *List Kategori*

Halaman ini menampilkan data-data kategori yang ada di *database*, juga terdapat tombol untuk menuju *form* kategori dan detail kategori. Berikut dapat dilihat pada gambar di bawah ini:

NO^	Nama Kategori	Gambar	Action

Gambar 4.46 Antarmuka *List Kategori* – Kasir

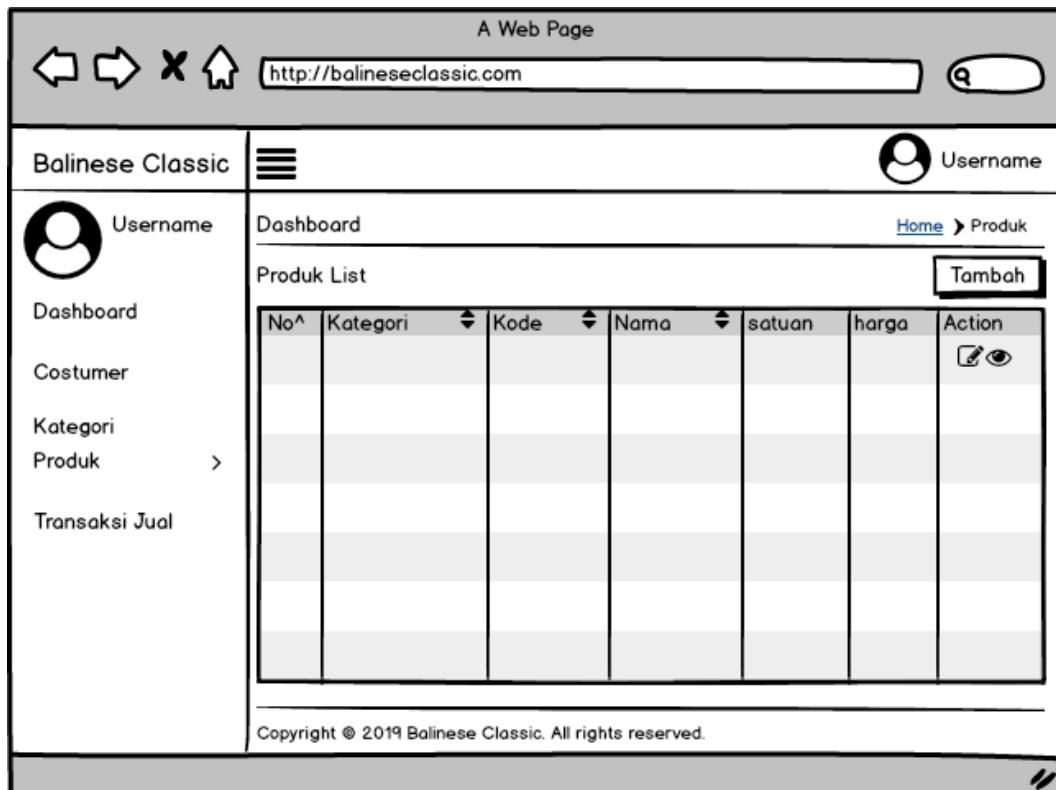
## 7. Desain *Form Kategori*

*Form* kelola data kategori adalah *form* yang digunakan untuk memasukkan data kategori dan mengedit data kategori. Berikut dapat dilihat pada gambar di bawah ini:

Gambar 4.47 Antarmuka *Form Kategori* – Kasir

#### 8. Desain Antarmuka *List Product*

Halaman ini menampilkan data-data *product* yang ada di *database*, juga terdapat tombol untuk menuju *form product* dan detail *product*. Berikut dapat dilihat pada gambar di bawah ini:



Gambar 4.48 Antarmuka *List Product* – Kasir

## 9. Desain Form Product

*Form* kelola data *product* adalah *form* yang digunakan untuk memasukkan data *product* dan mengedit data *product*. Berikut dapat dilihat pada gambar di bawah ini:

Form Product

Kode Produk  
kode produk

Nama Produk  
nama produk

Kategori  
kategori

Keterangan

Satuan  
satuan

Harga Dasar  
harga dasar

Harga Jual  
harga jual

gambar

This diagram illustrates a user interface for adding a product. It features a vertical list of input fields and buttons. At the top is a title 'Form Product'. Below it are fields for 'Kode Produk' (with placeholder 'kode produk'), 'Nama Produk' (placeholder 'nama produk'), 'Kategori' (placeholder 'kategori'), and 'Keterangan' (an empty text input field). The next group of fields includes 'Satuan' (placeholder 'satuan') and 'Harga Dasar' (placeholder 'harga dasar'). Following these are 'Harga Jual' (placeholder 'harga jual') and a file upload section with a browse button. At the bottom are two buttons: 'Tambah Product' and 'Close'.

Gambar 4.49 Antarmuka *Form Product* – Kasir

#### 10. Desain Detail *Product*

Detail *product* adalah *pop up* yang menampilkan data *product* secara detail/lengkap. Berikut dapat dilihat pada gambar di bawah ini:

Info Data Product

Kode Produk	Gambar
<input type="text"/>	
Nama	<input type="text"/>
Kategori	<input type="text"/>
Satuan	<input type="text"/>
Keterangan	<input type="text"/>
	Harga Dasar
	<input type="text"/>
	Harga Jual
	<input type="text"/>
	Stock
	<input type="text"/>

This diagram shows a detailed view of a product's information. It is titled 'Info Data Product'. The left side lists product details: Kode Produk, Nama, Kategori, Satuan, and Keterangan, each with an associated text input field. The right side displays price-related fields: Harga Dasar, Harga Jual, and Stock, each with its own text input field. A placeholder for the product image is shown with a large 'X' inside.

Gambar 4.50 Antarmuka Detail *Product* – Kasir

## 11. Desain Antarmuka *List* Penjualan

Halaman ini menampilkan data-data penjualan yang ada di *database*, juga terdapat tombol untuk menuju *form* penjualan dan *invoice* penjualan. Berikut dapat dilihat pada gambar di bawah ini:

No	Invoice Number	Tanggal Pesan	Customer	Product	Status	Action
1	INV-001	2023-01-01	Customer 1	Product A	Pending	<input checked="" type="checkbox"/>
2	INV-002	2023-01-02	Customer 2	Product B	Pending	<input checked="" type="checkbox"/>
3	INV-003	2023-01-03	Customer 3	Product C	Pending	<input checked="" type="checkbox"/>
4	INV-004	2023-01-04	Customer 4	Product D	Pending	<input checked="" type="checkbox"/>
5	INV-005	2023-01-05	Customer 5	Product E	Pending	<input checked="" type="checkbox"/>
6	INV-006	2023-01-06	Customer 6	Product F	Pending	<input checked="" type="checkbox"/>
7	INV-007	2023-01-07	Customer 7	Product G	Pending	<input checked="" type="checkbox"/>
8	INV-008	2023-01-08	Customer 8	Product H	Pending	<input checked="" type="checkbox"/>
9	INV-009	2023-01-09	Customer 9	Product I	Pending	<input checked="" type="checkbox"/>
10	INV-010	2023-01-10	Customer 10	Product J	Pending	<input checked="" type="checkbox"/>

Gambar 4.51 Antarmuka *List* Penjualan – Kasir

## 12. Desain Transaksi Penjualan

Pada halaman transaksi penjualan ini berisi *form* yang berfungsi untuk menampung data transaksi penjualan. Desain antarmuka transaksi penjualan dapat dilihat pada gambar berikut:

Form Transaksi

Date	<input type="text"/>	
Customer	<input type="text"/>	
Shipping	<input type="text"/>	
Status	<input type="text"/>	
Produk	<input type="text"/>	QTY <input type="text"/>
<input type="button" value="Tambah Produk"/>		<input type="button" value="Submit"/>

Gambar 4.52 Antarmuka *Form* Penjualan – Kasir

### 13. Desain *Invoice* Penjualan

Halaman ini menampilkan *invoice* dari transaksi penjualan yang sudah ada di dalam sistem. Desain antarmuka *invoice* penjualan dapat dilihat pada gambar berikut:

 Balinese Classic			
From <b>Balinese Classic</b> Jalan Kebo Iwa Perum Swamandala XIII no 3 Phone: (+62) 0895342574617	To <b>Customer</b> Jalan Kamboja No.5, Jawa Barat Phone: (+62) 0895342574617 Email: customer@amail.com	Invoice #20198089648	
Product	Qty	Price	Total Price
Tas ate	40	1000	40000
Tas ate	40	1000	40000

Sub\_total = RP 80.000  
 Shipping = Rp 20.000  
 Ppn(10%) = Rp 8.000  
 Total = Rp 108.000

**Print**

Gambar 4.53 Antarmuka *Invoice Penjualan – Kasir*

#### **4.4.3 Desain Antarmuka Gudang**

Bagian ini akan menjelaskan tentang rancangan antarmuka untuk *roles* Gudang. Rancangan antarmuka tersebut antaranya sebagai berikut:

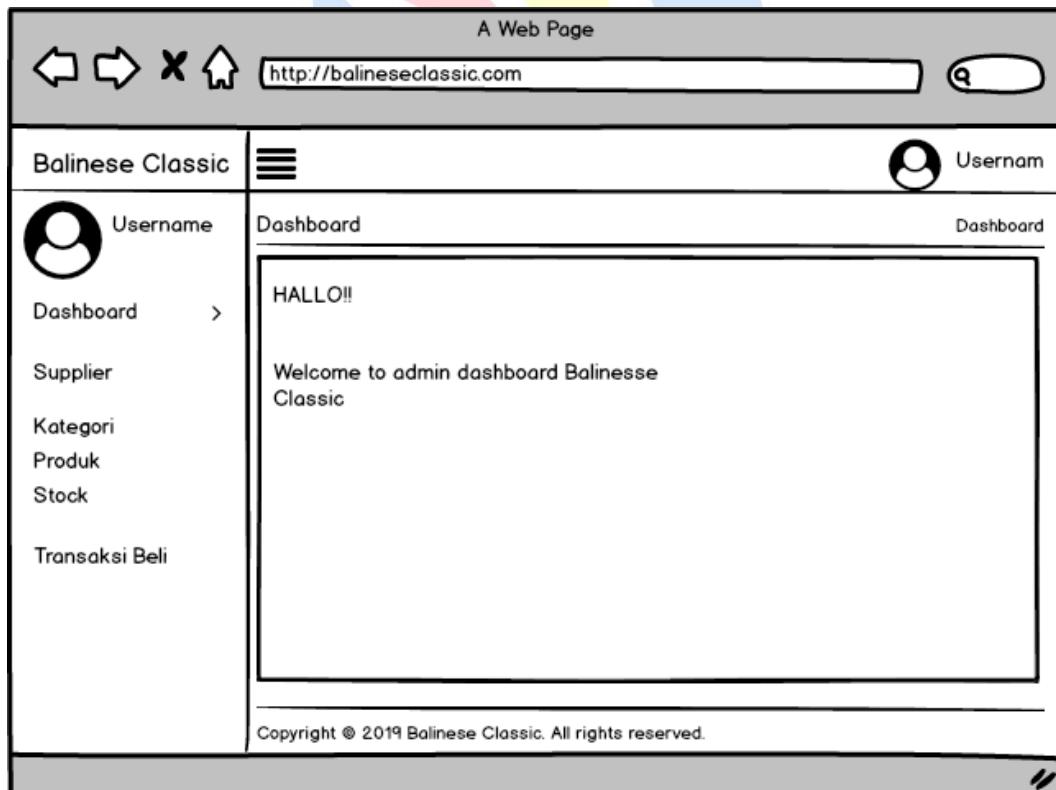
Halaman *login* adalah halaman pertama di mana kasir diminta untuk memasukkan email dan *password* agar bisa masuk ke dalam sistem. Pada tahap ini sistem akan menentukan tampilan selanjutnya pada halaman *dashboard* berdasarkan *roles* gudang tersebut. Desain antarmuka *login* dapat dilihat pada gambar berikut:



Gambar 4.54 Antarmuka *Login* – Gudang

## 2. Desain Antarmuka *Dashboard* Gudang

Halaman *Dashboard* Gudang merupakan halaman di mana gudang setelah berhasil melakukan *login*. Menu yang terdapat pada *dashboard* gudang meliputi, *Dashboard*, *Supplier*, Kategori, Produk, Stock, dan Transaksi Beli.



Gambar 4.55 Antarmuka *Dashboard Kasir* – Gudang

### 3. Desain Antarmuka *List Supplier*

Halaman ini menampilkan data-data *supplier* yang ada di *database*, juga terdapat tombol untuk menuju *form supplier* dan detail *supplier*. Berikut dapat dilihat pada gambar di bawah ini:

Gambar 4.56 Antarmuka *List Supplier* – Gudang

### 4. Desain *Form Supplier*

*Form* kelola data *supplier* adalah *form* yang digunakan untuk memasukkan data *supplier* dan mengedit data *supplier*. Berikut dapat dilihat pada gambar di bawah ini:

Gambar 4.57 Antarmuka *Form Supplier* – Gudang

## 5. Desain Detail *Supplier*

Detail *supplier* adalah *pop up* yang menampilkan data *supplier* secara detail/lengkap. Berikut dapat dilihat pada gambar di bawah ini:

Gambar 4.58 Antarmuka Detail *Supplier* – Gudang

## 6. Desain Antarmuka *List Kategori*

Halaman ini menampilkan data-data kategori yang ada di *database*, juga terdapat tombol untuk menuju *form* kategori dan detail kategori. Berikut dapat dilihat pada gambar di bawah ini:

Gambar 4.59 Antarmuka *List Kategori* – Gudang

## 7. Desain Form Kategori

*Form* kelola data kategori adalah *form* yang digunakan untuk memasukkan data kategori dan mengedit data kategori. Berikut dapat dilihat pada gambar di bawah ini:

The screenshot shows a modal window titled "Form Category". Inside, there is a text input field labeled "Name" containing the text "full name". Below this is an "Avatar" section with a "Browse" button. At the bottom of the window are two buttons: "Tambah" on the left and "Close" on the right.

Gambar 4.60 Antarmuka *Form* Kategori – Gudang

## 8. Desain Antarmuka *List Product*

Halaman ini menampilkan data-data *product* yang ada di *database*, juga terdapat tombol untuk menuju *form product* dan detail *product*. Berikut dapat dilihat pada gambar di bawah ini:

The screenshot shows a web page titled "A Web Page" with the URL "http://balineseclassic.com". The header includes standard browser controls and a search bar. On the left is a sidebar with a user profile icon and the text "Balinese Classic". The sidebar contains links: "Dashboard", "Supplier", "Kategori", "Produk", "Stock", and "Transaksi Beli". To the right of the sidebar is a main content area. At the top of this area is a navigation bar with "Dashboard" and "Home > Produk". Below this is a table titled "Produk List" with columns: "No^", "Kategori", "Kode", "Nama", "satuan", "harga", and "Action". A "Tambah" button is located at the top right of the table. At the bottom of the main content area is a copyright notice: "Copyright © 2019 Balinese Classic. All rights reserved."

Gambar 4.61 Antarmuka *List Product* – Gudang

## 9. Desain *Form Product*

*Form* kelola data *product* adalah *form* yang digunakan untuk memasukkan data *product* dan mengedit data *product*. Berikut dapat dilihat pada gambar di bawah ini:

Form Product

Kode Produk  
kode produk

Nama Produk  
nama produk

Kategori  
kategori

Keterangan

Satuan  
satuan

Harga Dasar  
harga dasar

Harga Jual  
harga jual

gambar

Gambar 4.62 Antarmuka *Form Product* – Gudang

#### 10. Desain Detail *Product*

Detail *product* adalah *pop up* yang menampilkan data *product* secara detail/lengkap. Berikut dapat dilihat pada gambar di bawah ini:

Info Data Product	
Kode Produk	Gambar
<input type="text"/>	
Nama	<input type="text"/>
Kategori	<input type="text"/>
Satuan	<input type="text"/>
Keterangan	<input type="text"/>
	Harga Dasar
	<input type="text"/>
	Harga Jual
	<input type="text"/>
	Stock
	<input type="text"/>

Gambar 4.63 Antarmuka Detail *Product* – Gudang

## 11. Desain Antarmuka *List Stock*

Halaman ini menampilkan data-data *product* dan jumlah *stock* yang ada di *database*, juga terdapat tombol untuk menuju *form stock*. Berikut dapat dilihat pada gambar di bawah ini:

A screenshot of a web browser displaying a user interface for managing stock. The title bar says "A Web Page" and the address bar shows "http://balineseclassic.com". The main header includes a logo for "Balinese Classic", a "Username" field, and a search icon. On the left, there's a sidebar with navigation links: "Dashboard", "Supplier", "Kategori", "Produk", "Stock" (which is currently selected), and "Transaksi Beli". The main content area is titled "Stock" and contains a table with columns "No^", "Nama", "Stock", and "Action". At the top of this table is a "Tambah" button. The footer of the page includes the copyright notice "Copyright © 2019 Balinese Classic. All rights reserved."

Gambar 4.64 Antarmuka *List Stock* – Gudang

## 12. Desain *Form Stock*

*Form* kelola data *stock* adalah *form* yang digunakan untuk memasukkan data *stock* dan mengedit data *stock*. Desain antarmuka *form* kelola data *stock* dapat dilihat pada gambar berikut:

The form is titled "Form Stock". It has two input fields: "Nama Produk" containing "Nama Produk" and "Jumlah Stock" containing "800". At the bottom are two buttons: "Edit" and "Close".

Gambar 4.65 Antarmuka *Form Stock* – Gudang

### 13. Desain Antarmuka *List* Pembelian

Halaman ini menampilkan data-data pembelian yang ada di *database*, juga terdapat tombol untuk menuju *form* pembelia. Berikut dapat dilihat pada gambar di bawah ini:

Gambar 4.66 Antarmuka *List* Pembelian – Gudang

### 14. Desain Transaksi Pembelian

Pada halaman transaksi pembelian ini berisi *form* yang berfungsi untuk menampung data transaksi pembelian. Desain antarmuka transaksi pembelian dapat dilihat pada gambar berikut:

Gambar 4.67 Antarmuka *Form* Pembelian – Gudang

## 4.5 Implementasi Sistem

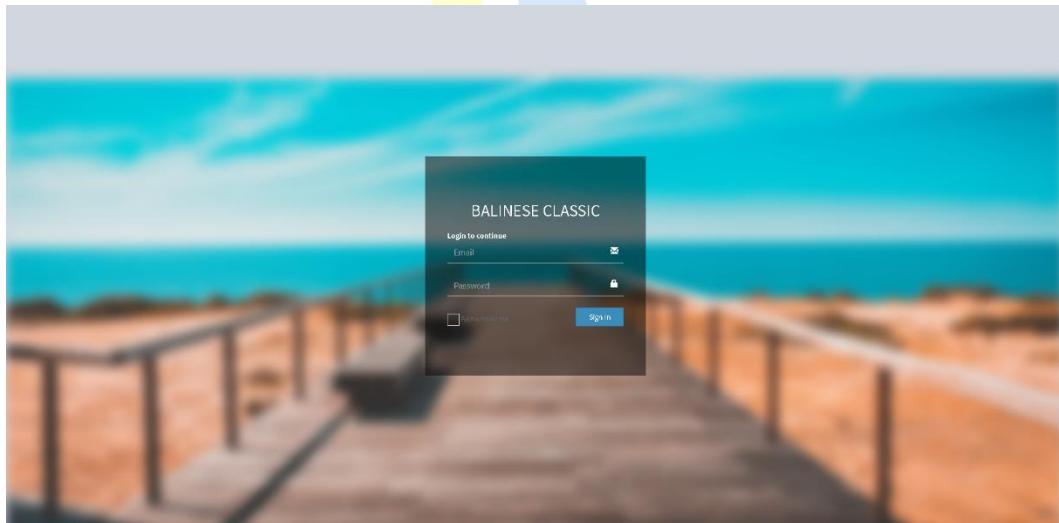
Pada implementasi sistem akan dijelaskan mengenai tampilan pada sistem. Berikut tampilan pada Sistem Informasi Eksekutif Penjualan Kerajinan Tas Rotan Pada Balinese Classic Menggunakan Framework Laravel

### 4.5.1 Tampilan Untuk Admin

Tampilan untuk Admin, ditujukan untuk *user* yang memiliki role admin, di mana terdapat tampilan sesuai dengan kebutuhannya. Adapun tampilannya sebagai berikut:

1. Tampilan Halaman *Login* – Admin

Pada halaman *login* sistem akan melakukan validasi agar Admin dapat masuk ke halaman sistem. Berikut adalah tampilan dari halaman *login*:



Gambar 4.68 Tampilan Halaman *Login* – Admin

2. Tampilan Halaman *Dashboard* – Admin

Halaman *Dashboard* merupakan tampilan setelah berhasil melakukan *login*. Berikut adalah tampilan halaman *login*:



Gambar 4.69 Tampilan Halaman *Dashboard* – Admin

### 3. Tampilan Halaman *List Users* – Admin

Halaman *List Users* merupakan halaman setelah klik menu *Users*. Pada halaman ini terdapat data-data *user* yang tersimpan di sistem. Berikut adalah tampilan halaman *list users*:

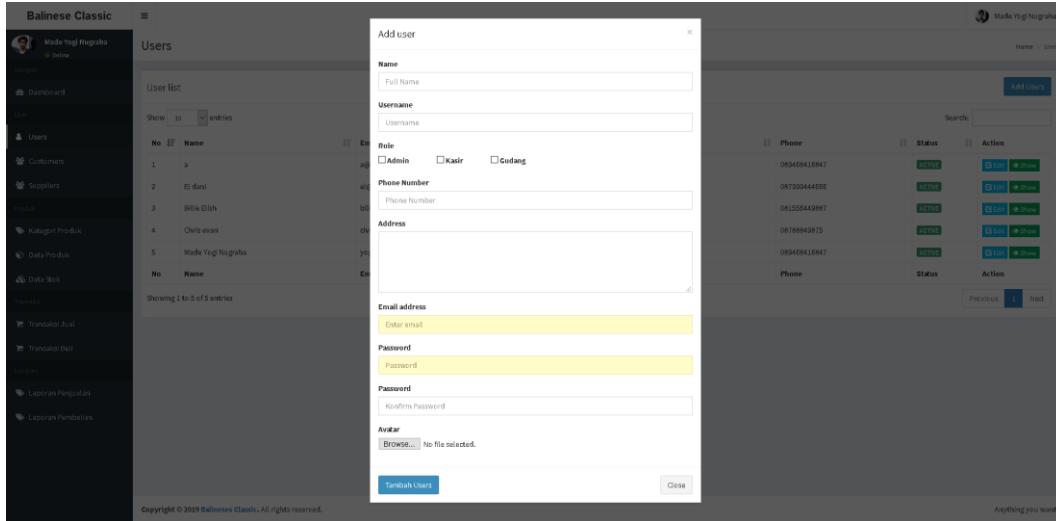
The screenshot shows the Balinese Classic *Users* list page with the following table:

No	Name	Email	Terdalat	Phone	Status	Action
1	a	a@gmail.com	2019-12-25 02:49:18	08468416947	ACTIVE	<a href="#">Edit</a> <a href="#">Show</a>
2	Ei dink	e@gmail.com	2019-12-24 05:42:34	087399444568	ACTIVE	<a href="#">Edit</a> <a href="#">Show</a>
3	Bille Bish	bille@gmail.com	2019-12-24 05:41:12	081556448987	ACTIVE	<a href="#">Edit</a> <a href="#">Show</a>
4	Chris evan	chrisewen@gmail.com	2019-12-24 05:40:03	08788849875	ACTIVE	<a href="#">Edit</a> <a href="#">Show</a>
5	Made Yogi Nugraha	yoginugraha19@gmail.com	2019-12-24 05:15:22	08468416947	ACTIVE	<a href="#">Edit</a> <a href="#">Show</a>

Gambar 4.70 Tampilan Halaman *List Users* – Admin

### 4. Tampilan *Form User* – Admin

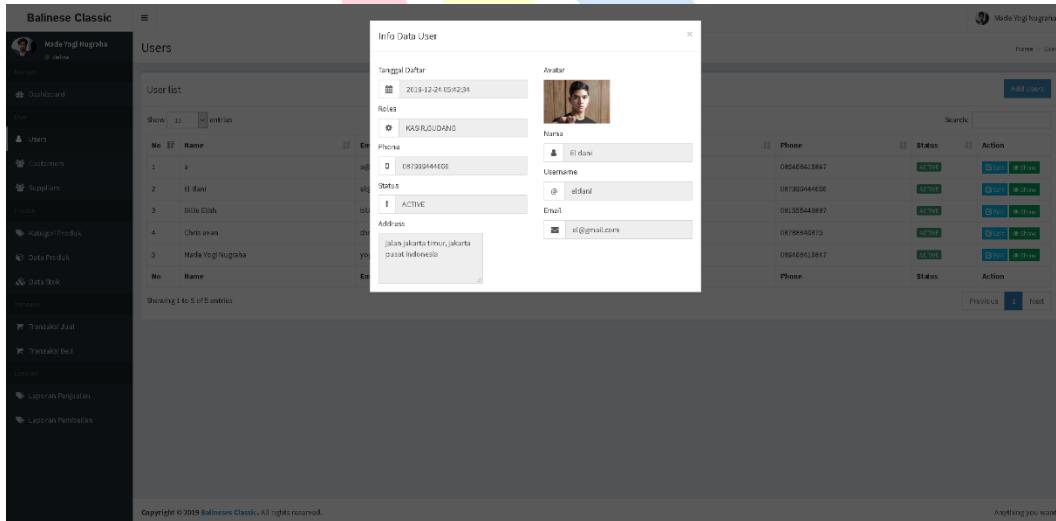
*Form User* merupakan *form* yang muncul setelah klik tombol *add user* pada halaman *list users*. Berikut adalah tampilan *form user*:



Gambar 4.71 Tampilan Form User – Admin

## 5. Tampilan Detail User – Admin

Detail *user* merupakan tampilan yang muncul setelah klik tombol *show* pada halaman *list users*. Berikut adalah tampilan detail *user*:



Gambar 4.72 Tampilan Detail User – Admin

## 6. Tampilan Halaman List Customers – Admin

Halaman *List Customers* merupakan halaman setelah klik menu *Customers*. Pada halaman ini terdapat data-data *Customers* yang tersimpan di sistem. Berikut adalah tampilan halaman *list Customers*:

No	Nama	Email	Perusahaan	Telepon	Tanggal Ditarik	status	Action
1	gunan	gunan@gmail.com	gunan corp	08245632111	2019-12-24 08:39:51	ACTIVE	
2	Muhamad	muhamad@gmail.com	muha gallery corp	021488955	2019-12-24 08:39:01	ACTIVE	
3	Moya	moya@gmail.com	moyay	0218529214	2019-12-24 08:38:04	ACTIVE	
4	anira	anira@gmail.com	anira corp	08221465054	2019-12-24 08:37:24	ACTIVE	
5	Aldi	aldi@gmail.com	aldi corp	021356477	2019-12-24 08:36:36	ACTIVE	
6	Bahia	bahia@gmail.com	bahia baby corp	0874548841	2019-12-24 08:35:59	ACTIVE	
7	Arie	arie@gmail.com	arie ford corp	0213565441	2019-12-24 08:35:07	ACTIVE	
8	Hilda	hilda@gmail.com	hildout corp	08554741425	2019-12-24 08:34:19	ACTIVE	
9	Elsa	elsa@gmail.com	frozen corp	0213335504	2019-12-24 08:32:18	ACTIVE	
10	Max	Max@gmail.com	Max car corp	0218022544	2019-12-24 07:12:08	ACTIVE	

Gambar 4.73 Tampilan Halaman *List Customers* – Admin

## 7. Tampilan *Form Customer* – Admin

*Form Customer* merupakan *form* yang muncul setelah klik tombol *add Customer* pada halaman *list Customers*. Berikut adalah tampilan *form Customer*.

Add customer

Name	<input type="text" value="Full Name"/>
Email address	<input type="text" value="Enter email"/>
Perusahaan	<input type="text" value="Perusahaan"/>
Phone Number	<input type="text" value="Phone Number"/>
Alamat	<input type="text" value="Alamat"/>
Avatar	<input type="button" value="Browse..."/> <span>No file selected.</span>
<input type="button" value="Tambah customer"/> <span style="margin-left: 20px;"><input type="button" value="Close"/></span>	

Gambar 4.74 Tampilan *Form Customer* – Admin

## 8. Tampilan Detail *Customer* – Admin

*Detail Customer* merupakan tampilan yang muncul setelah klik tombol *show* pada halaman *list Customers*. Berikut adalah tampilan detail *Customer*.

Gambar 4.75 Tampilan Detail Customer – Admin

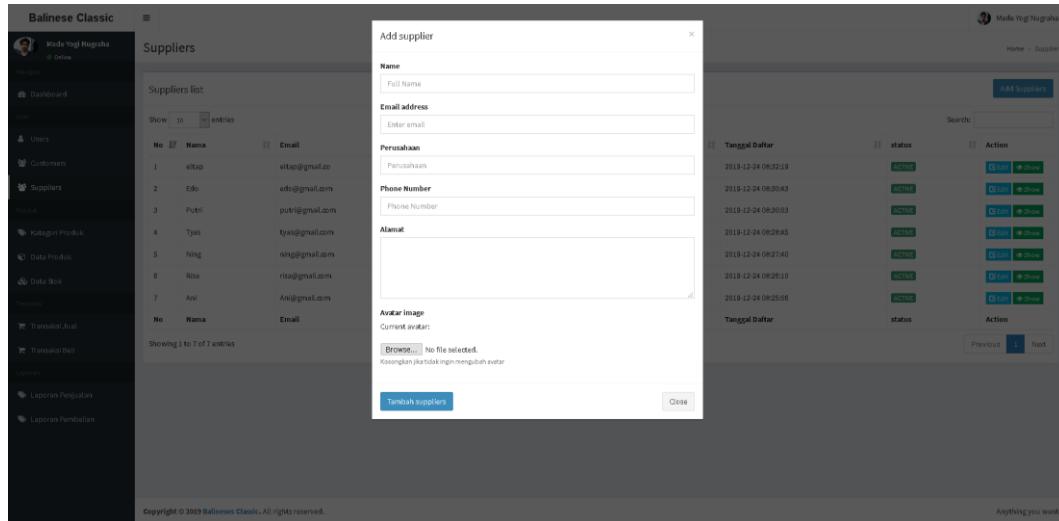
## 9. Tampilan Halaman *List Suppliers* – Admin

Halaman *List Suppliers* merupakan halaman setelah klik menu *Suppliers*. Pada halaman ini terdapat data-data *Suppliers* yang tersimpan di sistem. Berikut adalah tampilan halaman *list Suppliers*:

Gambar 4.76 Tampilan Halaman *List Suppliers* – Admin

## 10. Tampilan *Form Supplier* – Admin

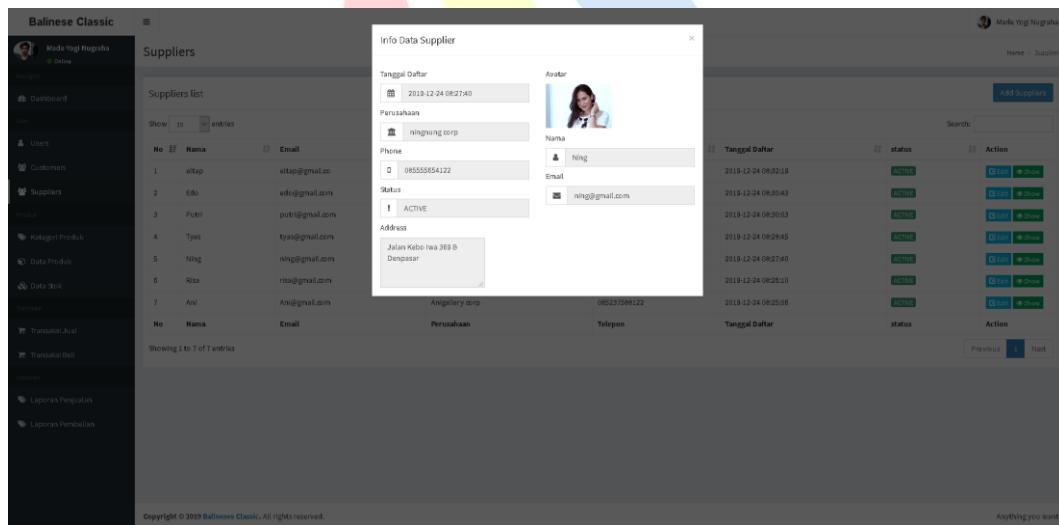
*Form Supplier* merupakan *form* yang muncul setelah klik tombol *add Supplier* pada halaman *list Suppliers*. Berikut adalah tampilan *form Supplier*:



Gambar 4.77 Tampilan *Form Supplier – Admin*

## 11. Tampilan Detail *Supplier* – Admin

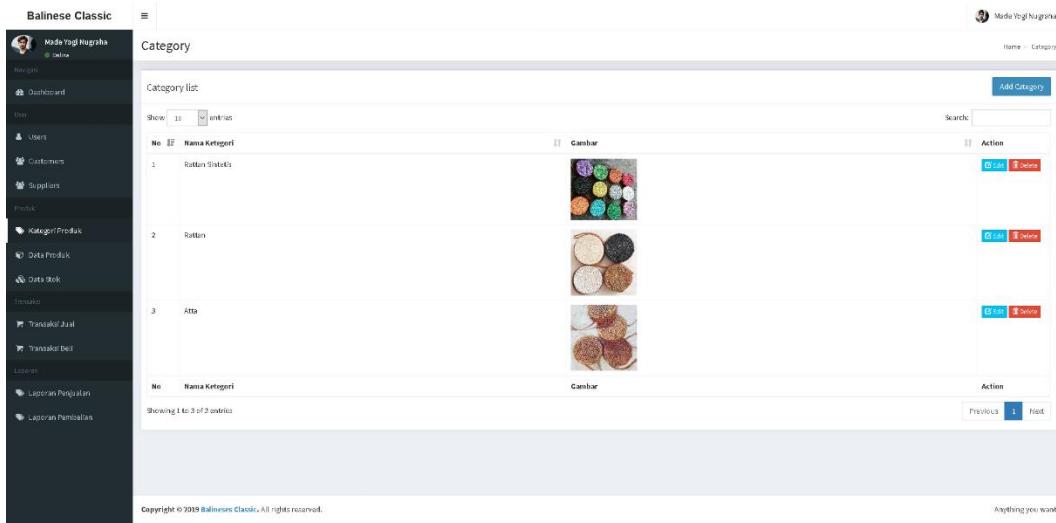
Detail *Supplier* merupakan tampilan yang muncul setelah klik tombol *show* pada halaman *list Suppliers*. Berikut adalah tampilan detail *Supplier*:



Gambar 4.78 Tampilan Detail *Supplier* – Admin

## 12. Tampilan Halaman *List Categories* – Admin

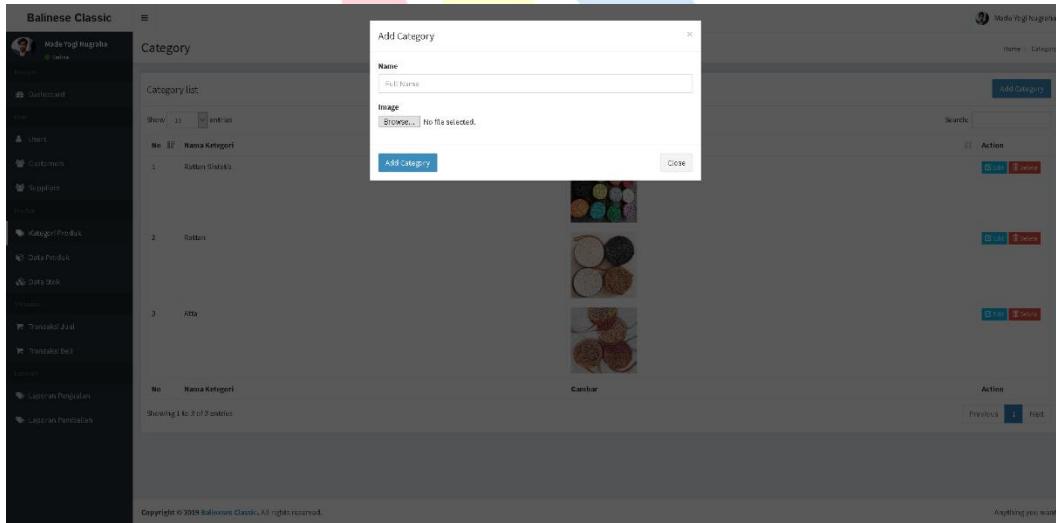
Halaman *List Categories* merupakan halaman setelah klik menu *Categories*. Pada halaman ini terdapat data-data *Categories* yang tersimpan di sistem. Berikut adalah tampilan halaman *list Categories*:



Gambar 4.79 Tampilan Halaman *List Categories* – Admin

### 13. Tampilan *Form Category* – Admin

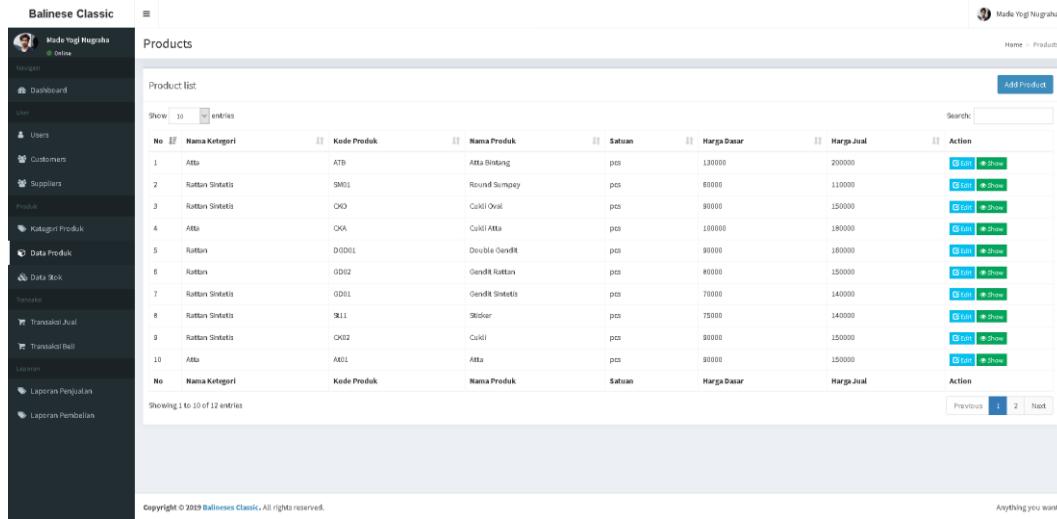
*Form Category* merupakan *form* yang muncul setelah klik tombol *add Category* pada halaman *list Categories*. Berikut adalah tampilan *form Category*:



Gambar 4.80 Tampilan *Form Category* – Admin

### 14. Tampilan Halaman *List Products* – Admin

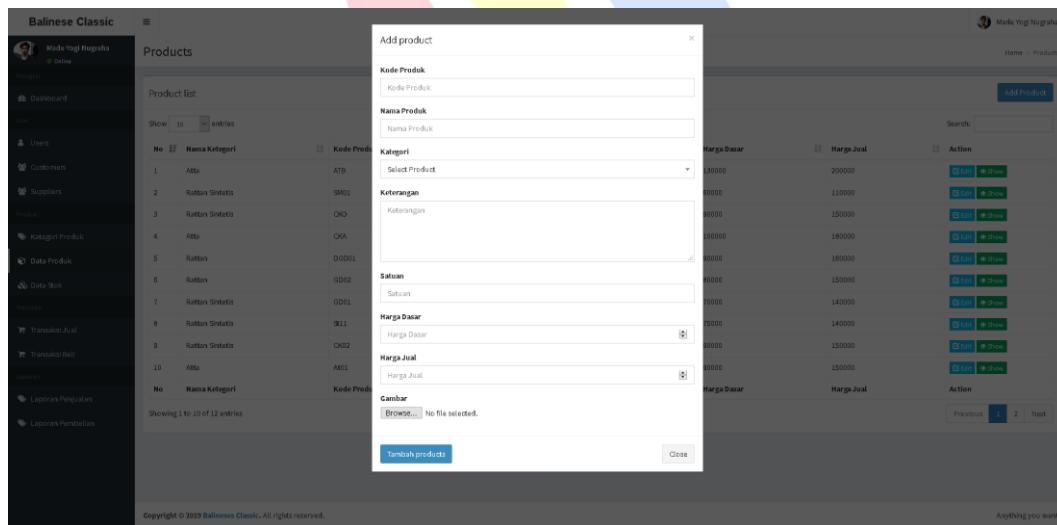
Detail *Category* merupakan tampilan yang muncul setelah klik tombol *show* pada halaman *list Categories*. Berikut adalah tampilan detail *Category*:



Gambar 4.81 Tampilan Halaman *List Products* – Admin

## 15. Tampilan *Form Product* – Admin

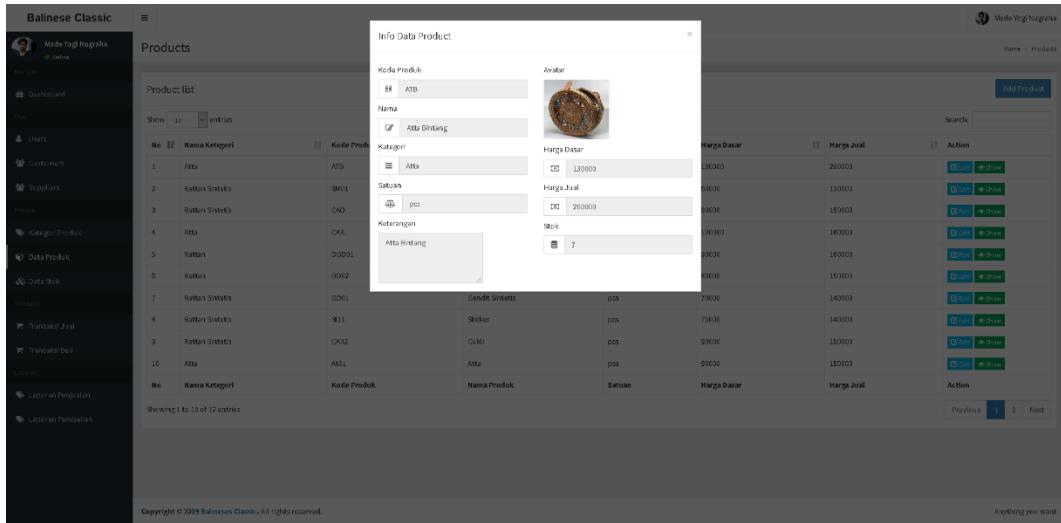
Halaman *List Products* merupakan halaman setelah klik menu *Products*. Pada halaman ini terdapat data-data *Products* yang tersimpan di sistem. Berikut adalah tampilan halaman *list Products*:



Gambar 4.82 Tampilan *Form Product* – Admin

## 16. Tampilan Detail *Product* – Admin

Detail *Product* merupakan tampilan yang muncul setelah klik tombol *show* pada halaman *list Products*. Berikut adalah tampilan detail *Product*:



Gambar 4.83 Tampilan Detail Product – Admin

## 17. Tampilan Halaman *List Stocks* – Admin

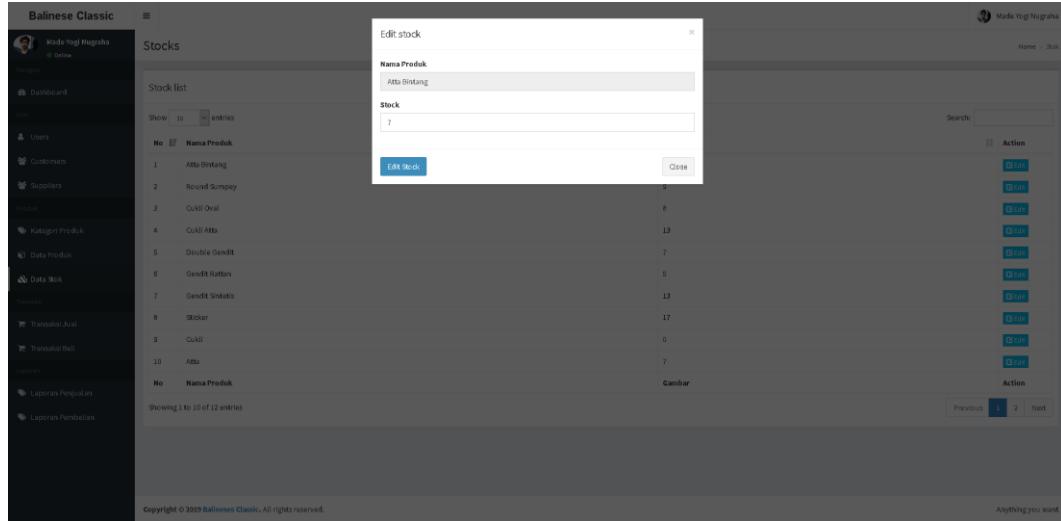
Halaman *List Stocks* merupakan halaman setelah klik menu *Stocks*. Pada halaman ini terdapat data-data *Stocks* yang tersimpan di sistem. Berikut adalah tampilan halaman *list Stocks*:

No	Nama Produk	Gambar	Action
1	Atta Bintang		
2	Round Sumpay		
3	Cukil Oval		
4	Cukil Atta		
5	Double Gendit		
6	Gendit Ratten		
7	Gendit Sintels		
8	Sticker		
9	Cukil		
10	Atta		

Gambar 4.84 Tampilan Halaman *List Stocks* – Admin

## 18. Tampilan *Form Stock* – Admin

*Form Stock* merupakan *form* yang muncul setelah klik tombol *add Stock* pada halaman *list Stocks*. Berikut adalah tampilan *form Stock*:



Gambar 4.85 Tampilan Form Stock – Admin

#### 19. Tampilan Halaman *List* Penjualan – Admin

Halaman *List* Penjualan merupakan halaman setelah klik menu transaksi jual. Pada halaman ini terdapat data-data transaksi jual yang tersimpan di sistem. Berikut adalah tampilan halaman *list* penjualan:

No	Invoice Number	Tanggal pesan	Customer	Produk	Total	Status	Action
1	7081	2019-12-30	Dika	Rattan   Ball   Gendit Sintets   Gendit Rattan   Cukil Atta	11160000	PENDING	<a href="#">Edit</a> <a href="#">Delete</a>
2	7080	2019-12-30	Balha	Rattan   Ball   Atta Bintang	2150000	PENDING	<a href="#">Edit</a> <a href="#">Delete</a>
3	7059	2019-12-30	Max	Gendit Sintets   Cukil Oval   Round Sumpay   Atta Bintang	6200000	PENDING	<a href="#">Edit</a> <a href="#">Delete</a>
4	7058	2019-12-29	Muhammad	Double Gendit   Cukil Atta   Cukil Oval   Round Sumpay   Atta Bintang	19650000	PENDING	<a href="#">Edit</a> <a href="#">Delete</a>
5	7057	2019-12-28	gurhan	Rattan   Ball   Atta   Cukil   Sticker	11850000	PENDING	<a href="#">Edit</a> <a href="#">Delete</a>
6	7056	2019-12-25	Elsa	Atta Bintang	5400000	PENDING	<a href="#">Edit</a> <a href="#">Delete</a>
7	7055	2019-12-25	Hilda	Rattan   Ball   Atta   Gendit Rattan	16700000	PENDING	<a href="#">Edit</a> <a href="#">Delete</a>
8	7054	2019-12-02	Yudi	Rattan   Ball   Atta   Sticker   Gendit Sintets   Double Gendit   Cukil Oval   Round Sumpay   Atta Bintang	29200000	PENDING	<a href="#">Edit</a> <a href="#">Delete</a>
9	7053	2019-11-27	Lutfi	Rattan   Ball   Sticker   Gendit Rattan   Double Gendit   Cukil Atta   Cukil Oval   Round Sumpay   Atta Bintang	33680000	PENDING	<a href="#">Edit</a> <a href="#">Delete</a>
10	7052	2019-11-20	Aldi	Rattan   Ball   Atta   Cukil   Sticker   Gendit Sintets   Cukil Atta	29350000	PENDING	<a href="#">Edit</a> <a href="#">Delete</a>

Gambar 4.86 Tampilan Halaman *List* Penjualan – Admin

#### 20. Tampilan Form Penjualan – Admin

Pada halaman ini terdapat *form* yang wajib diisi saat melakukan transaksi. Tampilan *form* penjualan dapat dilihat pada gambar berikut:

Gambar 4.87 Tampilan *Form Penjualan – Admin*

## 21. Tampilan Halaman *Invoice* Penjualan – Admin

Tampilan Halaman *Invoice* Penjualan merupakan tampilan setelah klik tombol *invoice* pada halaman *list* penjualan. Tampilan Halaman *Invoice* Penjualan dapat dilihat pada gambar berikut:

No	Product	Price	Qty	Subtotal
1	Rattan	Rp 110,000	10	Rp 1.100.000
2	Ball	Rp 120,000	15	Rp 1.800.000
3	Gendit Sintetis	Rp 140,000	8	Rp 1.120.000
4	Gendit Rattan	Rp 150,000	28	Rp 3.700.000
5	Guli Atta	Rp 180,000	8	Rp 1.440.000

Gambar 4.88 Tampilan Halaman *Invoice* Penjualan – Admin

## 22. Tampilan Halaman *List* Pembelian – Admin

Halaman *List* Pembelian merupakan halaman setelah klik menu transaksi beli. Pada halaman ini terdapat data-data transaksi beli yang tersimpan di sistem. Berikut adalah tampilan halaman *list* pembelian:

No	Invoice Number	Tanggal pesan	Customer	Produk	Total	Status	Action
1	3048	2019-12-25	Ari	Gendit Sintetis   Gendit Rattan   Cukil Atta	5750000	PENDING	<a href="#">View</a> <a href="#">Edit</a>
2	3047	2019-12-23	Tya	Cukil Oval   Round Sumpey   Atta Bintang	7350000	PENDING	<a href="#">View</a> <a href="#">Edit</a>
3	3046	2019-12-24	Ning	Gendit Sintetis   Gendit Rattan   Double Gendit   Cukil Atta	6800000	PENDING	<a href="#">View</a> <a href="#">Edit</a>
4	3045	2019-12-25	Risa	Rattan   Bali   Atta   Cukil   Sticker	7900000	PENDING	<a href="#">View</a> <a href="#">Edit</a>
5	3044	2019-12-25	Ning	Sticker	6000000	PENDING	<a href="#">View</a> <a href="#">Edit</a>
6	3043	2019-11-01	Putri	Rattan   Bali   Atta   Round Sumpey   Atta Bintang	15300000	PENDING	<a href="#">View</a> <a href="#">Edit</a>
7	3042	2019-10-27	elisa	Cukil Oval   Round Sumpey   Atta Bintang	10010000	PENDING	<a href="#">View</a> <a href="#">Edit</a>
8	3041	2019-10-23	Putri	Cukil Atta	5000000	PENDING	<a href="#">View</a> <a href="#">Edit</a>
9	3040	2019-11-01	Risa	Rattan   Bali   Sticker	9635000	PENDING	<a href="#">View</a> <a href="#">Edit</a>
10	3039	2019-10-10	Putri	Gendit Rattan   Double Gendit	3650000	PENDING	<a href="#">View</a> <a href="#">Edit</a>

Gambar 4.89 Tampilan Halaman *List Pembelian – Admin*

### 23. Tampilan Halaman *Form Pembelian – Admin*

Pada halaman ini terdapat *form* yang wajib diisi saat melakukan transaksi.

Tampilan *form* pembelian dapat dilihat pada gambar berikut:

Gambar 4.90 Tampilan Halaman *Form Pembelian – Admin*

#### 24. Tampilan Halaman *Invoice* Pembelian – Admin

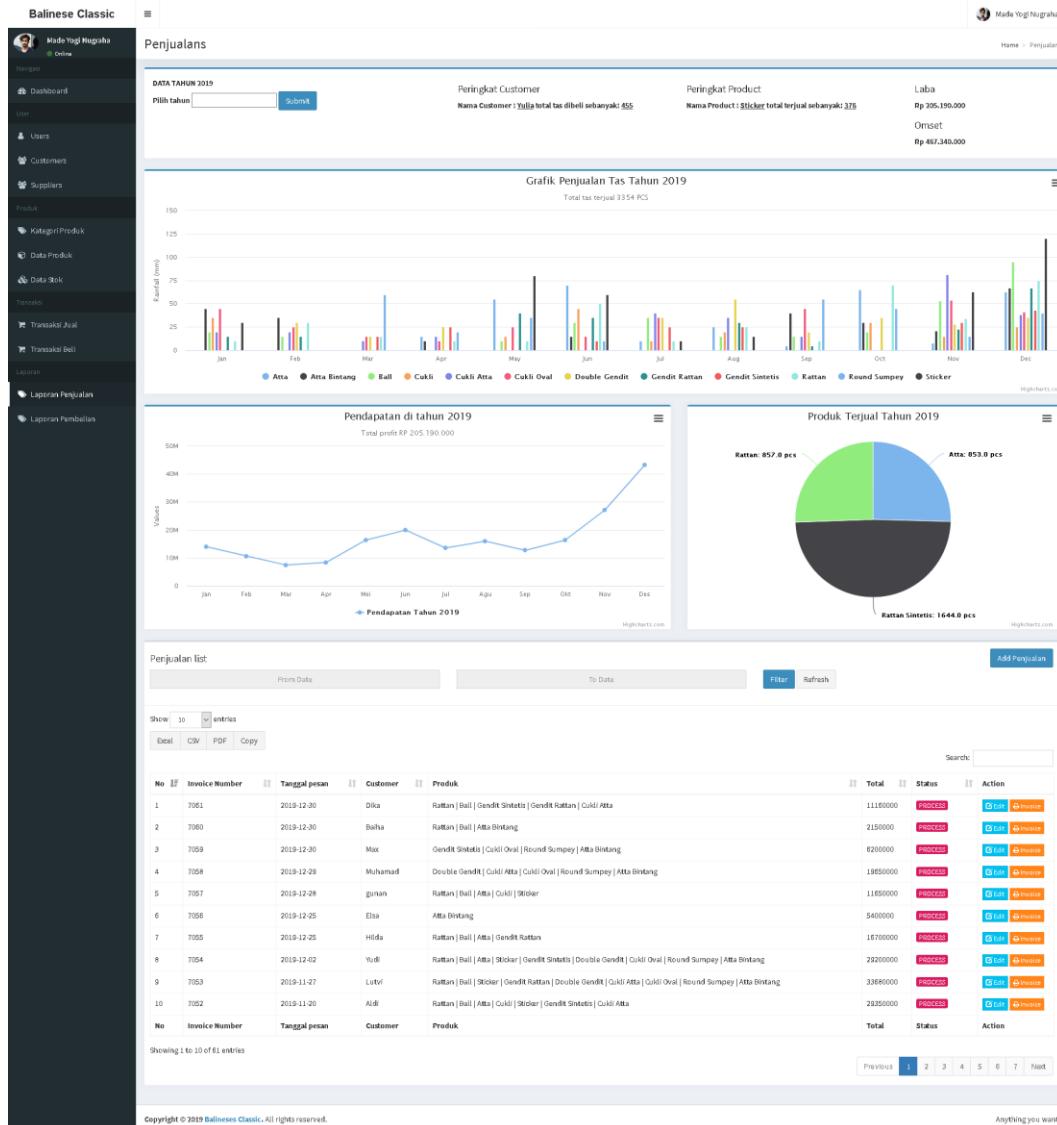
Tampilan Halaman *Invoice* Pembelian merupakan tampilan setelah klik tombol *invoice* pada halaman *list* pembelian. Tampilan Halaman *Invoice* Pembelian dapat dilihat pada gambar berikut:

Gambar 4.91 Tampilan Halaman *Invoice* Pembelian – Admin

#### 25. Tampilan Halaman Laporan Penjualan – Admin

Tampilan Halaman Laporan Penjualan merupakan tampilan setelah klik menu Laporan Penjualan. Pada halaman ini terdapat laporan dalam bentuk grafik dan data *grid*. Tampilan Halaman Laporan Penjualan dapat dilihat pada gambar berikut:



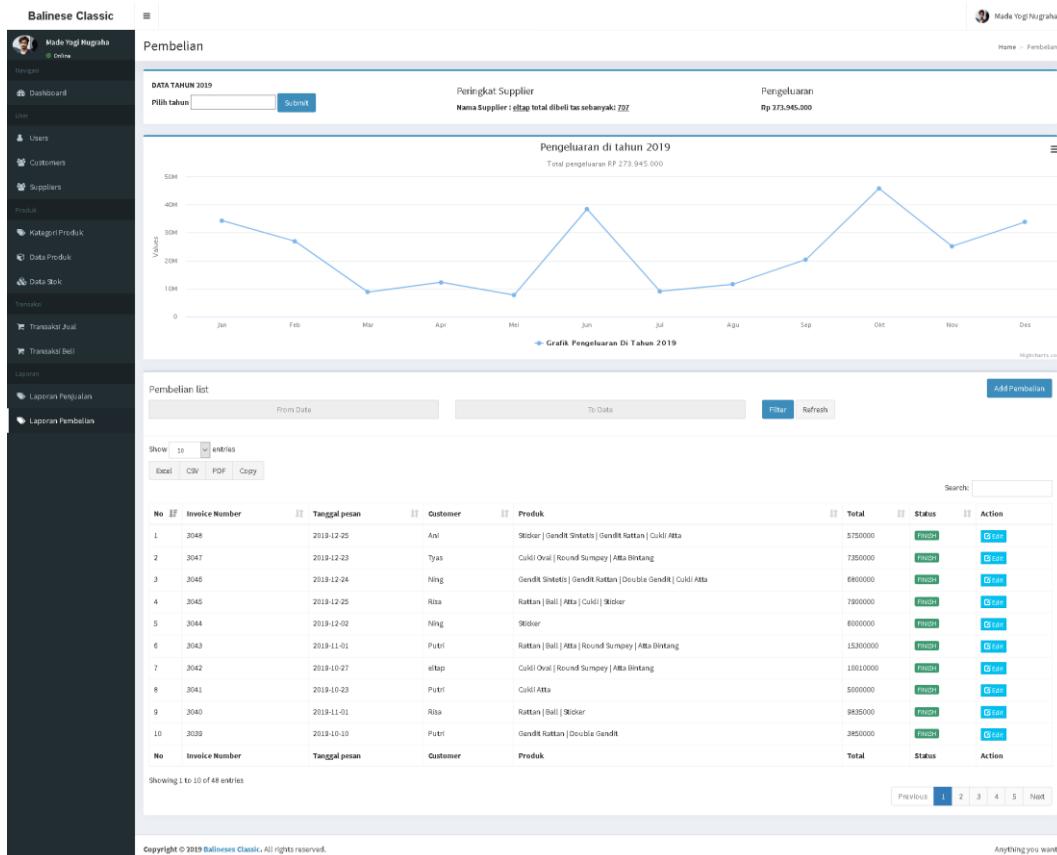


Gambar 4.92 Tampilan Halaman Laporan Penjualan – Admin

## INSTITUT TEKNOLOGI DAN BISNIS STIKOM BALI

### 26. Tampilan Halaman Laporan Pembelian – Admin

Tampilan Halaman Laporan Pembelian merupakan tampilan setelah klik menu Laporan Pembelian. Pada halaman ini terdapat laporan dalam bentuk grafik dan data grid. Tampilan Halaman Laporan Pembelian dapat dilihat pada gambar berikut:

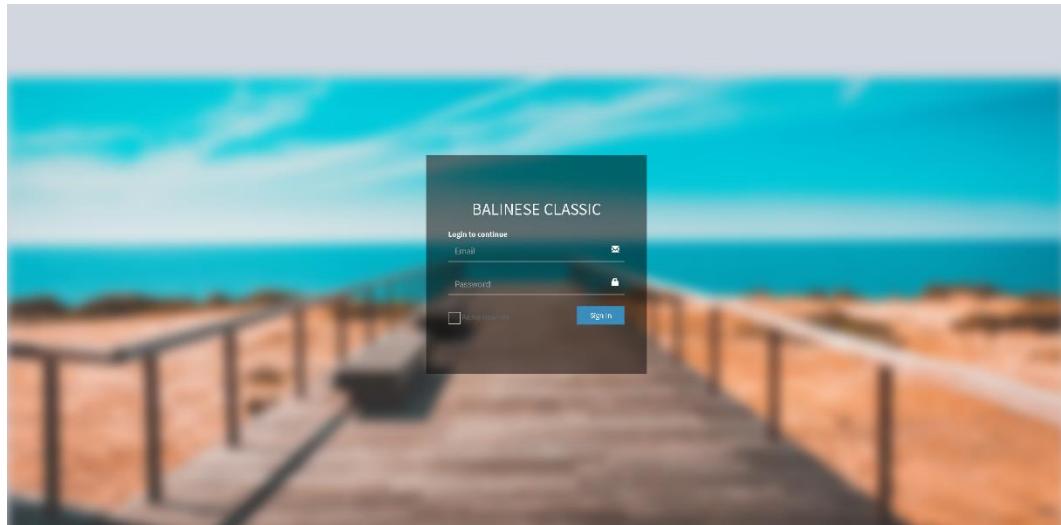


Gambar 4.93 Tampilan Halaman Laporan Pembelian – Admin

#### **4.5.2 Tampilan Untuk Kasir**

Tampilan untuk Kasir, ditujukan untuk *user* yang memiliki role Kasir, di mana terdapat tampilan sesuai dengan kebutuhannya. Adapun tampilannya sebagai berikut:

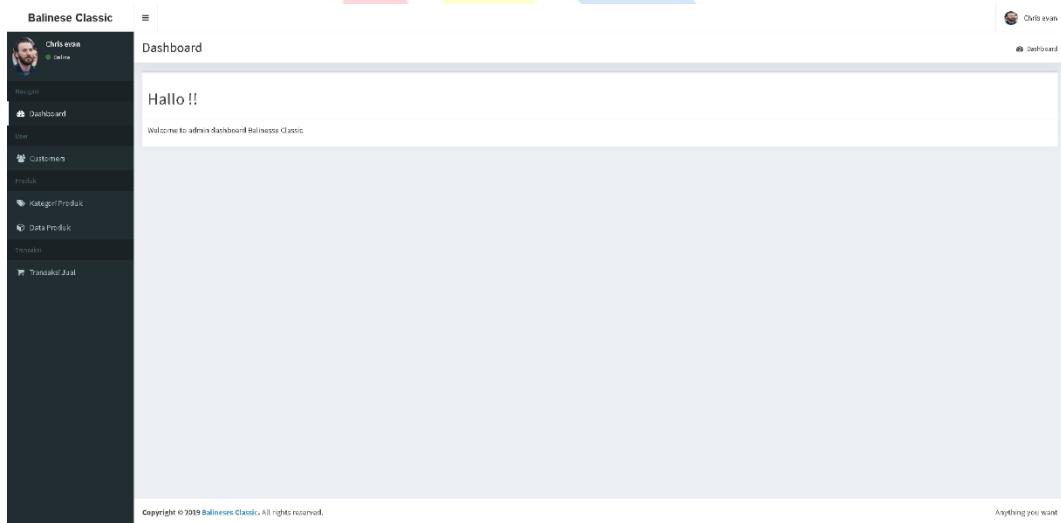
Pada halaman *login* sistem akan melakukan validasi agar Kasir dapat masuk ke halaman sistem. Berikut adalah tampilan dari halaman *login*:



Gambar 4.94 Tampilan Halaman *Login* – Kasir

## 2. Tampilan Halaman *Dashboard* – Kasir

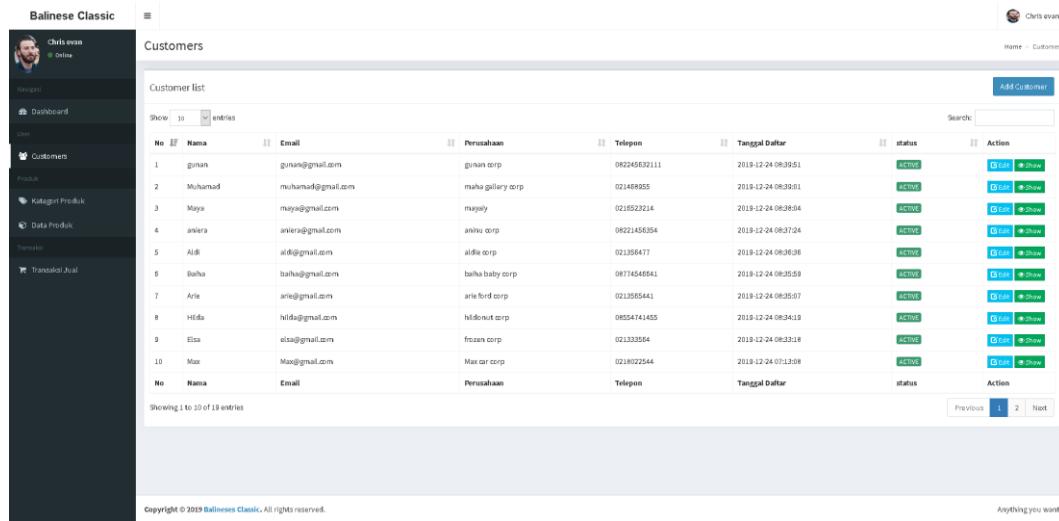
Halaman *Dashboard* merupakan tampilan setelah berhasil melakukan *login*. Berikut adalah tampilan halaman *login*:



Gambar 4.95 Tampilan Halaman *Dashboard* – Kasir

## 3. Tampilan Halaman *List Customers* – Kasir

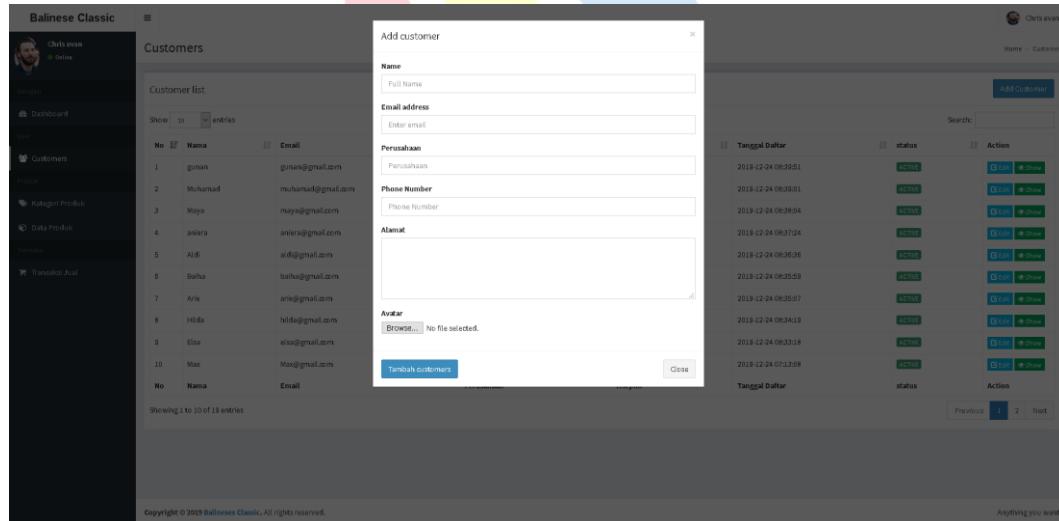
Halaman *List Customers* merupakan halaman setelah klik menu *Customers*. Pada halaman ini terdapat data-data *Customers* yang tersimpan di sistem. Berikut adalah tampilan halaman *list Customers*:



Gambar 4.96 Tampilan Halaman List Customers – Kasir

#### 4. Tampilan Form Customer – Kasir

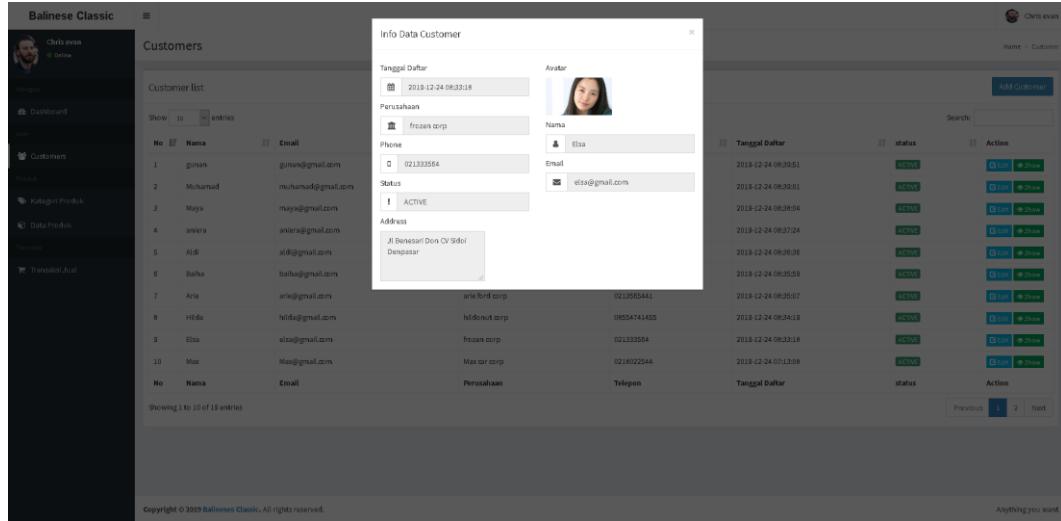
Form Customer merupakan form yang muncul setelah klik tombol add Customer pada halaman list Customers. Berikut adalah tampilan form Customer.



Gambar 4.97 Tampilan Form Customer – Kasir

#### 5. Tampilan Detail Customer – Kasir

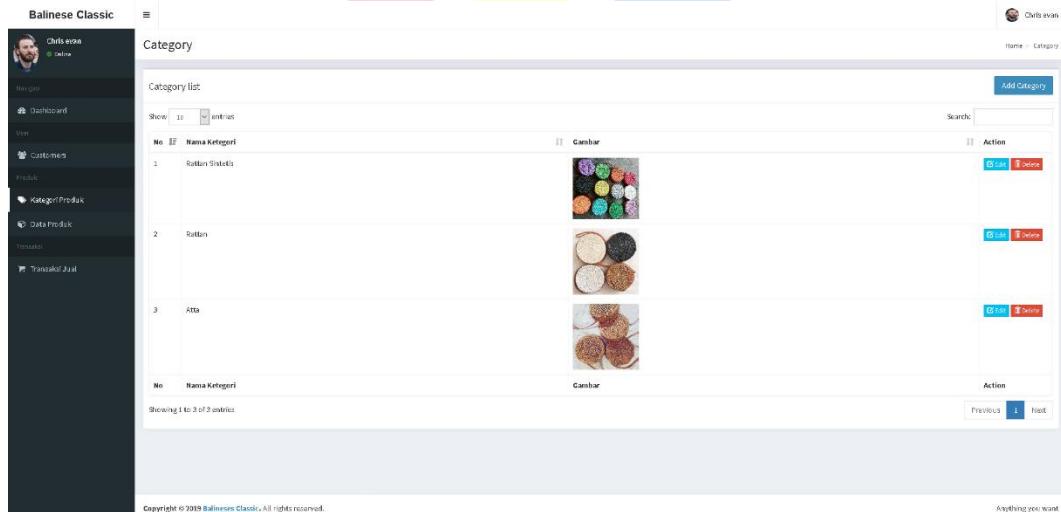
Detail Customer merupakan tampilan yang muncul setelah klik tombol show pada halaman list Customers. Berikut adalah tampilan detail Customer.



Gambar 4.98 Tampilan Detail Customer – Kasir

## 6. Tampilan Halaman *List Categories* – Kasir

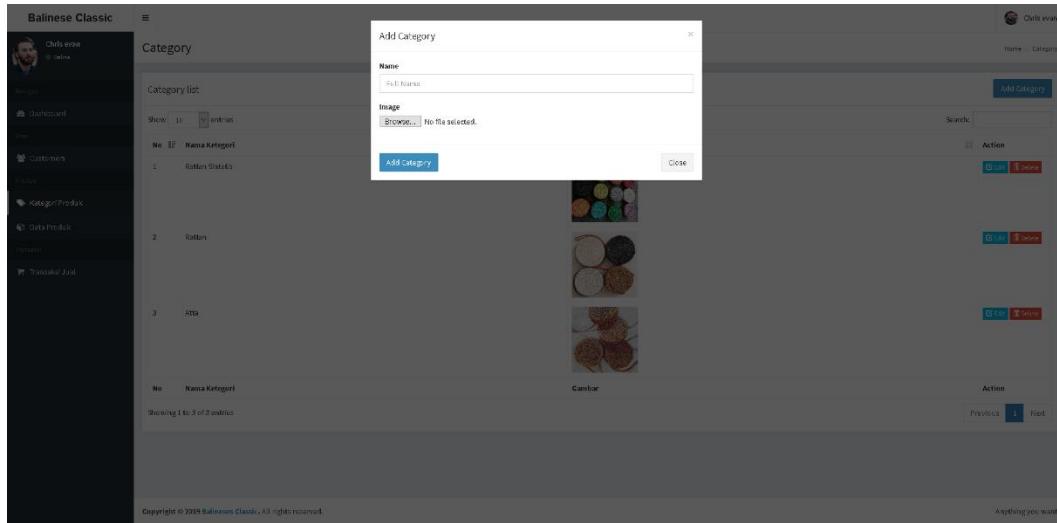
Halaman *List Categories* merupakan halaman setelah klik menu *Categories*. Pada halaman ini terdapat data-data *Categories* yang tersimpan di sistem. Berikut adalah tampilan halaman *list Categories*:



Gambar 4.99 Tampilan Halaman *List Categories* – Kasir

## 7. Tampilan *Form Category* – Kasir

*Form Category* merupakan *form* yang muncul setelah klik tombol *add Category* pada halaman *list Categories*. Berikut adalah tampilan *form Category*:



Gambar 4.100 Tampilan *Form Category – Kasir*

## 8. Tampilan Halaman *List Products – Kasir*

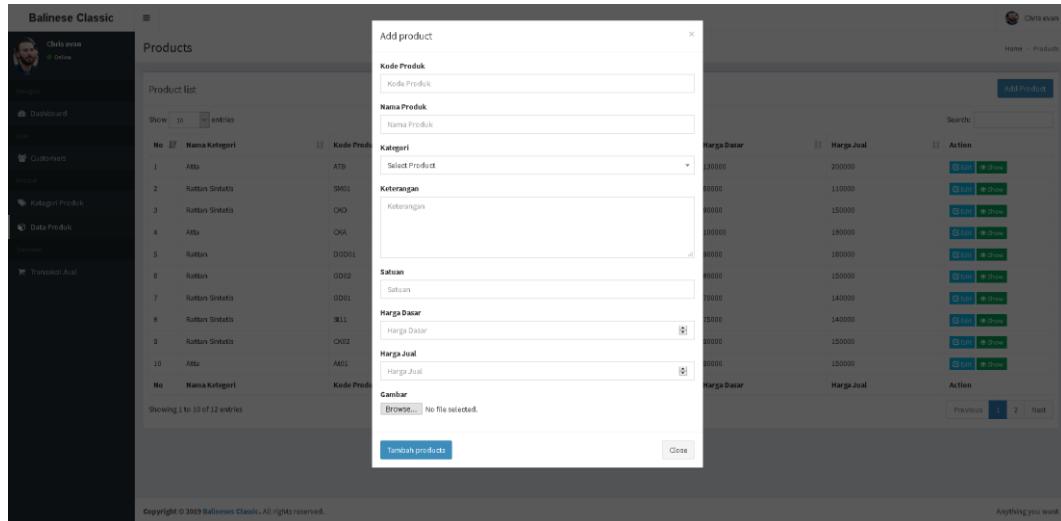
Detail *Category* merupakan tampilan yang muncul setelah klik tombol *Show* pada halaman *list Categories*. Berikut adalah tampilan detail *Category*:

No	Nama Kategori	Kode Produk	Nama Produk	Satuan	Harga Dasar	Harga Jual	Action
1	Atta	ATB	Atta Bintang	pcs	130000	200000	
2	Rattan Sintetis	SM01	Round Sumpay	pcs	80000	110000	
3	Rattan Sintetis	OJO	Cakil Oval	pcs	90000	150000	
4	Atta	OKA	Cakil Atta	pcs	100000	180000	
5	Rattan	DOD01	Double Gendit	pcs	90000	180000	
6	Rattan	GD012	Gendit Rattan	pcs	80000	150000	
7	Rattan Sintetis	GD011	Gendit Sintetis	pcs	70000	140000	
8	Rattan Sintetis	GD11	Gidder	pcs	75000	140000	
9	Rattan Sintetis	GD012	Cakil	pcs	80000	150000	
10	Atta	AM01	Atta	pcs	80000	150000	

Gambar 4.101 Tampilan Halaman *List Products – Kasir*

## 9. Tampilan *Form Product – Kasir*

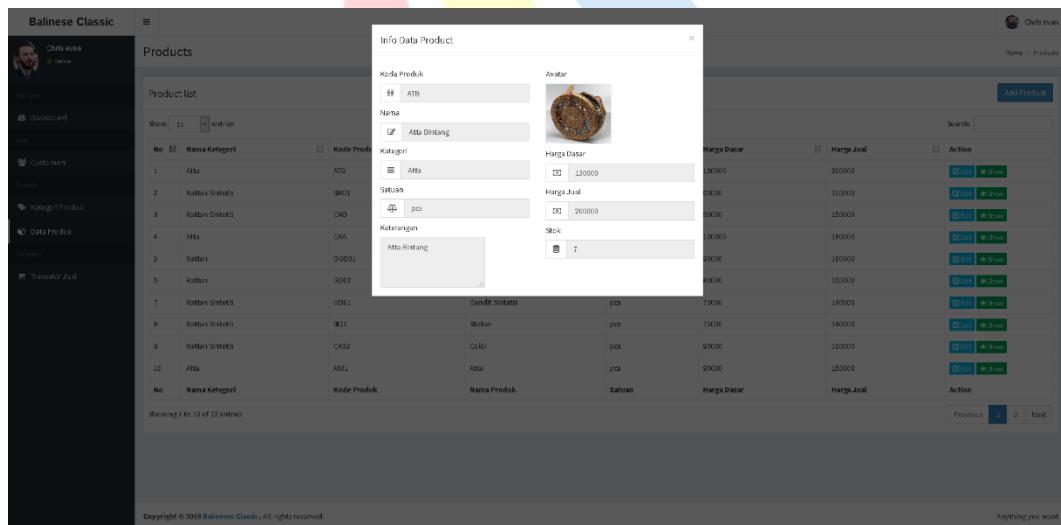
Halaman *List Products* merupakan halaman setelah klik menu *Products*. Pada halaman ini terdapat data-data *Products* yang tersimpan di sistem. Berikut adalah tampilan halaman *list Products*:



Gambar 4.102 Tampilan Form Product – Kasir

## 10. Tampilan Detail Product – Kasir

Detail *Product* merupakan tampilan yang muncul setelah klik tombol *show* pada halaman *list Products*. Berikut adalah tampilan detail *Product*:



Gambar 4.103 Tampilan Detail Product – Kasir

## 11. Tampilan Halaman *List* Penjualan – Kasir

Halaman *List* Penjualan merupakan halaman setelah klik menu transaksi jual. Pada halaman ini terdapat data-data transaksi jual yang tersimpan di sistem. Berikut adalah tampilan halaman *list* penjualan:

No	Invoice Number	Tanggal pesan	Customer	Produk	Total	Status	Action
1	7051	2019-12-30	Dika	Rattan   Ball   Gendit Sintetis   Gendit Rattan   Cukil Atta	11100000	PROSES	
2	7050	2019-12-30	Bella	Rattan   Ball   Atta Bintang	2150000	PROSES	
3	7059	2019-12-30	Max	Gendit Sintetis   Cukil Oval   Round Sumpay   Atta Bintang	6200000	PROSES	
4	7058	2019-12-29	Muhammad	Double Gendit   Cukil Atta   Cukil Oval   Round Sumpay   Atta Bintang	19650000	PROSES	
5	7057	2019-12-28	prawin	Rattan   Ball   Atta   Cukil   Sticker	11600000	PROSES	
6	7056	2019-12-25	Elsa	Atta Bintang	5400000	PROSES	
7	7055	2019-12-25	Hilda	Rattan   Ball   Atta   Gendit Rattan	16700000	PROSES	
8	7054	2019-12-02	Yudi	Rattan   Ball   Sticker   Gendit Sintetis   Double Gendit   Cukil Oval   Round Sumpay   Atta Bintang	29200000	PROSES	
9	7053	2019-11-27	Lubri	Rattan   Ball   Sticker   Gendit Rattan   Double Gendit   Cukil Atta   Cukil Oval   Round Sumpay   Atta Bintang	38600000	PROSES	
10	7052	2019-11-20	Aldi	Rattan   Ball   Atta   Cukil   Sticker   Gendit Sintetis   Cukil Atta	23500000	PROSES	

Showing 1 to 10 of 61 entries

Copyright © 2019 Balinese Classic. All rights reserved.

Anything you want.

Gambar 4.104 Tampilan Halaman *List Penjualan – Kasir*

## 12. Tampilan *Form Penjualan – Kasir*

Pada halaman ini terdapat *form* yang wajib diisi saat melakukan transaksi.

Tampilan *form* penjualan dapat dilihat pada gambar berikut:

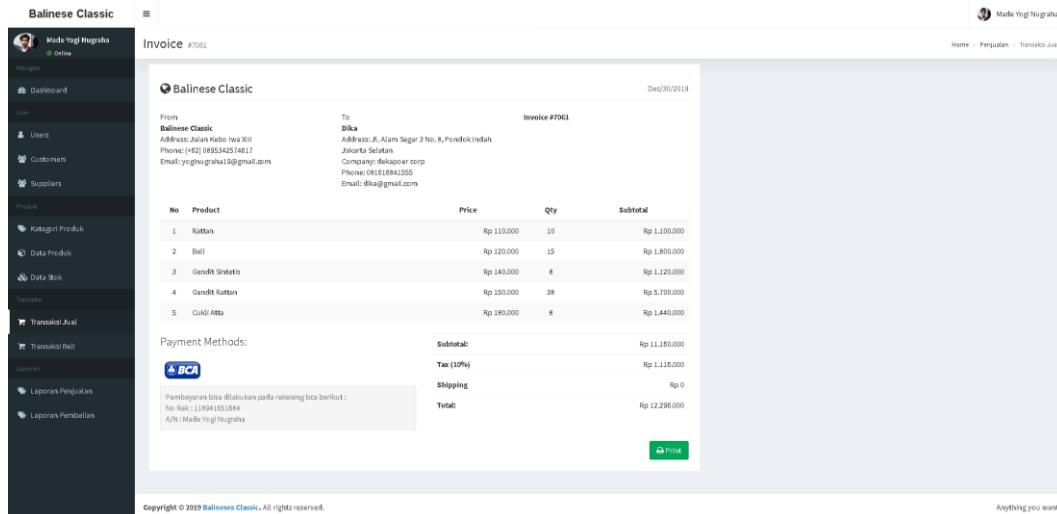
Copyright © 2019 Balinese Classic. All rights reserved.

Anything you want.

Gambar 4.105 Tampilan *Form Penjualan – Kasir*

## 13. Tampilan Halaman *Invoice Penjualan – Kasir*

Tampilan Halaman *Invoice Penjualan* merupakan tampilan setelah klik tombol *invoice* pada halaman *list* penjualan. Tampilan Halaman *Invoice Penjualan* dapat dilihat pada gambar berikut:



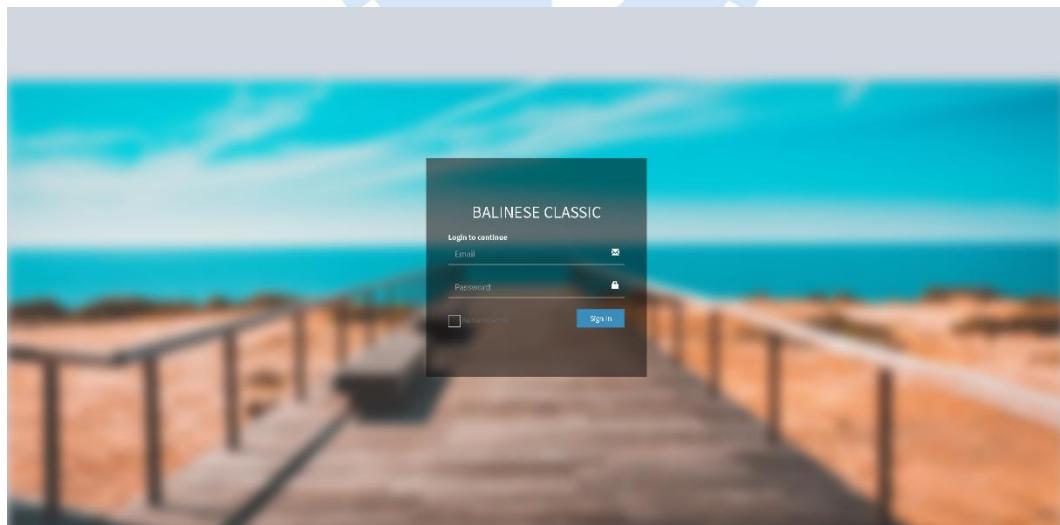
Gambar 4.106 Tampilan Halaman *Invoice Penjualan – Kasir*

#### 4.5.3 Tampilan Untuk Gudang

Tampilan untuk Gudang, ditujukan untuk *user* yang memiliki role Gudang, di mana terdapat tampilan sesuai dengan kebutuhannya. Adapun tampilannya sebagai berikut:

##### 1. Tampilan Halaman *Login – Gudang*

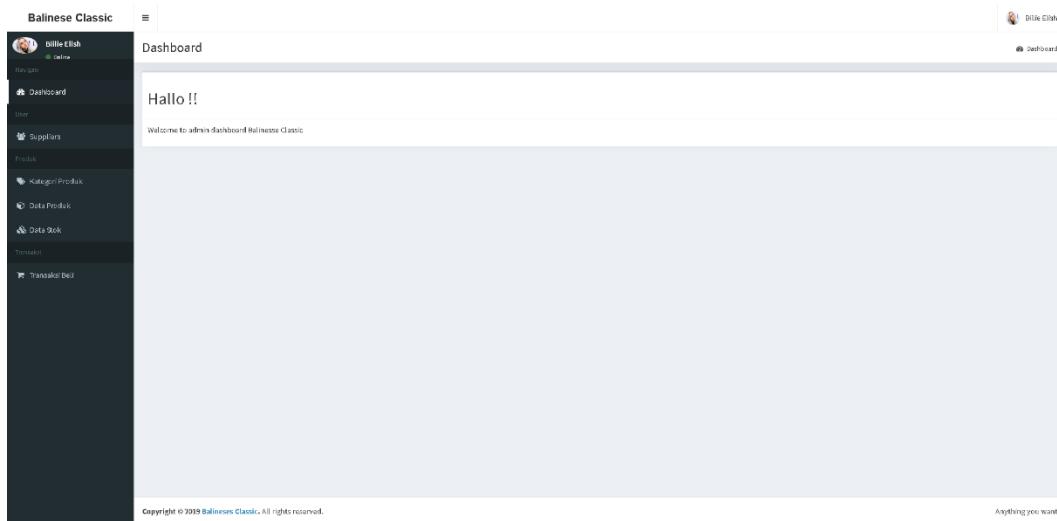
Pada halaman *login* sistem akan melakukan validasi agar Gudang dapat masuk ke halaman sistem. Berikut adalah tampilan dari halaman *login*:



Gambar 4.107 Tampilan Halaman *Login – Gudang*

## 2. Tampilan Halaman *Dashboard* – Gudang

Halaman *Dashboard* merupakan tampilan setelah berhasil melakukan *login*. Berikut adalah tampilan halaman *login*:



Gambar 4.108 Tampilan Halaman *Dashboard* – Gudang

## 3. Tampilan Halaman *List Suppliers* – Gudang

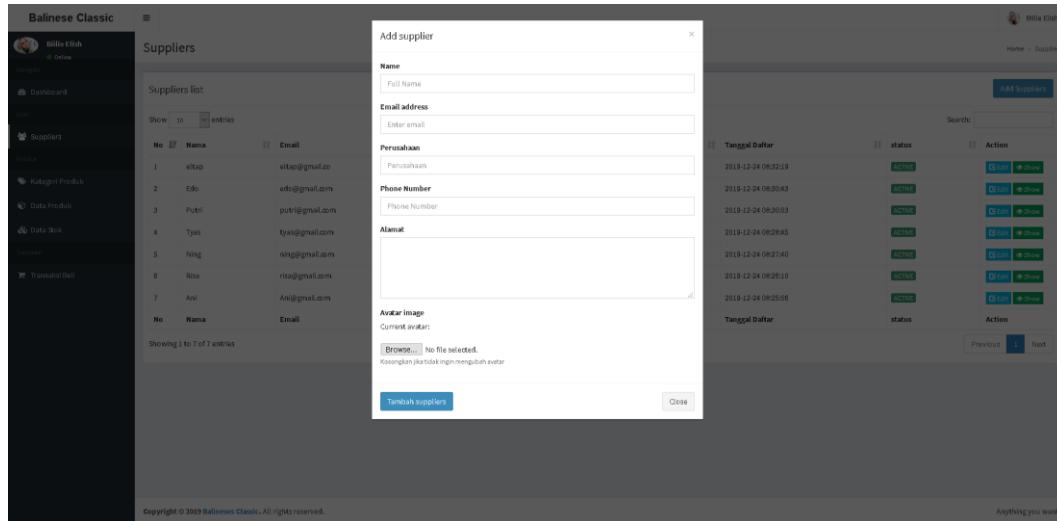
Halaman *List Suppliers* merupakan halaman setelah klik menu *Suppliers*. Pada halaman ini terdapat data-data *Suppliers* yang tersimpan di sistem. Berikut adalah tampilan halaman *list Suppliers*:

No	Nama	Email	Perusahaan	Telepon	Tanggal Daftar	Status	Action
1	elstap	elstap@gmail.co	elstap gallery corp	082245574155	2019-12-24 08:32:19	Aktif	<a href="#">Edit</a> <a href="#">Show</a>
2	Edo	edo@gmail.com	edo corp	02133392154	2019-12-24 08:30:43	Aktif	<a href="#">Edit</a> <a href="#">Show</a>
3	Putri	putri@gmail.com	putri snack corp	0218221154	2019-12-24 08:30:03	Aktif	<a href="#">Edit</a> <a href="#">Show</a>
4	Tya	tyas@gmail.com	tyasa gold corp	082208592111	2019-12-24 08:28:45	Aktif	<a href="#">Edit</a> <a href="#">Show</a>
5	Ning	ning@gmail.com	ningnung corp	085550584122	2019-12-24 08:27:40	Aktif	<a href="#">Edit</a> <a href="#">Show</a>
6	Rita	rita@gmail.com	rita bakery corp	082244155541	2019-12-24 08:26:10	Aktif	<a href="#">Edit</a> <a href="#">Show</a>
7	Ani	Ani@gmail.com	Anigallery corp	085237586122	2019-12-24 08:25:08	Aktif	<a href="#">Edit</a> <a href="#">Show</a>

Gambar 4.109 Tampilan Halaman *List Suppliers* – Gudang

## 4. Tampilan *Form Supplier* – Gudang

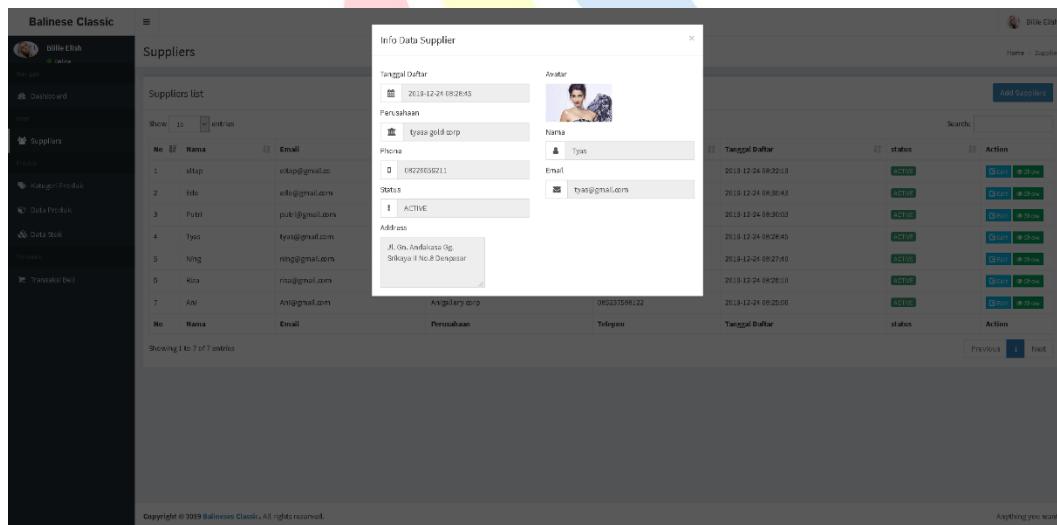
*Form Supplier* merupakan *form* yang muncul setelah klik tombol *add Supplier* pada halaman *list Suppliers*. Berikut adalah tampilan *form Supplier*:



Gambar 4.110 Tampilan Form Supplier – Gudang

## 5. Tampilan Detail Supplier – Gudang

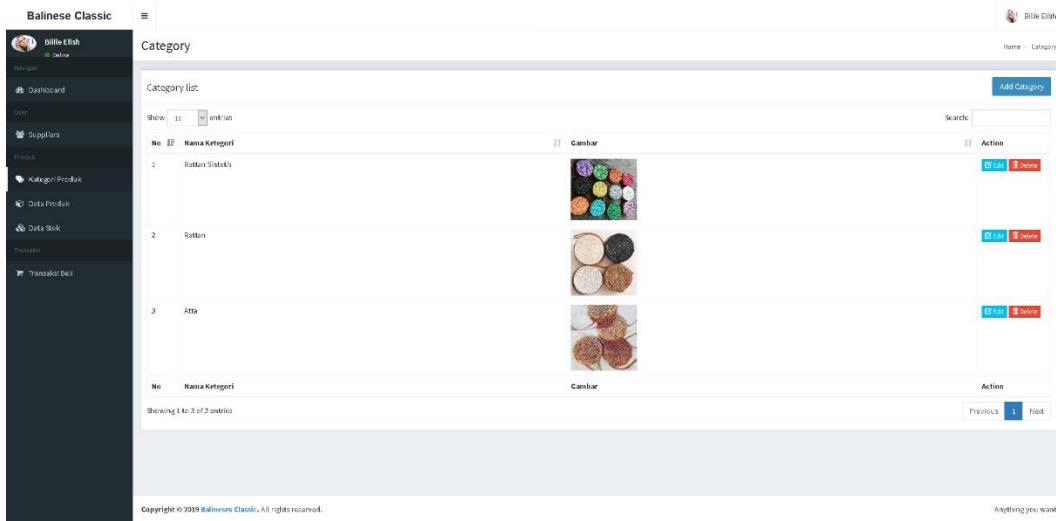
Detail *Supplier* merupakan tampilan yang muncul setelah klik tombol *show* pada halaman *list Suppliers*. Berikut adalah tampilan detail *Supplier*:



Gambar 4.111 Tampilan Detail Supplier – Gudang

## 6. Tampilan Halaman List Categories – Gudang

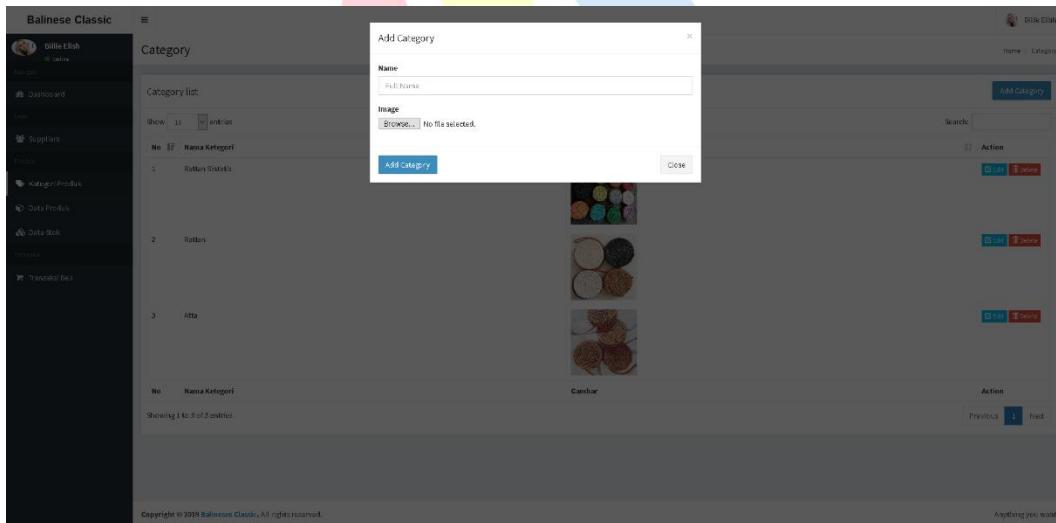
Halaman *List Categories* merupakan halaman setelah klik menu *Categories*. Pada halaman ini terdapat data-data *Categories* yang tersimpan di sistem. Berikut adalah tampilan halaman *list Categories*:



Gambar 4.112 Tampilan Halaman *List Categories* – Gudang

## 7. Tampilan *Form Category* – Gudang

*Form Category* merupakan *form* yang muncul setelah klik tombol *add Category* pada halaman *list Categories*. Berikut adalah tampilan *form Category*:



Gambar 4.113 Tampilan *Form Category* – Gudang

## 8. Tampilan Halaman *List Products* – Gudang

Detail *Category* merupakan tampilan yang muncul setelah klik tombol *show* pada halaman *list Categories*. Berikut adalah tampilan detail *Category*:

No	Nama Kategori	Kode Produk	Nama Produk	Satuan	Harga Dasar	Harga Jual	Action
1	Atta	ATB	Atta Bintang	pcs	100000	200000	<a href="#">Edit</a> <a href="#">Show</a>
2	Rattan Shlefis	SM01	Round Sumpay	pcs	60000	110000	<a href="#">Edit</a> <a href="#">Show</a>
3	Rattan Shlefis	OKD	Oval Oval	pcs	90000	150000	<a href="#">Edit</a> <a href="#">Show</a>
4	Atta	OKA	Oval Atta	pcs	100000	180000	<a href="#">Edit</a> <a href="#">Show</a>
5	Rattan	DOD01	Double Gentil	pcs	90000	180000	<a href="#">Edit</a> <a href="#">Show</a>
6	Rattan	OD02	Gentil Rattan	pcs	60000	150000	<a href="#">Edit</a> <a href="#">Show</a>
7	Rattan Shlefis	OD01	Gentil Shlefis	pcs	70000	140000	<a href="#">Edit</a> <a href="#">Show</a>
8	Rattan Shlefis	SL11	Sticker	pcs	75000	140000	<a href="#">Edit</a> <a href="#">Show</a>
9	Rattan Shlefis	OK02	Oval	pcs	90000	150000	<a href="#">Edit</a> <a href="#">Show</a>
10	Atta	AT01	Atta	pcs	80000	150000	<a href="#">Edit</a> <a href="#">Show</a>

Gambar 4.114 Tampilan Halaman *List Products* – Gudang

## 9. Tampilan *Form Product* – Gudang

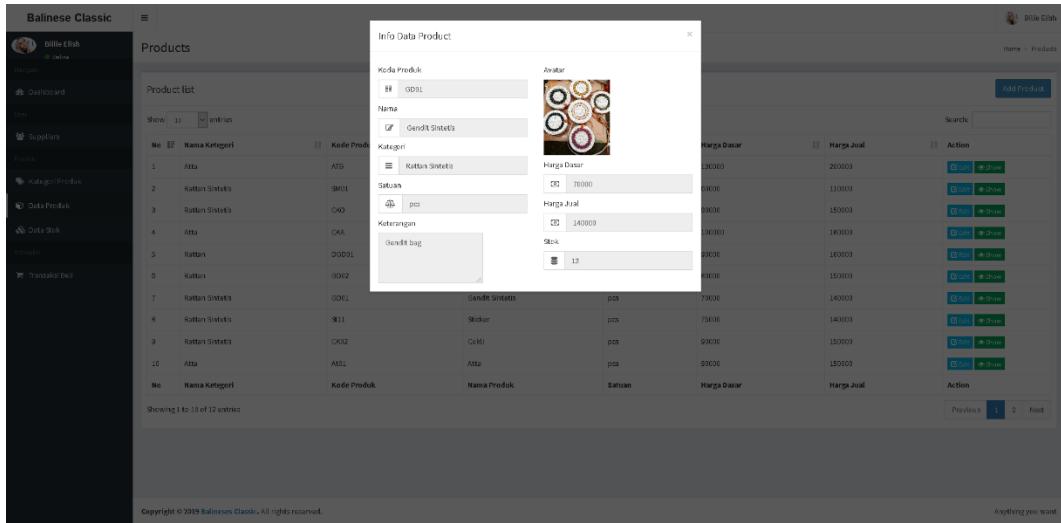
Halaman *List Products* merupakan halaman setelah klik menu *Products*. Pada halaman ini terdapat data-data *Products* yang tersimpan di sistem. Berikut adalah tampilan halaman *list Products*:

Add product	
Kode Produk	<input type="text"/>
Nama Produk	<input type="text"/>
Kategori	Select Product
Keterangan	<input type="text"/>
Satuan	Satuan
Harga Dasar	<input type="text"/>
Harga Jual	<input type="text"/>
Gambar	<input type="button" value="Browse..."/> No file selected.
	<a href="#">Tambah produk</a>
	<a href="#">Close</a>

Gambar 4.115 Tampilan *Form Product* – Gudang

## 10. Tampilan Detail *Product* – Gudang

Detail *Product* merupakan tampilan yang muncul setelah klik tombol *show* pada halaman *list Products*. Berikut adalah tampilan detail *Product*:



Gambar 4.116 Tampilan Detail Product – Gudang

## 11. Tampilan Halaman *List Stocks* – Gudang

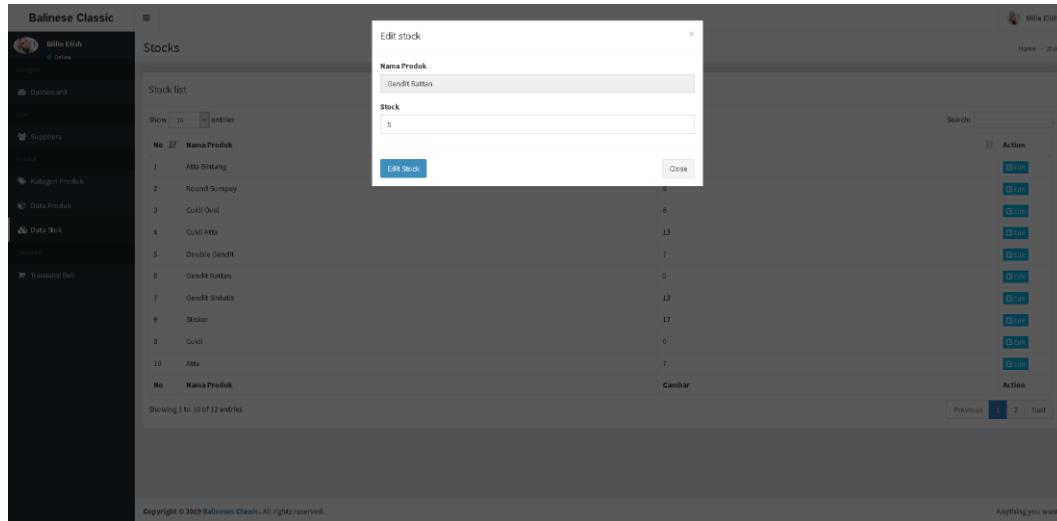
Halaman *List Stocks* merupakan halaman setelah klik menu *Stocks*. Pada halaman ini terdapat data-data *Stocks* yang tersimpan di sistem. Berikut adalah tampilan halaman *list Stocks*:

No	Nama Produk	Gambar	Action
1	Atta Bintang	7	
2	Round Sumpay	9	
3	Cukil Oval	8	
4	Cukil Atta	13	
5	Double Gendit	7	
6	Gendit Rattan	5	
7	Gendit Sintetis	13	
8	Sticker	17	
9	Cukil	0	
10	Atta	7	

Gambar 4.117 Tampilan Halaman *List Stocks* – Gudang

## 12. Tampilan *Form Stock* – Gudang

*Form Stock* merupakan *form* yang muncul setelah klik tombol *add Stock* pada halaman *list Stocks*. Berikut adalah tampilan *form Stock*:



Gambar 4.118 Tampilan Form Stock – Gudang

### 13. Tampilan Halaman *List* Pembelian – Gudang

Halaman *List* Pembelian merupakan halaman setelah klik menu transaksi beli. Pada halaman ini terdapat data-data transaksi beli yang tersimpan di sistem. Berikut adalah tampilan halaman *list* pembelian:

Pembelian					
Pembelian list					
No	Invoice Number	Tanggal pesan	Customer	Produk	Total
1	3048	2019-12-25	Ani	Gendit Sintetis   Gendit Rattan   Cukil Atta	\$750000
2	3047	2019-12-23	Tya	Cukil Oval   Round Sumpay   Atta Bintang	7350000
3	3045	2019-12-24	Ning	Gendit Sintetis   Gendit Rattan   Double Gendit   Cukil Atta	6800000
4	3045	2019-12-25	Risa	Rattan   Ball   Atta   Cukil   Sticker	7900000
5	3044	2019-12-02	Ning	Sticker	6000000
6	3043	2019-11-01	Putri	Rattan   Ball   Atta   Round Sumpay   Atta Bintang	15300000
7	3042	2019-10-27	elap	Cukil Oval   Round Sumpay   Atta Bintang	10010000
8	3041	2019-10-23	Putri	Cukil Atta	5000000
9	3040	2019-11-01	Risa	Rattan   Ball   Sticker	9825000
10	3039	2019-10-10	Putri	Gendit Rattan   Double Gendit	3650000

Gambar 4.119 Tampilan Halaman *List* Pembelian – Gudang

### 14. Tampilan Halaman *Form* Pembelian – Gudang

Pada halaman ini terdapat *form* yang wajib diisi saat melakukan transaksi. Tampilan *form* pembelian dapat dilihat pada gambar berikut:

Copyright © 2019 Balinese Classic. All rights reserved.

Anything you want.

Gambar 4.120 Tampilan Halaman *Form Pembelian – Gudang*

### 15. Tampilan Halaman *Invoice Pembelian – Gudang*

Tampilan Halaman *Invoice Pembelian* merupakan tampilan setelah klik tombol *invoice* pada halaman *list pembelian*. Tampilan Halaman *Invoice Pembelian* dapat dilihat pada gambar berikut:

Copyright © 2019 Balinese Classic. All rights reserved.

Anything you want.

Gambar 4.121 Tampilan Halaman *Invoice Pembelian – Gudang*

## 4.6 Pengujian Black Box

Pada pengujian Black Box ini ditunjukkan untuk menemukan *error* dari sistem yang telah dibangun. Adapun penjelasannya sebagai berikut:

### 4.6.1 Pengujian Halaman Admin

Berikut adalah hasil dari pengujian *Blackbox* yang telah dilakukan sesuai dengan rencana pengujian:

1. Pengujian pada Halaman *Login* – Admin

Pengujian dilakukan pada tiga sampel data *input* untuk menguji fungsionalitas tombol *Login*. Hasil pengujian halaman *login* dapat dilihat pada Tabel 4.15.

Tabel 4.15 Hasil Pengujian Halaman *Login* – Admin

No	Data <i>Input</i>	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Admin memasukkan email dan <i>password</i> dengan benar kemudian tekan tombol <i>Login</i> .	Proses autentikasi <i>user</i> berhasil dan diarahkan ke halaman <i>Dashboard</i>	Admin berhasil <i>login</i> dan diarahkan ke halaman <i>Dashboard</i>	Sesuai
2	Admin memasukkan email dan <i>password</i> yang salah kemudian tekan tombol <i>Login</i> .	Muncul pesan pemberitahuan <i>login gagal</i>	Admin tidak berhasil <i>login</i> dan muncul pesan pemberitahuan bahwa <i>login gagal</i>	Sesuai
3	Admin tidak memasukkan email dan <i>password</i> kemudian tekan tombol <i>Login</i> .	Muncul pemberitahuan untuk mengisi <i>field</i> yang masih kosong.	Admin tidak berhasil <i>login</i> dan muncul pemberitahuan untuk melengkapi <i>field</i> yang masih kosong.	Sesuai

2. Pengujian pada Navigasi Menu – Admin

Pengujian dilakukan pada sebelas menu untuk menguji fungsionalitasnya. Hasil pengujian navigasi menu dapat dilihat pada Tabel 4.16.

Tabel 4.16 Hasil Pengujian pada Navigasi Menu – Admin

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Klik menu <i>Dashboard</i> .	Menampilkan halaman <i>Dashboard</i> .	Admin berhasil diarahkan ke halaman <i>Dashboard</i> .	Sesuai
2	Klik menu <i>Users</i> .	Menampilkan halaman <i>Users</i> .	Admin berhasil diarahkan ke halaman <i>Users</i> .	Sesuai
3	Klik menu <i>Customers</i>	Menampilkan halaman <i>Customers</i> .	Admin berhasil diarahkan ke halaman <i>Customers</i> .	Sesuai
4	Klik menu <i>Suppliers</i> .	Menampilkan halaman <i>Suppliers</i> .	Admin berhasil diarahkan ke halaman <i>Suppliers</i> .	Sesuai
5	Klik menu <i>Categories</i> .	Menampilkan halaman <i>Categories</i> .	Admin berhasil diarahkan ke halaman <i>Categories</i> .	Sesuai
6	Klik menu <i>Products</i> .	Menampilkan halaman <i>Products</i> .	Admin berhasil diarahkan ke halaman <i>Products</i> .	Sesuai
7	Klik menu <i>Stocks</i> .	Menampilkan halaman <i>Stocks</i> .	Admin berhasil diarahkan ke halaman <i>Stocks</i> .	Sesuai
8	Klik menu Penjualan.	Menampilkan halaman Penjualan.	Admin berhasil diarahkan ke halaman Penjualan.	Sesuai
9	Klik menu Pembelian.	Menampilkan halaman Pembelian.	Admin berhasil diarahkan ke halaman Pembelian.	Sesuai
10	Klik menu Laporan Penjualan.	Menampilkan halaman	Admin berhasil diarahkan ke	Sesuai

		Laporan Penjualan.	halaman Laporan Penjualan.	
11	Klik menu Laporan Pembelian.	Menampilkan halaman Laporan Pembelian.	Admin berhasil diarahkan ke halaman Laporan Pembelian.	Sesuai

3. Pengujian pada Halaman *Users – Admin*

Pengujian dilakukan pada beberapa fungsi yang terdapat pada halaman *Users*. Hasil pengujian pada halaman *users* dapat dilihat pada Tabel 4.17.

Tabel 4.17 Pengujian pada Halaman *Users – Admin*

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Klik tombol <i>Add User</i>	Menampilkan modal <i>form</i> tambah <i>User</i> .	Berhasil membuka <i>form</i> untuk tambah data <i>User</i> .	Sesuai
2	Klik tombol <i>Edit</i>	Menampilkan modal <i>form edit user</i> .	Berhasil membuka <i>form</i> untuk <i>edit</i> data <i>User</i> .	Sesuai
3	Klik tombol <i>Show</i>	Menampilkan modal detail data <i>user</i> .	Berhasil membuka detail data dari <i>User</i> .	Sesuai
4	Pilih <i>select option Entries</i>	Menampilkan <i>list</i> data <i>User</i> sesuai pilihan.	Berhasil menampilkan banyaknya <i>list</i> data <i>User</i> sesuai pilihan.	Sesuai
5	Cari data pada <i>field Search</i>	Menampilkan data sesuai isi <i>field Search</i> .	Berhasil menampilkan data <i>Users</i> sesuai dengan isi <i>field Search</i> .	Sesuai
6	Klik tombol <i>Pagination</i>	Menampilkan data selanjutnya.	Berhasil menampilkan data sesuai dengan halaman <i>Pagination</i> .	Sesuai

4. Pengujian *Form Tambah* pada Halaman *Users – Admin*

Pengujian dilakukan pada tiga sampel data *input* untuk menguji fungsionalitas tombol *Tambah User*. Hasil pengujian *form tambah* pada halaman *users* dapat dilihat pada Tabel 4.18.

Tabel 4.18 Pengujian *Form Tambah* pada Halaman *Users – Admin*

No	Data <i>Input</i>	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Admin memasukkan data dengan benar kemudian tekan tombol <i>Tambah</i> .	Proses <i>insert</i> data berhasil dan <i>pop up form</i> tambah menghilang.	Admin berhasil memasukkan data <i>users</i> dan <i>pop up form</i> tambah menghilang.	Sesuai
2	Admin memasukkan data yang salah kemudian tekan tombol <i>Tambah</i> .	Muncul peringatan bahwa masih ada <i>field</i> yang salah.	Admin tidak berhasil memasukkan data <i>user</i> dan muncul peringatan bahwa masih ada <i>field</i> yang salah.	Sesuai
3	Admin tidak memasukkan data dengan lengkap pada <i>field</i> kemudian tekan tombol <i>Tambah</i> .	Muncul peringatan bahwa masih ada <i>field</i> yang kosong.	Admin tidak berhasil memasukkan data <i>user</i> dan muncul peringatan bahwa masih ada <i>field</i> yang kosong.	Sesuai

5. Pengujian *Form Edit* pada Halaman *Users – Admin*

Pengujian dilakukan pada dua sampel data *input* untuk menguji fungsionalitas tombol *Edit User*. Hasil pengujian *form edit* pada halaman *users* dapat dilihat pada Tabel 4.19.

Tabel 4.19 Pengujian *Form Edit* pada Halaman *Users – Admin*

No	Data <i>Input</i>	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
----	-------------------	-----------------------	------------------	------------

1	Admin memperbarui data dengan benar kemudian tekan tombol <i>Edit</i> .	Proses <i>edit</i> data berhasil dan <i>pop up form edit</i> menghilang.	Admin berhasil memperbarui data <i>users</i> dan <i>pop up form edit</i> menghilang.	Sesuai
2	Admin memperbarui data dengan salah kemudian tekan tombol <i>Edit</i> .	Muncul peringatan bahwa masih ada data yang salah.	Admin tidak berhasil memperbarui data <i>user</i> dan muncul peringatan bahwa masih ada data yang salah.	Sesuai

#### 6. Pengujian pada Halaman *Customers* – Admin

Pengujian dilakukan pada beberapa fungsi yang terdapat pada halaman *Customers*. Hasil pengujian pada halaman *customers* dapat dilihat pada Tabel 4.20.

Tabel 4.20 Pengujian pada Halaman *Customers* – Admin

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Klik tombol <i>Add Customer</i>	Menampilkan modal <i>form</i> tambah <i>Customer</i> .	Berhasil membuka <i>form</i> untuk tambah data <i>Customer</i> .	Sesuai
2	Klik tombol <i>Edit</i>	Menampilkan modal <i>form edit</i> <i>Customer</i> .	Berhasil membuka <i>form</i> untuk <i>edit</i> data <i>Customer</i> .	Sesuai
3	Klik tombol <i>Show</i>	Menampilkan modal detail data <i>Customer</i> .	Berhasil membuka detail data dari <i>Customer</i> .	Sesuai
4	Pilih <i>select option Entries</i>	Menampilkan <i>list</i> data <i>Customer</i> sesuai pilihan.	Berhasil menampilkan banyaknya <i>list</i> data <i>Customer</i> sesuai pilihan.	Sesuai

5	Cari data pada <i>field Search</i>	Menampilkan data sesuai isi <i>field Search.</i>	Berhasil menampilkan data <i>Customer</i> sesuai dengan isi <i>field Search.</i>	Sesuai
6	Klik tombol <i>Pagination</i>	Menampilkan data selanjutnya.	Berhasil menampilkan data sesuai dengan halaman <i>Pagination.</i>	Sesuai

7. Pengujian *Form Tambah* pada Halaman *Customers – Admin*

Pengujian dilakukan pada tiga sampel data *input* untuk menguji fungsionalitas tombol *Tambah Customer*. Hasil pengujian *form tambah* pada halaman *Customers* dapat dilihat pada Tabel 4.21.

Tabel 4.21 Pengujian *Form Tambah* pada Halaman *Customers – Admin*

No	Data Input	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Admin memasukkan data dengan benar kemudian tekan tombol <i>Tambah</i> .	Proses <i>insert</i> data berhasil dan <i>pop up form tambah</i> menghilang.	Admin berhasil memasukkan data <i>customer</i> dan <i>pop up form tambah</i> menghilang.	Sesuai
2	Admin memasukkan data yang salah kemudian tekan tombol <i>Tambah</i> .	Muncul peringatan bahwa masih ada <i>field</i> yang salah.	Admin tidak berhasil memasukkan data <i>customer</i> dan muncul peringatan bahwa masih ada <i>field</i> yang salah.	Sesuai
3	Admin tidak memasukkan data dengan lengkap pada <i>field</i> kemudian	Muncul peringatan bahwa masih ada <i>field</i> yang kosong.	Admin tidak berhasil memasukkan data <i>customer</i> dan muncul peringatan bahwa masih ada <i>field</i> yang kosong.	Sesuai

	tekan tombol Tambah.			
--	-------------------------	--	--	--

8. Pengujian *Form Edit* pada Halaman *Customers* – Admin

Pengujian dilakukan pada dua sampel data *input* untuk menguji fungsionalitas tombol *Edit customer*. Hasil pengujian *form edit* pada halaman *Customers* dapat dilihat pada Tabel 4.22.

Tabel 4.22 Pengujian *Form Edit* pada Halaman *Customers* – Admin

No	Data Input	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Admin memperbarui data dengan benar kemudian tekan tombol <i>Edit</i> .	Proses <i>edit</i> data berhasil dan <i>pop up form edit</i> menghilang.	Admin berhasil memperbarui data <i>customer</i> dan <i>pop up form edit</i> menghilang.	Sesuai
2	Admin memperbarui data dengan salah kemudian tekan tombol <i>Edit</i> .	Muncul peringatan bahwa masih ada data yang salah.	Admin tidak berhasil memperbarui data <i>customer</i> dan muncul peringatan bahwa masih ada data yang salah.	Sesuai

9. Pengujian pada Halaman *Suppliers* – Admin

Pengujian dilakukan pada beberapa fungsi yang terdapat pada halaman *Suppliers*. Hasil pengujian pada halaman *suppliers* dapat dilihat pada Tabel 4.23.

Tabel 4.23 Pengujian pada Halaman *Suppliers* – Admin

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Klik tombol <i>Add Supplier</i>	Menampilkan modal <i>form tambah Supplier</i> .	Berhasil membuka <i>form</i> untuk tambah data <i>Supplier</i> .	Sesuai

2	Klik tombol <i>Edit</i>	Menampilkan modal <i>form edit Supplier</i> .	Berhasil membuka <i>form</i> untuk <i>edit</i> data <i>Supplier</i> .	Sesuai
3	Klik tombol <i>Show</i>	Menampilkan modal detail data <i>Supplier</i> .	Berhasil membuka detail data dari <i>Supplier</i> .	Sesuai
4	Pilih <i>select option Entries</i>	Menampilkan <i>list</i> data <i>Supplier</i> sesuai pilihan.	Berhasil menampilkan banyaknya <i>list</i> data <i>Supplier</i> sesuai pilihan.	Sesuai
5	Cari data pada <i>field Search</i>	Menampilkan data sesuai isi <i>field Search</i> .	Berhasil menampilkan data <i>Supplier</i> sesuai dengan isi <i>field Search</i> .	Sesuai
6	Klik tombol <i>Pagination</i>	Menampilkan data selanjutnya.	Berhasil menampilkan data sesuai dengan halaman <i>Pagination</i> .	Sesuai

10. Pengujian *Form Tambah* pada Halaman *Suppliers* – Admin

Pengujian dilakukan pada tiga sampel data *input* untuk menguji fungsionalitas tombol Tambah *Supplier*. Hasil pengujian *form* tambah pada halaman *Suppliers* dapat dilihat pada Tabel 4.24.

Tabel 4.24 Pengujian *Form Tambah* pada Halaman *Suppliers* – Admin

No	Data Input	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Admin memasukkan data dengan benar kemudian tekan tombol Tambah.	Proses <i>insert</i> data berhasil dan <i>pop up form</i> tambah menghilang.	Admin berhasil memasukkan data <i>supplier</i> dan <i>pop up form</i> tambah menghilang.	Sesuai

2	Admin memasukkan data yang salah kemudian tekan tombol Tambah.	Muncul peringatan bahwa masih ada <i>field</i> yang salah.	Admin tidak berhasil memasukkan data <i>supplier</i> dan muncul peringatan bahwa masih ada <i>field</i> yang salah.	Sesuai
3	Admin tidak memasukkan data dengan lengkap pada <i>field</i> kemudian tekan tombol Tambah.	Muncul peringatan bahwa masih ada <i>field</i> yang kosong.	Admin tidak berhasil memasukkan data <i>supplier</i> dan muncul peringatan bahwa masih ada <i>field</i> yang kosong.	Sesuai

#### 11. Pengujian *Form Edit* pada Halaman *Suppliers* – Admin

Pengujian dilakukan pada dua sampel data *input* untuk menguji fungsionalitas tombol *Edit supplier*. Hasil pengujian *form edit* pada halaman *Suppliers* dapat dilihat pada Tabel 4.25.

Tabel 4.25 Pengujian *Form Edit* pada Halaman *Customers* – Admin

No	Data Input	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Admin memperbarui data dengan benar kemudian tekan tombol <i>Edit</i> .	Proses <i>edit</i> data berhasil dan <i>pop up form edit</i> menghilang.	Admin berhasil memperbarui data <i>supplier</i> dan <i>pop up form edit</i> menghilang.	Sesuai
2	Admin memperbarui data dengan salah kemudian tekan tombol <i>Edit</i> .	Muncul peringatan bahwa masih ada data yang salah.	Admin tidak berhasil memperbarui data <i>supplier</i> dan muncul peringatan bahwa masih ada data yang salah.	Sesuai

#### 12. Pengujian pada Halaman *Categories* – Admin

Pengujian dilakukan pada beberapa fungsi yang terdapat pada halaman *Categories*. Hasil pengujian pada halaman *categories* dapat dilihat pada Tabel 4.26.

Tabel 4.26 Pengujian pada Halaman *Categories* – Admin

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Klik tombol <i>Add Category</i>	Menampilkan modal <i>form</i> tambah <i>Category</i> .	Berhasil membuka <i>form</i> untuk tambah data <i>Category</i> .	Sesuai
2	Klik tombol <i>Edit</i>	Menampilkan modal <i>form edit Category</i> .	Berhasil membuka <i>form</i> untuk <i>edit</i> data <i>Category</i> .	Sesuai
3	Pilih <i>select option Entries</i>	Menampilkan <i>list</i> data <i>Category</i> sesuai pilihan.	Berhasil menampilkan banyaknya <i>list</i> data <i>Category</i> sesuai pilihan.	Sesuai
4	Cari data pada <i>field Search</i>	Menampilkan data sesuai isi <i>field Search</i> .	Berhasil menampilkan data <i>Category</i> sesuai dengan isi <i>field Search</i> .	Sesuai
5	Klik tombol <i>Pagination</i>	Menampilkan data selanjutnya.	Berhasil menampilkan data sesuai dengan halaman <i>Pagination</i> .	Sesuai

### 13. Pengujian *Form Tambah* pada Halaman *Categories* – Admin

Pengujian dilakukan pada tiga sampel data *input* untuk menguji fungsionalitas tombol Tambah *Category*. Hasil pengujian *form* tambah pada halaman *Categories* dapat dilihat pada Tabel 4.27.

Tabel 4.27 Pengujian *Form Tambah* pada Halaman *Categories* – Admin

No	Data <i>Input</i>	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan

1	Admin memasukkan data dengan benar kemudian tekan tombol Tambah.	Proses <i>insert</i> data berhasil dan <i>pop up form</i> tambah menghilang.	Admin berhasil memasukkan data <i>category</i> dan <i>pop up form</i> tambah menghilang.	Sesuai
2	Admin memasukkan data yang salah kemudian tekan tombol Tambah.	Muncul peringatan bahwa masih ada <i>field</i> yang salah.	Admin tidak berhasil memasukkan data <i>category</i> dan muncul peringatan bahwa masih ada <i>field</i> yang salah.	Sesuai
3	Admin tidak memasukkan data dengan lengkap pada <i>field</i> kemudian tekan tombol Tambah.	Muncul peringatan bahwa masih ada <i>field</i> yang kosong.	Admin tidak berhasil memasukkan data <i>category</i> dan muncul peringatan bahwa masih ada <i>field</i> yang kosong.	Sesuai

14. Pengujian *Form Edit* pada Halaman *Categories* – Admin

Pengujian dilakukan pada dua sampel data *input* untuk menguji fungsionalitas tombol *Edit category*. Hasil pengujian *form edit* pada halaman *Categories* dapat dilihat pada Tabel 4.28.

Tabel 4.28 Pengujian *Form Edit* pada Halaman *Categories* – Admin

No	Data Input	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Admin memperbarui data dengan benar kemudian tekan tombol <i>Edit</i> .	Proses <i>edit</i> data berhasil dan <i>pop up form edit</i> menghilang.	Admin berhasil memperbarui data <i>category</i> dan <i>pop up form edit</i> menghilang.	Sesuai
2	Admin memperbarui	Muncul peringatan	Admin tidak berhasil memperbarui	Sesuai

	data dengan salah kemudian tekan tombol <i>Edit.</i>	bawa masih ada data yang salah.	data <i>category</i> dan muncul peringatan bahwa masih ada data yang salah.	
--	--	---------------------------------	---	--

15. Pengujian pada Halaman *Products* – Admin

Pengujian dilakukan pada beberapa fungsi yang terdapat pada halaman *Products*. Hasil pengujian pada halaman *products* dapat dilihat pada Tabel 4.29.

Tabel 4.29 Pengujian pada Halaman *Products* – Admin

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Klik tombol <i>Add Product</i>	Menampilkan modal <i>form tambah Product.</i>	Berhasil membuka <i>form</i> untuk tambah data <i>Product.</i>	Sesuai
2	Klik tombol <i>Edit</i>	Menampilkan modal <i>form edit Product.</i>	Berhasil membuka <i>form</i> untuk <i>edit</i> data <i>Product.</i>	Sesuai
3	Pilih <i>select option Entries</i>	Menampilkan <i>list</i> data <i>Product</i> sesuai pilihan.	Berhasil menampilkan banyaknya <i>list</i> data <i>Product</i> sesuai pilihan.	Sesuai
4	Cari data pada <i>field Search</i>	Menampilkan data sesuai isi <i>field Search.</i>	Berhasil menampilkan data <i>Product</i> sesuai dengan isi <i>field Search.</i>	Sesuai
5	Klik tombol <i>Pagination</i>	Menampilkan data selanjutnya.	Berhasil menampilkan data sesuai dengan halaman <i>Pagination.</i>	Sesuai

16. Pengujian *Form Tambah* pada Halaman *Products* – Admin

Pengujian dilakukan pada tiga sampel data *input* untuk menguji fungsionalitas tombol Tambah *Product*. Hasil pengujian *form tambah* pada halaman *Products* dapat dilihat pada Tabel 4.30.

Tabel 4.30 Pengujian *Form Tambah* pada Halaman *Products – Admin*

No	Data Input	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Admin memasukkan data dengan benar kemudian tekan tombol Tambah.	Proses <i>insert</i> data berhasil dan <i>pop up form</i> tambah menghilang.	Admin berhasil memasukkan data <i>product</i> dan <i>pop up form</i> tambah menghilang.	Sesuai
2	Admin memasukkan data yang salah kemudian tekan tombol Tambah.	Muncul peringatan bahwa masih ada <i>field</i> yang salah.	Admin tidak berhasil memasukkan data <i>product</i> dan muncul peringatan bahwa masih ada <i>field</i> yang salah.	Sesuai
3	Admin tidak memasukkan data dengan lengkap pada <i>field</i> kemudian tekan tombol Tambah.	Muncul peringatan bahwa masih ada <i>field</i> yang kosong.	Admin tidak berhasil memasukkan data <i>product</i> dan muncul peringatan bahwa masih ada <i>field</i> yang kosong.	Sesuai

#### 17. Pengujian *Form Edit* pada Halaman *Products – Admin*

Pengujian dilakukan pada dua sampel data *input* untuk menguji fungsionalitas tombol *Edit product*. Hasil pengujian *form edit* pada halaman *Products* dapat dilihat pada Tabel 4.31.

Tabel 4.31 Pengujian *Form Edit* pada Halaman *Products – Admin*

No	Data Input	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Admin memperbarui	Proses <i>edit</i> data berhasil dan	Admin berhasil memperbarui	Sesuai

	data dengan benar kemudian tekan tombol <i>Edit</i> .	<i>pop up form edit</i> menghilang.	data <i>product</i> dan <i>pop up form edit</i> menghilang.	
2	Admin memperbarui data dengan salah kemudian tekan tombol <i>Edit</i> .	Muncul peringatan bahwa masih ada data yang salah.	Admin tidak berhasil memperbarui data <i>product</i> dan muncul peringatan bahwa masih ada data yang salah.	Sesuai

18. Pengujian pada Halaman *Stocks – Admin*

Pengujian dilakukan pada beberapa fungsi yang terdapat pada halaman *Stocks*. Hasil pengujian pada halaman *stock* dapat dilihat pada Tabel 4.32.

Tabel 4.32 Pengujian pada Halaman *Stocks – Admin*

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Klik tombol <i>Edit</i>	Menampilkan modal <i>form edit Stock</i> .	Berhasil membuka <i>form</i> untuk <i>edit</i> data <i>Stock</i> .	Sesuai
2	Pilih <i>select option Entries</i>	Menampilkan <i>list</i> data <i>Stock</i> sesuai pilihan.	Berhasil menampilkan banyaknya <i>list</i> data <i>Stock</i> sesuai pilihan.	Sesuai
3	Cari data pada <i>field Search</i>	Menampilkan data sesuai isi <i>field Search</i> .	Berhasil menampilkan data <i>Stock</i> sesuai dengan isi <i>field Search</i> .	Sesuai
4	Klik tombol <i>Pagination</i>	Menampilkan data selanjutnya.	Berhasil menampilkan data sesuai dengan halaman <i>Pagination</i> .	Sesuai

19. Pengujian *Form Edit* pada Halaman *Stocks – Admin*

Pengujian dilakukan pada dua sampel data *input* untuk menguji fungsionalitas tombol *Edit Stock*. Hasil pengujian *form edit* pada halaman Stocks dapat dilihat pada Tabel 4.33.

Tabel 4.33 Pengujian *Form Edit* pada Halaman Stocks – Admin

No	Data Input	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Admin memperbarui data dengan benar kemudian tekan tombol <i>Edit</i> .	Proses <i>edit</i> data berhasil dan <i>pop up form edit</i> menghilang.	Admin berhasil memperbarui data <i>stock</i> dan <i>pop up form edit</i> menghilang.	Sesuai
2	Admin memperbarui data dengan salah kemudian tekan tombol <i>Edit</i> .	Muncul peringatan bahwa masih ada data yang salah.	Admin tidak berhasil memperbarui data <i>stock</i> dan muncul peringatan bahwa masih ada data yang salah.	Sesuai

#### 20. Pengujian pada Halaman Penjualan – Admin

Pengujian dilakukan pada beberapa fungsi yang terdapat pada halaman Penjualan. Hasil pengujian pada halaman Penjualan dapat dilihat pada Tabel 4.34.

Tabel 4.34 Pengujian pada Halaman Penjualan – Admin

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Klik tombol <i>Add Penjualan</i>	Menampilkan Halaman tambah transaksi jual	Berhasil menampilkan Halaman tambah transaksi jual	Sesuai
2	Klik tombol <i>Edit</i>	Menampilkan Halaman <i>edit</i> transaksi jual	Berhasil menampilkan Halaman <i>edit</i> transaksi jual	Sesuai
3	Klik tombol <i>Invoice</i>	Menampilkan Halaman	Berhasil Menampilkan	Sesuai

		<i>invoice</i> transaksi jual	Halaman <i>invoice</i> transaksi jual	
4	Pilih <i>select option</i> Entries	Menampilkan <i>list</i> data Penjualan sesuai pilihan.	Berhasil menampilkan banyaknya <i>list</i> data Penjualan sesuai pilihan.	Sesuai
5	Cari data pada <i>field Search</i>	Menampilkan data sesuai isi <i>field Search</i> .	Berhasil menampilkan data Penjualan sesuai dengan isi <i>field Search</i> .	Sesuai
6	Klik tombol <i>Pagination</i>	Menampilkan data selanjutnya.	Berhasil menampilkan data sesuai dengan halaman <i>Pagination</i> .	Sesuai

#### 21. Pengujian pada Halaman Tambah Transaksi Jual – Admin

Pengujian dilakukan pada tiga sampel data *input* untuk menguji fungsionalitas tombol *Submit*. Hasil pengujian pada halaman Transaksi Jual dapat dilihat pada Tabel 4.35.

Tabel 4.35 Pengujian pada Halaman Tambah Transaksi Jual – Admin

No	Data <i>Input</i>	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Admin memasukkan data dengan benar kemudian tekan tombol <i>Submit</i> .	Proses <i>insert</i> data berhasil dan isi <i>field</i> direset.	Admin berhasil memasukkan data penjualan dan isi <i>field</i> direset.	Sesuai
2	Admin memasukkan data yang salah kemudian tekan tombol <i>Submit</i> .	Muncul peringatan bahwa masih ada <i>field</i> yang salah.	Admin tidak berhasil memasukkan data penjualan dan muncul peringatan	Sesuai

			bawa masih ada <i>field</i> yang salah.	
3	Admin tidak memasukkan data dengan lengkap pada <i>field</i> kemudian tekan tombol <i>Submit</i> .	Muncul peringatan bahwa masih ada <i>field</i> yang kosong.	Admin tidak berhasil memasukkan data penjualan dan muncul peringatan bahwa masih ada <i>field</i> yang kosong.	Sesuai

22. Pengujian pada Halaman *Edit* Transaksi Jual – Admin

Pengujian dilakukan pada dua sampel data *input* untuk menguji fungsionalitas tombol *Submit*. Hasil pengujian pada halaman *Edit* Transaksi Jual dapat dilihat pada Tabel 4.36.

Tabel 4.36 Pengujian pada Halaman *Edit* Transaksi Jual – Admin

No	Data <i>Input</i>	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Admin memperbarui data dengan benar kemudian tekan tombol <i>Submit</i> .	Proses <i>edit</i> data berhasil dan muncul peringatan berhasil	Admin berhasil memperbarui data penjualan dan muncul peringatan berhasil	Sesuai
2	Admin memperbarui data dengan salah kemudian tekan tombol <i>Submit</i> .	Muncul peringatan bahwa masih ada data yang salah.	Admin tidak berhasil memperbarui data penjualan dan muncul peringatan bahwa masih ada data yang salah.	Sesuai

23. Pengujian pada Halaman Pembelian – Admin

Pengujian dilakukan pada beberapa fungsi yang terdapat pada halaman Pembelian. Hasil pengujian pada halaman Pembelian dapat dilihat pada Tabel 4.37.

Tabel 4.37 Pengujian pada Halaman Pembelian – Admin

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Klik tombol <i>Add Pembelian</i>	Menampilkan Halaman tambah transaksi beli	Berhasil menampilkan Halaman tambah transaksi beli	Sesuai
2	Klik tombol <i>Edit</i>	Menampilkan Halaman <i>edit</i> transaksi beli	Berhasil menampilkan Halaman <i>edit</i> transaksi beli	Sesuai
3	Klik tombol <i>Invoice</i>	Menampilkan Halaman <i>invoice</i> transaksi beli	Berhasil Menampilkan Halaman <i>invoice</i> transaksi beli	Sesuai
4	Pilih <i>select option Entries</i>	Menampilkan <i>list</i> data Penjualan sesuai pilihan.	Berhasil menampilkan banyaknya <i>list</i> data Pembelian sesuai pilihan.	Sesuai
5	Cari data pada <i>field Search</i>	Menampilkan data sesuai isi <i>field Search</i> .	Berhasil menampilkan data Pembelian sesuai dengan isi <i>field Search</i> .	Sesuai
6	Klik tombol <i>Pagination</i>	Menampilkan data selanjutnya.	Berhasil menampilkan data sesuai dengan halaman <i>Pagination</i> .	Sesuai

24. Pengujian pada Halaman Tambah Transaksi Beli – Admin

Pengujian dilakukan pada tiga sampel data *input* untuk menguji fungsionalitas tombol *Submit*. Hasil pengujian pada halaman Tambah Transaksi Beli dapat dilihat pada Tabel 4.38.

Tabel 4.38 Pengujian pada Halaman Tambah Transaksi Beli – Admin

No	Data Input	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Admin memasukkan data dengan benar kemudian tekan tombol <i>Submit</i> .	Proses <i>insert</i> data berhasil dan isi <i>field</i> direset.	Admin berhasil memasukkan data pembelian dan isi <i>field</i> direset.	Sesuai
2	Admin memasukkan data yang salah kemudian tekan tombol <i>Submit</i> .	Muncul peringatan bahwa masih ada <i>field</i> yang salah.	Admin tidak berhasil memasukkan data pembelian dan muncul peringatan bahwa masih ada <i>field</i> yang salah.	Sesuai
3	Admin tidak memasukkan data dengan lengkap pada <i>field</i> kemudian tekan tombol <i>Submit</i> .	Muncul peringatan bahwa masih ada <i>field</i> yang kosong.	Admin tidak berhasil memasukkan data pembelian dan muncul peringatan bahwa masih ada <i>field</i> yang kosong.	Sesuai

25. Pengujian pada Halaman *Edit* Transaksi Beli – Admin

Pengujian dilakukan pada dua sampel data *input* untuk menguji fungsionalitas tombol *Submit*. Hasil pengujian pada halaman *Edit* Transaksi Beli dapat dilihat pada Tabel 4.39.

Tabel 4.39 Pengujian pada Halaman *Edit* Transaksi Beli – Admin

No	Data Input	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Admin memperbarui data dengan benar kemudian tekan tombol <i>Submit</i> .	Proses <i>edit</i> data berhasil dan muncul peringatan berhasil	Admin berhasil memperbarui data pembelian dan muncul peringatan berhasil	Sesuai

2	Admin memperbaharui data dengan salah kemudian tekan tombol <i>Submit.</i>	Muncul peringatan bahwa masih ada data yang salah.	Admin tidak berhasil memperbaharui data pembelian dan muncul peringatan bahwa masih ada data yang salah.	Sesuai
---	--	--	--	--------

26. Pengujian pada Halaman Laporan Penjualan – Admin

Pengujian dilakukan pada beberapa fungsi yang terdapat pada halaman Laporan Penjualan. Hasil pengujian pada halaman Laporan Penjualan dapat dilihat pada Tabel 4.40.

Tabel 4.40 Pengujian pada Halaman Laporan Penjualan – Admin

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Pilih tahun dan klik tombol <i>Submit.</i>	Menampilkan data grafik sesuai dengan tahun yang dipilih.	Berhasil Menampilkan data grafik sesuai dengan tahun yang dipilih.	Sesuai
2	Pilih tanggal dan klik tombol <i>Filter.</i>	Menampilkan data <i>grid</i> sesuai dengan tanggal yang dipilih	Berhasil menampilkan data <i>grid</i> sesuai dengan tanggal yang dipilih	Sesuai
3	Klik tombol <i>copy</i>	<i>Copy</i> data <i>grid</i>	<i>Copy</i> data <i>grid</i>	Sesuai
4	Klik tombol <i>csv</i>	<i>Download</i> data <i>grid</i> bertipe <i>csv</i>	Berhasil <i>Download</i> data <i>grid</i> bertipe <i>csv</i>	Sesuai
5	Klik tombol <i>excel</i>	<i>Download</i> data <i>grid</i> bertipe <i>xlsx</i>	Berhasil <i>Download</i> data <i>grid</i> bertipe <i>xlsx</i>	Sesuai
6	Klik tombol <i>print</i>	<i>Print</i> data <i>grid</i>		Sesuai
7	Klik tombol <i>Add Penjualan</i>	Menampilkan Halaman tambah transaksi jual	Berhasil menampilkan Halaman tambah transaksi jual	Sesuai

8	Klik tombol <i>Edit</i>	Menampilkan Halaman <i>edit</i> transaksi jual	Berhasil menampilkan Halaman <i>edit</i> transaksi jual	Sesuai
9	Klik tombol <i>Invoice</i>	Menampilkan Halaman <i>invoice</i> transaksi jual	Berhasil Menampilkan Halaman <i>invoice</i> transaksi jual	Sesuai
10	Pilih <i>select option Entries</i>	Menampilkan <i>list data Penjualan</i> sesuai pilihan.	Berhasil menampilkan banyaknya <i>list data Penjualan</i> sesuai pilihan.	Sesuai
11	Cari data pada <i>field Search</i>	Menampilkan data sesuai isi <i>field Search.</i>	Berhasil menampilkan data Penjualan sesuai dengan isi <i>field Search.</i>	Sesuai
12	Klik tombol <i>Pagination</i>	Menampilkan data selanjutnya.	Berhasil menampilkan data sesuai dengan halaman <i>Pagination.</i>	Sesuai

#### 27. Pengujian pada Halaman Laporan Pembelian – Admin

Pengujian dilakukan pada beberapa fungsi yang terdapat pada halaman Laporan Pembelian. Hasil pengujian pada halaman Laporan Pembelian dapat dilihat pada Tabel 4.41.

Tabel 4.41 Pengujian pada Halaman Laporan Pembelian – Admin

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Pilih tahun dan klik tombol Submit.	Menampilkan data grafik sesuai dengan tahun yang dipilih.	Berhasil Menampilkan data grafik sesuai dengan tahun yang dipilih.	Sesuai

2	Pilih tanggal dan klik tombol Filter.	Menampilkan data <i>grid</i> sesuai dengan tanggal yang dipilih	Berhasil menampilkan data <i>grid</i> sesuai dengan tanggal yang dipilih	Sesuai
3	Klik tombol copy	Copy data <i>grid</i>	Copy data <i>grid</i>	Sesuai
4	Klik tombol csv	Download data <i>grid</i> bertipe csv	Berhasil Download data <i>grid</i> bertipe csv	Sesuai
5	Klik tombol excel	Download data <i>grid</i> bertipe xlsx	Berhasil Download data <i>grid</i> bertipe xlsx	Sesuai
6	Klik tombol print	Print data <i>grid</i>		Sesuai
7	Klik tombol Add Pembelian	Menampilkan Halaman tambah transaksi beli	Berhasil menampilkan Halaman tambah transaksi beli	Sesuai
8	Klik tombol <i>Edit</i>	Menampilkan Halaman <i>edit</i> transaksi beli	Berhasil menampilkan Halaman <i>edit</i> transaksi beli	Sesuai
9	Klik tombol <i>Invoice</i>	Menampilkan Halaman <i>invoice</i> transaksi beli	Berhasil Menampilkan Halaman <i>invoice</i> transaksi beli	Sesuai
10	Pilih <i>select option Entries</i>	Menampilkan <i>list</i> data Pembelian sesuai pilihan.	Berhasil menampilkan banyaknya <i>list</i> data Pembelian sesuai pilihan.	Sesuai
11	Cari data pada <i>field Search</i>	Menampilkan data sesuai isi <i>field Search</i> .	Berhasil menampilkan data Pembelian sesuai dengan isi <i>field Search</i> .	Sesuai

12	Klik tombol <i>Pagination</i>	Menampilkan data selanjutnya.	Berhasil menampilkan data sesuai dengan halaman <i>Pagination</i> .	Sesuai
----	-------------------------------	-------------------------------	---	--------

#### 4.6.2 Pengujian Halaman Kasir

Berikut adalah hasil dari pengujian *Blackbox* yang telah dilakukan sesuai dengan rencana pengujian:

1. Pengujian pada Halaman *Login* – Kasir

Pengujian dilakukan pada tiga sampel data *input* untuk menguji fungsionalitas tombol *Login*. Hasil pengujian halaman *login* dapat dilihat pada Tabel 4.42.

Tabel 4.42 Hasil Pengujian Halaman *Login* – Kasir

No	Data <i>Input</i>	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Kasir memasukkan email dan <i>password</i> dengan benar kemudian tekan tombol <i>Login</i> .	Proses autentikasi <i>user</i> berhasil dan diarahkan ke halaman <i>Dashboard</i>	Kasir berhasil <i>login</i> dan diarahkan ke halaman <i>Dashboard</i>	Sesuai
2	Kasir memasukkan email dan <i>password</i> yang salah kemudian tekan tombol <i>Login</i> .	Muncul pesan pemberitahuan <i>login</i> gagal	Kasir tidak berhasil <i>login</i> dan muncul pesan pemberitahuan bahwa <i>login</i> gagal	Sesuai
3	Kasir tidak memasukkan email dan <i>password</i>	Muncul pemberitahuan untuk mengisi <i>field</i> yang masih kosong.	Kasir tidak berhasil <i>login</i> dan muncul pemberitahuan untuk melengkapi	Sesuai

	kemudian tekan tombol <i>Login</i> .		<i>field</i> yang masih kosong.	
--	--------------------------------------	--	---------------------------------	--

2. Pengujian pada Navigasi Menu – Kasir

Pengujian dilakukan pada sebelas menu untuk menguji fungsionalitasnya.

Hasil pengujian navigasi menu dapat dilihat pada Tabel 4.43.

Tabel 4.43 Hasil Pengujian pada Navigasi Menu – Kasir

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Klik menu <i>Dashboard</i> .	Menampilkan halaman <i>Dashboard</i> .	Kasir berhasil diarahkan ke halaman <i>Dashboard</i> .	Sesuai
2	Klik menu <i>Customers</i>	Menampilkan halaman <i>Customers</i> .	Kasir berhasil diarahkan ke halaman <i>Customers</i> .	Sesuai
3	Klik menu <i>Categories</i> .	Menampilkan halaman <i>Categories</i> .	Kasir berhasil diarahkan ke halaman <i>Categories</i> .	Sesuai
4	Klik menu <i>Products</i> .	Menampilkan halaman <i>Products</i> .	Kasir berhasil diarahkan ke halaman <i>Products</i> .	Sesuai
5	Klik menu Penjualan.	Menampilkan halaman Penjualan.	Kasir berhasil diarahkan ke halaman Penjualan.	Sesuai

3. Pengujian pada Halaman *Customers* – Kasir

Pengujian dilakukan pada beberapa fungsi yang terdapat pada halaman *Customers*. Hasil pengujian pada halaman *customers* dapat dilihat pada Tabel 4.44.

Tabel 4.44 Pengujian pada Halaman *Customers* – Kasir

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
----	--------------------	-----------------------	------------------	------------

1	Klik tombol <i>Add Customer</i>	Menampilkan modal <i>form tambah Customer.</i>	Berhasil membuka <i>form</i> untuk tambah data <i>Customer.</i>	Sesuai
2	Klik tombol <i>Edit</i>	Menampilkan modal <i>form edit Customer.</i>	Berhasil membuka <i>form</i> untuk <i>edit</i> data <i>Customer.</i>	Sesuai
3	Klik tombol <i>Show</i>	Menampilkan modal detail data dari <i>Customer.</i>	Berhasil membuka detail data dari <i>Customer.</i>	Sesuai
4	Pilih <i>select option Entries</i>	Menampilkan <i>list</i> data <i>Customer</i> sesuai pilihan.	Berhasil menampilkan banyaknya <i>list</i> data <i>Customer</i> sesuai pilihan.	Sesuai
5	Cari data pada <i>field Search</i>	Menampilkan data sesuai isi <i>field Search.</i>	Berhasil menampilkan data <i>Customer</i> sesuai dengan isi <i>field Search.</i>	Sesuai
6	Klik tombol <i>Pagination</i>	Menampilkan data selanjutnya.	Berhasil menampilkan data sesuai dengan halaman <i>Pagination.</i>	Sesuai

4. Pengujian *Form Tambah* pada Halaman *Customers – Kasir*

Pengujian dilakukan pada tiga sampel data *input* untuk menguji fungsionalitas tombol *Tambah Customer*. Hasil pengujian *form* tambah pada halaman *Customers* dapat dilihat pada Tabel 4.45.

Tabel 4.45 Pengujian *Form Tambah* pada Halaman *Customers – Kasir*

No	Data Input	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Kasir memasukkan data dengan	Proses <i>insert</i> data berhasil dan <i>pop up</i>	Kasir berhasil memasukkan data <i>customer</i> dan <i>pop</i>	Sesuai

	benar kemudian tekan tombol Tambah.	<i>form tambah</i> menghilang.	<i>up form tambah</i> menghilang.	
2	Kasir memasukkan data yang salah kemudian tekan tombol Tambah.	Muncul peringatan bahwa masih ada <i>field</i> yang salah.	Kasir tidak berhasil memasukkan data <i>customer</i> dan muncul peringatan bahwa masih ada <i>field</i> yang salah.	Sesuai
3	Kasir tidak memasukkan data dengan lengkap pada <i>field</i> kemudian tekan tombol Tambah.	Muncul peringatan bahwa masih ada <i>field</i> yang kosong.	Kasir tidak berhasil memasukkan data <i>customer</i> dan muncul peringatan bahwa masih ada <i>field</i> yang kosong.	Sesuai

##### 5. Pengujian *Form Edit* pada Halaman *Customers* – Kasir

Pengujian dilakukan pada dua sampel data *input* untuk menguji fungsionalitas tombol *Edit customer*. Hasil pengujian *form edit* pada halaman *Customers* dapat dilihat pada Tabel 4.46.

Tabel 4.46 Pengujian *Form Edit* pada Halaman *Customers* – Kasir

No	Data <i>Input</i>	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Kasir memperbarui data dengan benar kemudian tekan tombol <i>Edit</i> .	Proses <i>edit</i> data berhasil dan <i>pop up form edit</i> menghilang.	Kasir berhasil memperbarui data <i>customer</i> dan <i>pop up form edit</i> menghilang.	Sesuai
2	Kasir memperbarui data dengan salah kemudian	Muncul peringatan bahwa masih ada data yang salah.	Kasir tidak berhasil memperbarui data <i>customer</i> dan muncul peringatan	Sesuai

	tekan tombol <i>Edit.</i>		bawa masih ada data yang salah.	
--	------------------------------	--	------------------------------------	--

6. Pengujian pada Halaman *Categories* – Kasir

Pengujian dilakukan pada beberapa fungsi yang terdapat pada halaman *Categories*. Hasil pengujian pada halaman *categories* dapat dilihat pada Tabel 4.47.

Tabel 4.47 Pengujian pada Halaman *Categories* – Kasir

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Klik tombol <i>Add Category</i>	Menampilkan modal <i>form</i> tambah <i>Category</i> .	Berhasil membuka <i>form</i> untuk tambah data <i>Category</i> .	Sesuai
2	Klik tombol <i>Edit</i>	Menampilkan modal <i>form edit</i> <i>Category</i> .	Berhasil membuka <i>form</i> untuk <i>edit</i> data <i>Category</i> .	Sesuai
3	Pilih <i>select option Entries</i>	Menampilkan <i>list data Category</i> sesuai pilihan.	Berhasil menampilkan banyaknya <i>list data Category</i> sesuai pilihan.	Sesuai
4	Cari data pada <i>field Search</i>	Menampilkan data sesuai isi <i>field Search</i> .	Berhasil menampilkan data <i>Category</i> sesuai dengan isi <i>field Search</i> .	Sesuai
5	Klik tombol <i>Pagination</i>	Menampilkan data selanjutnya.	Berhasil menampilkan data sesuai dengan halaman <i>Pagination</i> .	Sesuai

7. Pengujian *Form Tambah* pada Halaman *Categories* – Kasir

Pengujian dilakukan pada tiga sampel data *input* untuk menguji fungsionalitas tombol Tambah *Category*. Hasil pengujian *form tambah* pada halaman *Categories* dapat dilihat pada Tabel 4.48.

Tabel 4.48 Pengujian *Form Tambah* pada Halaman *Categories* – Kasir

No	Data Input	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Kasir memasukkan data dengan benar kemudian tekan tombol Tambah.	Proses <i>insert</i> data berhasil dan <i>pop up form</i> tambah menghilang.	Kasir berhasil memasukkan data <i>category</i> dan <i>pop up form</i> tambah menghilang.	Sesuai
2	Kasir memasukkan data yang salah kemudian tekan tombol Tambah.	Muncul peringatan bahwa masih ada <i>field</i> yang salah.	Kasir tidak berhasil memasukkan data <i>category</i> dan muncul peringatan bahwa masih ada <i>field</i> yang salah.	Sesuai
3	Kasir tidak memasukkan data dengan lengkap pada <i>field</i> kemudian tekan tombol Tambah.	Muncul peringatan bahwa masih ada <i>field</i> yang kosong.	Kasir tidak berhasil memasukkan data <i>category</i> dan muncul peringatan bahwa masih ada <i>field</i> yang kosong.	Sesuai

#### 8. Pengujian *Form Edit* pada Halaman *Categories* – Kasir

Pengujian dilakukan pada dua sampel data *input* untuk menguji fungsionalitas tombol *Edit category*. Hasil pengujian *form edit* pada halaman *Categories* dapat dilihat pada Tabel 4.49.

Tabel 4.49 Pengujian *Form Edit* pada Halaman *Categories* – Kasir

No	Data Input	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Kasir memperbarui	Proses <i>edit</i> data berhasil dan	Kasir berhasil memperbarui	Sesuai

	data dengan benar kemudian tekan tombol <i>Edit</i> .	<i>pop up form edit</i> menghilang.	data <i>category</i> dan <i>pop up form edit</i> menghilang.	
2	Kasir memperbarui data dengan salah kemudian tekan tombol <i>Edit</i> .	Muncul peringatan bahwa masih ada data yang salah.	Kasir tidak berhasil memperbarui data <i>category</i> dan muncul peringatan bahwa masih ada data yang salah.	Sesuai

#### 9. Pengujian pada Halaman *Products* – Kasir

Pengujian dilakukan pada beberapa fungsi yang terdapat pada halaman *Products*. Hasil pengujian pada halaman *products* dapat dilihat pada Tabel 4.50.

Tabel 4.50 Pengujian pada Halaman *Products* – Kasir

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Klik tombol <i>Add Product</i>	Menampilkan modal <i>form</i> tambah <i>Product</i> .	Berhasil membuka <i>form</i> untuk tambah data <i>Product</i> .	Sesuai
2	Klik tombol <i>Edit</i>	Menampilkan modal <i>form edit</i> <i>Product</i> .	Berhasil membuka <i>form</i> untuk <i>edit</i> data <i>Product</i> .	Sesuai
3	Pilih <i>select option Entries</i>	Menampilkan <i>list</i> data <i>Product</i> sesuai pilihan.	Berhasil menampilkan banyaknya <i>list</i> data <i>Product</i> sesuai pilihan.	Sesuai
4	Cari data pada <i>field Search</i>	Menampilkan data sesuai isi <i>field Search</i> .	Berhasil menampilkan data <i>Product</i> sesuai dengan isi <i>field Search</i> .	Sesuai

5	Klik tombol <i>Pagination</i>	Menampilkan data selanjutnya.	Berhasil menampilkan data sesuai dengan halaman <i>Pagination</i> .	Sesuai
---	-------------------------------	-------------------------------	---	--------

10. Pengujian *Form Tambah* pada Halaman *Products – Kasir*

Pengujian dilakukan pada tiga sampel data *input* untuk menguji fungsionalitas tombol Tambah *Product*. Hasil pengujian *form* tambah pada halaman *Products* dapat dilihat pada Tabel 4.51.

Tabel 4.51 Pengujian *Form Tambah* pada Halaman *Products – Kasir*

No	Data Input	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Kasir memasukkan data dengan benar kemudian tekan tombol Tambah.	Proses <i>insert</i> data berhasil dan <i>pop up form</i> tambah menghilang.	Kasir berhasil memasukkan data <i>product</i> dan <i>pop up form</i> tambah menghilang.	Sesuai
2	Kasir memasukkan data yang salah kemudian tekan tombol Tambah.	Muncul peringatan bahwa masih ada <i>field</i> yang salah.	Kasir tidak berhasil memasukkan data <i>product</i> dan muncul peringatan bahwa masih ada <i>field</i> yang salah.	Sesuai
3	Kasir tidak memasukkan data dengan lengkap pada <i>field</i> kemudian tekan tombol Tambah.	Muncul peringatan bahwa masih ada <i>field</i> yang kosong.	Kasir tidak berhasil memasukkan data <i>product</i> dan muncul peringatan bahwa masih ada <i>field</i> yang kosong.	Sesuai

11. Pengujian *Form Edit* pada Halaman *Products – Kasir*

Pengujian dilakukan pada dua sampel data *input* untuk menguji fungsionalitas tombol *Edit product*. Hasil pengujian *form edit* pada halaman *Products* dapat dilihat pada Tabel 4.52.

Tabel 4.52 Pengujian *Form Edit* pada Halaman *Products* – Kasir

No	Data Input	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Kasir memperbarui data dengan benar kemudian tekan tombol <i>Edit</i> .	Proses <i>edit</i> data berhasil dan <i>pop up form edit</i> menghilang.	Kasir berhasil memperbarui data <i>product</i> dan <i>pop up form edit</i> menghilang.	Sesuai
2	Kasir memperbarui data dengan salah kemudian tekan tombol <i>Edit</i> .	Muncul peringatan bahwa masih ada data yang salah.	Kasir tidak berhasil memperbarui data <i>product</i> dan muncul peringatan bahwa masih ada data yang salah.	Sesuai

#### 12. Pengujian pada Halaman Penjualan – Kasir

Pengujian dilakukan pada beberapa fungsi yang terdapat pada halaman Penjualan. Hasil pengujian pada halaman Penjualan dapat dilihat pada Tabel 4.53.

Tabel 4.53 Pengujian pada Halaman Penjualan – Kasir

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Klik tombol <i>Add Penjualan</i>	Menampilkan Halaman tambah transaksi jual	Berhasil menampilkan Halaman tambah transaksi jual	Sesuai
2	Klik tombol <i>Edit</i>	Menampilkan Halaman <i>edit</i> transaksi jual	Berhasil menampilkan Halaman <i>edit</i> transaksi jual	Sesuai
3	Klik tombol <i>Invoice</i>	Menampilkan Halaman	Berhasil Menampilkan	Sesuai

		<i>invoice</i> transaksi jual	Halaman <i>invoice</i> transaksi jual	
4	Pilih <i>select option</i> Entries	Menampilkan <i>list</i> data Penjualan sesuai pilihan.	Berhasil menampilkan banyaknya <i>list</i> data Penjualan sesuai pilihan.	Sesuai
5	Cari data pada <i>field Search</i>	Menampilkan data sesuai isi <i>field Search</i> .	Berhasil menampilkan data Penjualan sesuai dengan isi <i>field Search</i> .	Sesuai
6	Klik tombol <i>Pagination</i>	Menampilkan data selanjutnya.	Berhasil menampilkan data sesuai dengan halaman <i>Pagination</i> .	Sesuai

### 13. Pengujian pada Halaman Tambah Transaksi Jual – Kasir

Pengujian dilakukan pada tiga sampel data *input* untuk menguji fungsionalitas tombol *Submit*. Hasil pengujian pada halaman Transaksi Jual dapat dilihat pada Tabel 4.54.

Tabel 4.54 Pengujian pada Halaman Tambah Transaksi Jual – Kasir

No	Data <i>Input</i>	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Kasir memasukkan data dengan benar kemudian tekan tombol <i>Submit</i> .	Proses <i>insert</i> data berhasil dan isi <i>field</i> direset.	Kasir berhasil memasukkan data penjualan dan isi <i>field</i> direset.	Sesuai
2	Kasir memasukkan data yang salah kemudian tekan tombol <i>Submit</i> .	Muncul peringatan bahwa masih ada <i>field</i> yang salah.	Kasir tidak berhasil memasukkan data penjualan dan muncul peringatan	Sesuai

			bawa masih ada <i>field</i> yang salah.	
3	Kasir tidak memasukkan data dengan lengkap pada <i>field</i> kemudian tekan tombol <i>Submit</i> .	Muncul peringatan bahwa masih ada <i>field</i> yang kosong.	Kasir tidak berhasil memasukkan data penjualan dan muncul peringatan bahwa masih ada <i>field</i> yang kosong.	Sesuai

14. Pengujian pada Halaman *Edit* Transaksi Jual – Kasir

Pengujian dilakukan pada dua sampel data *input* untuk menguji fungsionalitas tombol *Submit*. Hasil pengujian pada halaman *Edit* Transaksi Jual dapat dilihat pada Tabel 4.55.

Tabel 4.55 Pengujian pada Halaman *Edit* Transaksi Jual – Kasir

No	Data <i>Input</i>	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Kasir memperbarui data dengan benar kemudian tekan tombol <i>Submit</i> .	Proses <i>edit</i> data berhasil dan muncul peringatan berhasil	Kasir berhasil memperbarui data penjualan dan muncul peringatan berhasil	Sesuai
2	Kasir memperbarui data dengan salah kemudian tekan tombol <i>Submit</i> .	Muncul peringatan bahwa masih ada data yang salah.	Kasir tidak berhasil memperbarui data penjualan dan muncul peringatan bahwa masih ada data yang salah.	Sesuai

#### 4.6.3 Pengujian Halama Gudang

Berikut adalah hasil dari pengujian *Blackbox* yang telah dilakukan sesuai dengan rencana pengujian:

1. Pengujian pada Halaman *Login* – Gudang

Pengujian dilakukan pada tiga sampel data *input* untuk menguji fungsionalitas tombol *Login*. Hasil pengujian halaman *login* dapat dilihat pada Tabel 4.56.

Tabel 4.56 Hasil Pengujian Halaman *Login* – Gudang

No	Data <i>Input</i>	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Gudang memasukkan email dan <i>password</i> dengan benar kemudian tekan tombol <i>Login</i> .	Proses autentikasi <i>user</i> berhasil dan diarahkan ke halaman <i>Dashboard</i>	Gudang berhasil <i>login</i> dan diarahkan ke halaman <i>Dashboard</i>	Sesuai
2	Gudang memasukkan email dan <i>password</i> yang salah kemudian tekan tombol <i>Login</i> .	Muncul pesan pemberitahuan <i>login gagal</i>	Gudang tidak berhasil <i>login</i> dan muncul pesan pemberitahuan bahwa <i>login gagal</i>	Sesuai
3	Gudang tidak memasukkan email dan <i>password</i> kemudian tekan tombol <i>Login</i> .	Muncul pemberitahuan untuk mengisi <i>field</i> yang masih kosong.	Gudang tidak berhasil <i>login</i> dan muncul pemberitahuan untuk melengkapi <i>field</i> yang masih kosong.	Sesuai

## 2. Pengujian pada Navigasi Menu – Gudang

Pengujian dilakukan pada sebelas menu untuk menguji fungsionalitasnya.

Hasil pengujian navigasi menu dapat dilihat pada Tabel 4.57.

Tabel 4.57 Hasil Pengujian pada Navigasi Menu – Gudang

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan

1	Klik menu <i>Dashboard</i> .	Menampilkan halaman <i>Dashboard</i> .	Gudang berhasil diarahkan ke halaman <i>Dashboard</i> .	Sesuai
2	Klik menu <i>Suppliers</i> .	Menampilkan halaman <i>Suppliers</i> .	Gudang berhasil diarahkan ke halaman <i>Suppliers</i> .	Sesuai
3	Klik menu <i>Categories</i> .	Menampilkan halaman <i>Categories</i> .	Gudang berhasil diarahkan ke halaman <i>Categories</i> .	Sesuai
4	Klik menu <i>Products</i> .	Menampilkan halaman <i>Products</i> .	Gudang berhasil diarahkan ke halaman <i>Products</i> .	Sesuai
5	Klik menu <i>Stocks</i> .	Menampilkan halaman <i>Stocks</i> .	Gudang berhasil diarahkan ke halaman <i>Stocks</i> .	Sesuai
6	Klik menu Pembelian.	Menampilkan halaman Pembelian.	Gudang berhasil diarahkan ke halaman Pembelian.	Sesuai

### 3. Pengujian pada Halaman *Suppliers* – Gudang

Pengujian dilakukan pada beberapa fungsi yang terdapat pada halaman *Suppliers*. Hasil pengujian pada halaman *suppliers* dapat dilihat pada Tabel 4.58.

Tabel 4.58 Pengujian pada Halaman *Suppliers* – Gudang

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Klik tombol <i>Add Supplier</i>	Menampilkan modal <i>form tambah Supplier</i> .	Berhasil membuka <i>form</i> untuk tambah data <i>Supplier</i> .	Sesuai
2	Klik tombol <i>Edit</i>	Menampilkan modal <i>form edit Supplier</i> .	Berhasil membuka <i>form</i> untuk <i>edit</i> data <i>Supplier</i> .	Sesuai

3	Klik tombol <i>Show</i>	Menampilkan modal detail data <i>Supplier</i> .	Berhasil membuka detail data dari <i>Supplier</i> .	Sesuai
4	Pilih <i>select option Entries</i>	Menampilkan <i>list</i> data <i>Supplier</i> sesuai pilihan.	Berhasil menampilkan banyaknya <i>list</i> data <i>Supplier</i> sesuai pilihan.	Sesuai
5	Cari data pada <i>field Search</i>	Menampilkan data sesuai isi <i>field Search</i> .	Berhasil menampilkan data <i>Supplier</i> sesuai dengan isi <i>field Search</i> .	Sesuai
6	Klik tombol <i>Pagination</i>	Menampilkan data selanjutnya.	Berhasil menampilkan data sesuai dengan halaman <i>Pagination</i> .	Sesuai

#### 4. Pengujian *Form Tambah* pada Halaman *Suppliers* – Gudang

Pengujian dilakukan pada tiga sampel data *input* untuk menguji fungsionalitas tombol Tambah *Supplier*. Hasil pengujian *form* tambah pada halaman *Suppliers* dapat dilihat pada Tabel 4.59.

Tabel 4.59 Pengujian *Form Tambah* pada Halaman *Suppliers* – Gudang

No	Data <i>Input</i>	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Gudang memasukkan data dengan benar kemudian tekan tombol Tambah.	Proses <i>insert</i> data berhasil dan <i>pop up form</i> tambah menghilang.	Gudang berhasil memasukkan data <i>supplier</i> dan <i>pop up form</i> tambah menghilang.	Sesuai
2	Gudang memasukkan data yang salah	Muncul peringatan bahwa masih	Gudang tidak berhasil memasukkan data <i>supplier</i> dan muncul	Sesuai

	kemudian tekan tombol Tambah.	ada <i>field</i> yang salah.	peringatan bahwa masih ada <i>field</i> yang salah.	
3	Gudang tidak memasukkan data dengan lengkap pada <i>field</i> kemudian tekan tombol Tambah.	Muncul peringatan bahwa masih ada <i>field</i> yang kosong.	Gudang tidak berhasil memasukkan data <i>supplier</i> dan muncul peringatan bahwa masih ada <i>field</i> yang kosong.	Sesuai

5. Pengujian *Form Edit* pada Halaman *Suppliers* – Gudang

Pengujian dilakukan pada dua sampel data *input* untuk menguji fungsionalitas tombol *Edit supplier*. Hasil pengujian *form edit* pada halaman *Suppliers* dapat dilihat pada Tabel 4.60.

Tabel 4.60 Pengujian *Form Edit* pada Halaman *Customers* – Gudang

No	Data <i>Input</i>	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Gudang memperbarui data dengan benar kemudian tekan tombol <i>Edit</i> .	Proses <i>edit</i> data berhasil dan <i>pop up form edit</i> menghilang.	Gudang berhasil memperbarui data <i>supplier</i> dan <i>pop up form edit</i> menghilang.	Sesuai
2	Gudang memperbarui data dengan salah kemudian tekan tombol <i>Edit</i> .	Muncul peringatan bahwa masih ada data yang salah.	Gudang tidak berhasil memperbarui data <i>supplier</i> dan muncul peringatan bahwa masih ada data yang salah.	Sesuai

6. Pengujian pada Halaman *Categories* – Gudang

Pengujian dilakukan pada beberapa fungsi yang terdapat pada halaman *Categories*. Hasil pengujian pada halaman *categories* dapat dilihat pada Tabel 4.61.

Tabel 4.61 Pengujian pada Halaman *Categories* – Gudang

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Klik tombol <i>Add Category</i>	Menampilkan modal <i>form</i> tambah <i>Category</i> .	Berhasil membuka <i>form</i> untuk tambah data <i>Category</i> .	Sesuai
2	Klik tombol <i>Edit</i>	Menampilkan modal <i>form edit Category</i> .	Berhasil membuka <i>form</i> untuk <i>edit</i> data <i>Category</i> .	Sesuai
3	Pilih <i>select option Entries</i>	Menampilkan <i>list</i> data <i>Category</i> sesuai pilihan.	Berhasil menampilkan banyaknya <i>list</i> data <i>Category</i> sesuai pilihan.	Sesuai
4	Cari data pada <i>field Search</i>	Menampilkan data sesuai isi <i>field Search</i> .	Berhasil menampilkan data <i>Category</i> sesuai dengan isi <i>field Search</i> .	Sesuai
5	Klik tombol <i>Pagination</i>	Menampilkan data selanjutnya.	Berhasil menampilkan data sesuai dengan halaman <i>Pagination</i> .	Sesuai

#### 7. Pengujian *Form Tambah* pada Halaman *Categories* – Gudang

Pengujian dilakukan pada tiga sampel data *input* untuk menguji fungsionalitas tombol Tambah *Category*. Hasil pengujian *form* tambah pada halaman *Categories* dapat dilihat pada Tabel 4.62.

Tabel 4.62 Pengujian *Form Tambah* pada Halaman *Categories* – Gudang

No	Data <i>Input</i>	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan

1	Gudang memasukkan data dengan benar kemudian tekan tombol Tambah.	Proses <i>insert</i> data berhasil dan <i>pop up form</i> tambah menghilang.	Gudang berhasil memasukkan data <i>category</i> dan <i>pop up form</i> tambah menghilang.	Sesuai
2	Gudang memasukkan data yang salah kemudian tekan tombol Tambah.	Muncul peringatan bahwa masih ada <i>field</i> yang salah.	Gudang tidak berhasil memasukkan data <i>category</i> dan muncul peringatan bahwa masih ada <i>field</i> yang salah.	Sesuai
3	Gudang tidak memasukkan data dengan lengkap pada <i>field</i> kemudian tekan tombol Tambah.	Muncul peringatan bahwa masih ada <i>field</i> yang kosong.	Gudang tidak berhasil memasukkan data <i>category</i> dan muncul peringatan bahwa masih ada <i>field</i> yang kosong.	Sesuai

#### 8. Pengujian *Form Edit* pada Halaman *Categories* – Gudang

Pengujian dilakukan pada dua sampel data *input* untuk menguji fungsionalitas tombol *Edit category*. Hasil pengujian *form edit* pada halaman *Categories* dapat dilihat pada Tabel 4.63.

Tabel 4.63 Pengujian *Form Edit* pada Halaman *Categories* – Gudang

No	Data Input	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Gudang memperbaharui data dengan benar kemudian tekan tombol <i>Edit</i> .	Proses <i>edit</i> data berhasil dan <i>pop up form edit</i> menghilang.	Gudang berhasil memperbaharui data <i>category</i> dan <i>pop up form edit</i> menghilang.	Sesuai

2	Gudang memperbaharui data dengan salah kemudian tekan tombol <i>Edit.</i>	Muncul peringatan bahwa masih ada data yang salah.	Gudang tidak berhasil memperbaharui data <i>category</i> dan muncul peringatan bahwa masih ada data yang salah.	Sesuai
---	---	--	---	--------

9. Pengujian pada Halaman *Products* – Gudang

Pengujian dilakukan pada beberapa fungsi yang terdapat pada halaman *Products*. Hasil pengujian pada halaman *products* dapat dilihat pada Tabel 4.64.

Tabel 4.64 Pengujian pada Halaman *Products* – Gudang

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Klik tombol <i>Add Product</i>	Menampilkan modal form tambah <i>Product.</i>	Berhasil membuka form untuk tambah data <i>Product.</i>	Sesuai
2	Klik tombol <i>Edit</i>	Menampilkan modal form <i>edit Product.</i>	Berhasil membuka form untuk <i>edit</i> data <i>Product.</i>	Sesuai
3	Pilih <i>select option Entries</i>	Menampilkan <i>list</i> data <i>Product</i> sesuai pilihan.	Berhasil menampilkan banyaknya <i>list</i> data <i>Product</i> sesuai pilihan.	Sesuai
4	Cari data pada <i>field Search</i>	Menampilkan data sesuai isi <i>field Search.</i>	Berhasil menampilkan data <i>Product</i> sesuai dengan isi <i>field Search.</i>	Sesuai
5	Klik tombol <i>Pagination</i>	Menampilkan data selanjutnya.	Berhasil menampilkan data sesuai dengan halaman <i>Pagination.</i>	Sesuai

10. Pengujian *Form Tambah* pada Halaman *Products* – Gudang

Pengujian dilakukan pada tiga sampel data *input* untuk menguji fungsionalitas tombol *Tambah Product*. Hasil pengujian *form tambah* pada halaman *Products* dapat dilihat pada Tabel 4.65.

Tabel 4.65 Pengujian *Form Tambah* pada Halaman *Products* – Gudang

No	Data Input	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Gudang memasukkan data dengan benar kemudian tekan tombol Tambah.	Proses <i>insert</i> data berhasil dan <i>pop up form</i> tambah menghilang.	Gudang berhasil memasukkan data <i>product</i> dan <i>pop up form</i> tambah menghilang.	Sesuai
2	Gudang memasukkan data yang salah kemudian tekan tombol Tambah.	Muncul peringatan bahwa masih ada <i>field</i> yang salah.	Gudang tidak berhasil memasukkan data <i>product</i> dan muncul peringatan bahwa masih ada <i>field</i> yang salah.	Sesuai
3	Gudang tidak memasukkan data dengan lengkap pada <i>field</i> kemudian tekan tombol Tambah.	Muncul peringatan bahwa masih ada <i>field</i> yang kosong.	Gudang tidak berhasil memasukkan data <i>product</i> dan muncul peringatan bahwa masih ada <i>field</i> yang kosong.	Sesuai

11. Pengujian *Form Edit* pada Halaman *Products* – Gudang

Pengujian dilakukan pada dua sampel data *input* untuk menguji fungsionalitas tombol *Edit product*. Hasil pengujian *form edit* pada halaman *Products* dapat dilihat pada Tabel 4.66.

Tabel 4.66 Pengujian *Form Edit* pada Halaman *Products* – Gudang

No	Data Input	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Gudang memperbarui data dengan benar kemudian tekan tombol <i>Edit</i> .	Proses <i>edit</i> data berhasil dan <i>pop up form edit</i> menghilang.	Gudang berhasil memperbarui data <i>product</i> dan <i>pop up form edit</i> menghilang.	Sesuai
2	Gudang memperbarui data dengan salah kemudian tekan tombol <i>Edit</i> .	Muncul peringatan bahwa masih ada data yang salah.	Gudang tidak berhasil memperbarui data <i>product</i> dan muncul peringatan bahwa masih ada data yang salah.	Sesuai

## 12. Pengujian pada Halaman Stocks – Gudang

Pengujian dilakukan pada beberapa fungsi yang terdapat pada halaman Stocks. Hasil pengujian pada halaman stock dapat dilihat pada Tabel 4.29.

Tabel 4.67 Pengujian pada Halaman Stocks – Gudang

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Klik tombol <i>Edit</i>	Menampilkan modal <i>form edit Stock</i> .	Berhasil membuka <i>form</i> untuk <i>edit</i> data <i>Stock</i> .	Sesuai
2	Pilih <i>select option Entries</i>	Menampilkan <i>list</i> data <i>Stock</i> sesuai pilihan.	Berhasil menampilkan banyaknya <i>list</i> data <i>Stock</i> sesuai pilihan.	Sesuai
3	Cari data pada <i>field Search</i>	Menampilkan data sesuai isi <i>field Search</i> .	Berhasil menampilkan data <i>Stock</i> sesuai dengan isi <i>field Search</i> .	Sesuai

4	Klik tombol <i>Pagination</i>	Menampilkan data selanjutnya.	Berhasil menampilkan data sesuai dengan halaman <i>Pagination</i> .	Sesuai
---	-------------------------------	-------------------------------	---	--------

13. Pengujian *Form Edit* pada Halaman Stocks – Gudang

Pengujian dilakukan pada dua sampel data *input* untuk menguji fungsionalitas tombol *Edit Stock*. Hasil pengujian *form edit* pada halaman Stocks dapat dilihat pada Tabel 4.68.

Tabel 4.67 Pengujian *Form Edit* pada Halaman Stocks – Gudang

No	Data Input	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Gudang memperbarui data dengan benar kemudian tekan tombol <i>Edit</i> .	Proses <i>edit</i> data berhasil dan <i>pop up form edit</i> menghilang.	Gudang berhasil memperbarui data <i>stock</i> dan <i>pop up form edit</i> menghilang.	Sesuai
2	Gudang memperbarui data dengan salah kemudian tekan tombol <i>Edit</i> .	Muncul peringatan bahwa masih ada data yang salah.	Gudang tidak berhasil memperbarui data <i>stock</i> dan muncul peringatan bahwa masih ada data yang salah.	Sesuai

14. Pengujian pada Halaman Pembelian – Gudang

Pengujian dilakukan pada beberapa fungsi yang terdapat pada halaman Pembelian. Hasil pengujian pada halaman Pembelian dapat dilihat pada Tabel 4.69.

Tabel 4.68 Pengujian pada Halaman Pembelian – Gudang

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Klik tombol <i>Add Pembelian</i>	Menampilkan Halaman	Berhasil menampilkan	Sesuai

		tambah transaksi beli	Halaman tambah transaksi beli	
2	Klik tombol <i>Edit</i>	Menampilkan Halaman <i>edit</i> transaksi beli	Berhasil menampilkan Halaman <i>edit</i> transaksi beli	Sesuai
3	Klik tombol <i>Invoice</i>	Menampilkan Halaman <i>invoice</i> transaksi beli	Berhasil Menampilkan Halaman <i>invoice</i> transaksi beli	Sesuai
4	Pilih <i>select option Entries</i>	Menampilkan <i>list</i> data Penjualan sesuai pilihan.	Berhasil menampilkan banyaknya <i>list</i> data Pembelian sesuai pilihan.	Sesuai
5	Cari data pada <i>field Search</i>	Menampilkan data sesuai isi <i>field Search</i> .	Berhasil menampilkan data Pembelian sesuai dengan isi <i>field Search</i> .	Sesuai
6	Klik tombol <i>Pagination</i>	Menampilkan data selanjutnya.	Berhasil menampilkan data sesuai dengan halaman <i>Pagination</i> .	Sesuai

15. Pengujian pada Halaman Tambah Transaksi Beli – Gudang

Pengujian dilakukan pada tiga sampel data *input* untuk menguji fungsionalitas tombol *Submit*. Hasil pengujian pada halaman Tambah Transaksi Beli dapat dilihat pada Tabel 4.70.

Tabel 4.69 Pengujian pada Halaman Tambah Transaksi Beli – Gudang

No	Data <i>Input</i>	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Gudang memasukkan data dengan	Proses <i>insert</i> data berhasil	Gudang berhasil memasukkan data	Sesuai

	benar kemudian tekan tombol <i>Submit</i> .	dan isi <i>field</i> direset.	pembelian dan isi <i>field</i> direset.	
2	Gudang memasukkan data yang salah kemudian tekan tombol <i>Submit</i> .	Muncul peringatan bahwa masih ada <i>field</i> yang salah.	Gudang tidak berhasil memasukkan data pembelian dan muncul peringatan bahwa masih ada <i>field</i> yang salah.	Sesuai
3	Gudang tidak memasukkan data dengan lengkap pada <i>field</i> kemudian tekan tombol <i>Submit</i> .	Muncul peringatan bahwa masih ada <i>field</i> yang kosong.	Gudang tidak berhasil memasukkan data pembelian dan muncul peringatan bahwa masih ada <i>field</i> yang kosong.	Sesuai

16. Pengujian pada Halaman *Edit* Transaksi Beli – Gudang

Pengujian dilakukan pada dua sampel data *input* untuk menguji fungsionalitas tombol *Submit*. Hasil pengujian pada halaman *Edit* Transaksi Beli dapat dilihat pada Tabel 4.71.

Tabel 4.70 Pengujian pada Halaman *Edit* Transaksi Beli – Gudang

No	Data Input	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Gudang memperbarui data dengan benar kemudian tekan tombol <i>Submit</i> .	Proses <i>edit</i> data berhasil dan muncul peringatan berhasil	Gudang berhasil memperbarui data pembelian dan muncul peringatan berhasil	Sesuai
2	Gudang memperbarui data dengan salah kemudian	Muncul peringatan bahwa masih	Gudang tidak berhasil memperbarui data pembelian dan	Sesuai

	tekan tombol <i>Submit</i> .	ada data yang salah.	muncul peringatan bahwa masih ada data yang salah.	
--	---------------------------------	----------------------	--	--

17. Pengujian pada Halaman Laporan Penjualan – Gudang

Pengujian dilakukan pada beberapa fungsi yang terdapat pada halaman Laporan Penjualan. Hasil pengujian pada halaman Laporan Penjualan dapat dilihat pada Tabel 4.72.

Tabel 4.71 Pengujian pada Halaman Laporan Penjualan – Gudang

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Pilih tahun dan klik tombol <i>Submit</i> .	Menampilkan data grafik sesuai dengan tahun yang dipilih.	Berhasil Menampilkan data grafik sesuai dengan tahun yang dipilih.	Sesuai
2	Pilih tanggal dan klik tombol <i>Filter</i> .	Menampilkan data <i>grid</i> sesuai dengan tanggal yang dipilih	Berhasil menampilkan data <i>grid</i> sesuai dengan tanggal yang dipilih	Sesuai
3	Klik tombol <i>copy</i>	<i>Copy</i> data <i>grid</i>	<i>Copy</i> data <i>grid</i>	Sesuai
4	Klik tombol <i>csv</i>	<i>Download</i> data <i>grid</i> bertipe <i>csv</i>	Berhasil <i>Download</i> data <i>grid</i> bertipe <i>csv</i>	Sesuai
5	Klik tombol <i>excel</i>	<i>Download</i> data <i>grid</i> bertipe <i>xlsx</i>	Berhasil <i>Download</i> data <i>grid</i> bertipe <i>xlsx</i>	Sesuai
6	Klik tombol <i>print</i>	<i>Print</i> data <i>grid</i>		Sesuai
7	Klik tombol <i>Add Penjualan</i>	Menampilkan Halaman tambah transaksi jual	Berhasil menampilkan Halaman tambah transaksi jual	Sesuai
8	Klik tombol <i>Edit</i>	Menampilkan Halaman <i>edit</i> transaksi jual	Berhasil menampilkan Halaman <i>edit</i> transaksi jual	Sesuai

9	Klik tombol <i>Invoice</i>	Menampilkan Halaman <i>invoice</i> transaksi jual	Berhasil Menampilkan Halaman <i>invoice</i> transaksi jual	Sesuai
10	Pilih <i>select option Entries</i>	Menampilkan <i>list</i> data Penjualan sesuai pilihan.	Berhasil menampilkan banyaknya <i>list</i> data Penjualan sesuai pilihan.	Sesuai
11	Cari data pada <i>field Search</i>	Menampilkan data sesuai isi <i>field Search</i> .	Berhasil menampilkan data Penjualan sesuai dengan isi <i>field Search</i> .	Sesuai
12	Klik tombol <i>Pagination</i>	Menampilkan data selanjutnya.	Berhasil menampilkan data sesuai dengan halaman <i>Pagination</i> .	Sesuai

#### 18. Pengujian pada Halaman Laporan Pembelian – Gudang

Pengujian dilakukan pada beberapa fungsi yang terdapat pada halaman Laporan Pembelian. Hasil pengujian pada halaman Laporan Pembelian dapat dilihat pada Tabel 4.73.

Tabel 4.72 Pengujian pada Halaman Laporan Pembelian – Gudang

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
1	Pilih tahun dan klik tombol Submit.	Menampilkan data grafik sesuai dengan tahun yang dipilih.	Berhasil Menampilkan data grafik sesuai dengan tahun yang dipilih.	Sesuai
2	Pilih tanggal dan klik tombol Filter.	Menampilkan data <i>grid</i> sesuai dengan tanggal yang dipilih	Berhasil menampilkan data <i>grid</i> sesuai dengan tanggal yang dipilih	Sesuai

3	Klik tombol <i>copy</i>	<i>Copy data grid</i>	<i>Copy data grid</i>	Sesuai
4	Klik tombol csv	<i>Download data grid</i> bertipe csv	Berhasil <i>Download data grid</i> bertipe csv	Sesuai
5	Klik tombol excel	<i>Download data grid</i> bertipe xlsx	Berhasil <i>Download data grid</i> bertipe xlsx	Sesuai
6	Klik tombol print	<i>Print data grid</i>		Sesuai
7	Klik tombol <i>Add Pembelian</i>	Menampilkan Halaman tambah transaksi beli	Berhasil menampilkan Halaman tambah transaksi beli	Sesuai
8	Klik tombol <i>Edit</i>	Menampilkan Halaman <i>edit</i> transaksi beli	Berhasil menampilkan Halaman <i>edit</i> transaksi beli	Sesuai
9	Klik tombol <i>Invoice</i>	Menampilkan Halaman <i>invoice</i> transaksi beli	Berhasil Menampilkan Halaman <i>invoice</i> transaksi beli	Sesuai
10	Pilih <i>select option Entries</i>	Menampilkan <i>list data Pembelian</i> sesuai pilihan.	Berhasil menampilkan banyaknya <i>list data Pembelian</i> sesuai pilihan.	Sesuai
11	Cari data pada <i>field Search</i>	Menampilkan data sesuai isi <i>field Search.</i>	Berhasil menampilkan data Pembelian sesuai dengan isi <i>field Search.</i>	Sesuai
12	Klik tombol <i>Pagination</i>	Menampilkan data selanjutnya.	Berhasil menampilkan data sesuai dengan halaman <i>Pagination.</i>	Sesuai

## **BAB V**

### **PENUTUP**

#### **5.1 Kesimpulan**

Berdasarkan hasil penelitian yang telah dilakukan pada Sistem Informasi Eksekutif Penjualan Kerajinan Tas Rotan Pada Balinese Classic Menggunakan *Framework Laravel*, maka dapat diambil kesimpulan sebagai berikut:

1. Telah dihasilkan suatu Sistem Informasi Eksekutif Penjualan Kerajinan Tas Rotan Pada Balinese Classic Menggunakan *Framework Laravel*.
2. Sistem ini telah melalui proses perancangan dengan menggunakan *Data Flow Diagram* (DFD), perancangan basis data dengan menggunakan *Entity Relationship Diagram* (ERD), Basis Data Konseptual dan Struktur Tabel.
3. Sistem berhasil dibangun berbasis Bahasa pemrograman PHP menggunakan *framework* Laravel dikombinasikan dengan Bahasa pemrograman javascript Ajax yang dimaksimalkan dengan *framework* CSS yaitu Bootstrap
4. Sistem ini memiliki beberapa *fitur* yaitu mengelola data seperti data *users*, data *product*, data transaksi, dan data laporan.
5. Sistem ini menghasilkan *output* data berupa data *grid* dan juga data grafik.
6. Sistem ini telah diuji menggunakan metode *Blackbox Testing* dan seluruh pengujian yang dilakukan telah memperoleh hasil yang sesuai.

#### **5.2 Saran**

Adapun saran yang dapat diberikan untuk menyempurnakan perancangan ini agar sistem menjadi lebih baik adalah sebagai berikut:

1. Sistem ini dapat dikembangkan menjadi aplikasi berbasis *Mobile Application* sehingga lebih praktis dalam penggunaannya.
2. Sistem dapat melakukan *scan barcode product*.



## DAFTAR PUSTAKA

- [1] Y. Maulana, "Sistem Informasi Penjualan Kayu Ulin Pada UD Karya Perdana Berbasis Web Menggunakan Framework Codeigniter", Denpasar: (STIMIK) STIKOM Bali, 2018.
- [2] I. K. A. Adnyana, "Sistem Informasi Eksekutif Penjualan Jajanan Sagu Pada Bagia Suli Menggunakan Framework Laravel", Denpasar: (STIMIK) STIKOM Bali, 2018.
- [3] K. O. E. Artawan, "Sistem Informasi Penyewaan Busana Pada Denata Kebaya Menggunakan Framework Laravel", Denpasar: (STIMIK) STIKOM Bali, 2019.
- [4] I. P. Kawiyana, "Sistem Informasi Eksekutif Pengadaan Barang Pada Saturday Cloth Menggunakan Framework Laravel", Denpasar: (STIMIK) STIKOM Bali, 2019.
- [5] S. Setyaningrum, Konsep dan Perancangan Basis Data, Yoyakarta: PT. Skripta Media Creativ, 2011.
- [6] J. S. P. Tyoso, Sistem Informasi Manajemen, Yogyakarta: CV BUDI UTAMA, 2016.
- [7] A. N. Wayan, "Sistem Informasi Eksekutif Bantuan Kelompok Usaha Bersama (KUBE) Pada Dinas SOSial Provinsi Bali", Denpasar: (STIMIK) STIKOM Bali, 2017.
- [8] R. Budi, Belajar Otodidak MySQL, Bandung: Informatika Bandung, 2015.
- [9] N. K. D. A. Jayanti, Teori Basis Data, Denpasar: CV Andi Offset, 2018.
- [10] A. Subagia, Membuat Web dengan PHP 7 dan Database PDO MySQLi, Jakarta: PT Elex Komputindo, 2016.
- [11] Sudarmaji, "Migrasi dan Optimalisasi Database Sistem Informasi berbasis E-Learning Program Diploma III Manajemen Informatika", Universitas Muhammadiyah Metro, 2017.
- [12] A. Rohman, Mengenal Framework Laravel, Ilmu TI, 2014.
- [13] [Online]. Available: <https://vuejs.org/>. [Diakses Kamis 23 2019].
- [14] J. Spurlock, Responsive Web Development, O'Reilly Media, 2013.

- [15] Pratama. E, Sistem Informasi dan Implementasinya, Bandung: Informatika Bandung, 2014.
- [16] F. H. U. Asnawati, Rekayasa Perangkat Lunak, Yogyakarta: CV Budi Utama, 2015.



INSTITUT TEKNOLOGI DAN BISNIS  
**STIKOM BALI**