# BAB 4
# IMPLEMENTASI DAN PENGUJIAN SISTEM

## 4.1 Implementasi Sistem

Tahapan implementasi dan pengujian adalah tahap dimana perancangan diimplementasikan pada bahasa pemrograman yang digunakan. Setelah diimplementasikan kemudian dilakukan pengujian. Tahap pengujian dilakukan untuk melihat kekurangan-kekurangan yang mungkin terjadi pada aplikasi yang baru dibangun.

### 4.1.1. Implementasi Perangkat Keras

Spesifikasi perangkat keras yang digunakan dalam mengimplementasikan aplikasi*supply chain management* ini adalah sebagai berikut:

**Tabel 4.1 Spesifikasi Perangkat Keras**

| Perangkat Keras | Yang digunakan | Yang dibutuhkan |
|---|---|---|
| *Processor* | Intel i5 2,8 GHz | Intel Pentium 4 1,5 GHz |
| RAM (*Memory*) | 4 GB | 1 GB |
| *Harddisk* | 500 Gb | 20 GB |
| Monitor | 17 Inch | 14 Inch |

### 4.1.2. Kebutuhan Perangkat Lunak

Spesifikasi perangkat lunak yang digunakan dalam mengimplementasikan aplikasi*supply chain management* ini adalah sebagai berikut:

**Tabel 4.2 Spesifikasi Perangkat Lunak**

| Perangkat Keras | Yang digunakan | Yang dibutuhkan |
|---|---|---|
| Sistem Operasi | Window 8 | Window XP |
| DBMS | MySQL | MySQL |
| Browser | Google Chrome, MozillaFirefox, | Google Chrome, MozillaFirefox, |
| Code Editor | Sublime 5 | Sublime 5 |

### 4.1.3. Implementasi Basis Data

Pembuatan basis data dilakukan dengan menggunakan DBMS *MySQL*. Implementasi basis data dalam bahasa *SQL* adalah sebagai berikut:

**1. Tabel Bagian**

Tabel bagian akan digunakan untuk menyimpan data bagian. Penjelasannya terdapat pada tabel 4.3 :

**Tabel 4.3 Tabel Bagian**

| No | SQL |
|---|---|
| 1 | `CREATE TABLE `bagian` (` |
| 2 | `  `kd_bagian` int(11) NOT NULL,` |
| 3 | `  `nama_bagian` varchar(20) NOT NULL,` |
| 4 | `  `deskripsi` varchar(30) NOT NULL` |
| 5 | `) ENGINE=InnoDB DEFAULT CHARSET=latin1;` |
| 6 | |
| 7 | `--` |
| 8 | `-- Dumping data for table `bagian`` |
| 9 | `--` |
| 10 | |
| 11 | `INSERT INTO `bagian` (`kd_bagian`, `nama_bagian`,` |
| 12 | `` `deskripsi`) VALUES `` |
| 13 | `(1, 'Staff', 'Bagian Staff'),` |
| 14 | `(2, 'Gudang', 'Bagian Gudang'),` |
| 15 | `(3, 'Pemasaran', 'Bagian Pemasaran'),` |
| 16 | `(4, 'Produksi', 'Bagian Produksi'),` |
| 17 | `(5, 'Keuangan', 'Bagian Keuangan');` |
| 18 | |
| 19 | `--` |
| 20 | `-- Indexes for dumped tables` |
| 21 | `--` |
| 22 | |
| 23 | `--` |
| 24 | `-- Indexes for table `bagian`` |
| 25 | `--` |
| 26 | `ALTER TABLE `bagian`` |
| 27 | `  ADD PRIMARY KEY (`kd_bagian`);` |
| 28 | |
| 29 | |
| 30 | |

**2. Tabel Barang**

Tabel barang akan digunakan untuk menyimpan data barang. Penjelasannya terdapat pada tabel 4.4 :

**Tabel 4.4 Tabel Barang**

| No | SQL |
|---|---|
| 1 | `CREATE TABLE `barang` (` |
| 2 | `  `id_barang` int(7) NOT NULL,` |
| 3 | `  `jumlahstok` double NOT NULL` |
| 4 | `) ENGINE=InnoDB DEFAULT CHARSET=latin1;` |
| 5 | |
| 6 | `--` |
| 7 | `-- Dumping data for table `barang`` |
| 8 | `--` |
| 9 | |
| 10 | `INSERT INTO `barang` (`id_barang`, `jumlahstok`)` |
| 11 | `VALUES` |
| 12 | `(1, 850);` |
| 13 | |
| 14 | `--` |
| 15 | `-- Indexes for dumped tables` |
| 16 | `--` |
| 17 | |
| 18 | `--` |
| 19 | `-- Indexes for table `barang`` |
| 20 | `--` |
| 21 | `ALTER TABLE `barang`` |
| 22 | `  ADD PRIMARY KEY (`id_barang`);` |
| 23 | |
| 24 | `--` |
| 25 | `-- AUTO_INCREMENT for dumped tables` |
| 26 | `--` |
| 27 | |
| 28 | `--` |
| 29 | `-- AUTO_INCREMENT for table `barang`` |
| 30 | `--` |
| 31 | `ALTER TABLE `barang`` |
| 32 | `  MODIFY `id_barang` int(7) NOT NULL AUTO_INCREMENT,` |
| 33 | `AUTO_INCREMENT=2;` |
| 34 | |
| 35 | |
| 36 | |

### 3. Tabel Bom

Tabel bom akan digunakan untuk menyimpan data bom yang digunakan. Penjelasannya terdapat pada tabel 4.5 :

**Tabel 4.5 Tabel Bom**

| No | SQL |
|---|---|
| 1 | `CREATE TABLE `bom` (` |
| 2 | `  `kd_bom` int(11) NOT NULL,` |
| 3 | `  `jenis_bom` enum('stoksekarang','stokperamalan')` |
| 4 | `NOT NULL,` |
| 5 | `  `stok_katun` double NOT NULL,` |
| 6 | `  `stok_obatbasa` double NOT NULL,` |
| 7 | `  `stok_obatwarna` double NOT NULL,` |
| 8 | `  `stok_obataprek` double NOT NULL,` |
| 9 | `  `stok_benang` double NOT NULL,` |
| 10 | `  `stok_size` double NOT NULL,` |
| 11 | `  `stok_rip` double NOT NULL,` |
| 12 | `  `stok_plastik` double NOT NULL,` |
| 13 | `  `stok_handbag` double NOT NULL,` |
| 14 | `  `stok_label` double NOT NULL,` |
| 15 | `  `tanggal` date NOT NULL,` |
| 16 | `  `status` varchar(30) NOT NULL` |
| 17 | `) ENGINE=InnoDB DEFAULT CHARSET=latin1;` |
| 18 | |
| 19 | `--` |
| 20 | `-- Dumping data for table `bom`` |
| 21 | `--` |
| 22 | |
| 23 | `INSERT INTO `bom` (`kd_bom`, `jenis_bom`,` |
| 24 | `` `stok_katun`, `stok_obatbasa`, `stok_obatwarna`, `` |
| 25 | `` `stok_obataprek`, `stok_benang`, `stok_size`, `` |
| 26 | `` `stok_rip`, `stok_plastik`, `stok_handbag`, `` |
| 27 | `` `stok_label`, `tanggal`, `status`) VALUES `` |
| 28 | `(1, 'stokperamalan', 100, 100, 100, 100, 100, 200,` |
| 29 | `200, 200, 200, 200, '2016-07-24', 'Ok');` |
| 30 | |
| 31 | `--` |
| 32 | `-- Indexes for dumped tables` |
| 33 | `--` |
| 34 | |
| 35 | `--` |
| 36 | `-- Indexes for table `bom`` |
| 37 | `--` |
| 38 | `ALTER TABLE `bom`` |
| 39 | `  ADD PRIMARY KEY (`kd_bom`);` |
| 40 | |
| 41 | |
| 42 | |
| 43 | |

```
44
45  --
46  -- AUTO_INCREMENT for dumped tables
47  --
48
49  --
50  -- AUTO_INCREMENT for table `bom`
51  --
52  ALTER TABLE `bom`
53    MODIFY `kd_bom` int(11) NOT NULL AUTO_INCREMENT,
54  AUTO_INCREMENT=2;
55
```

**4. Tabel Jabatan**

Tabel jabatan akan digunakan untuk menyimpan data jabatan yang digunakan. Penjelasannya terdapat pada tabel 4.6 :

**Tabel 4.6 Tabel Jabatan**

| No | SQL |
|---|---|

```
1   CREATE TABLE `jabatan` (
2     `kd_jabatan` int(11) NOT NULL,
3     `nama_jabatan` varchar(20) NOT NULL,
4     `deskripsi` varchar(30) NOT NULL
5   ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
6
7   --
8   -- Dumping data for table `bagian`
9   --
10
11  INSERT INTO `bagian` (`kd_bagian`, `nama_bagian`,
12  `deskripsi`) VALUES
13  (1, 'Staff', 'Bagian Staff'),
14  (2, 'Gudang', 'Bagian Gudang'),
15  (3, 'Pemasaran', 'Bagian Pemasaran'),
16  (4, 'Produksi', 'Bagian Produksi'),
17  (5, 'Keuangan', 'Bagian Keuangan');
18
19  --
20  -- Indexes for dumped tables
21  --
22
23  --
24  -- Indexes for table `bagian`
25  --
26  ALTER TABLE `bagian`
27
28
```

```
  ADD PRIMARY KEY (`kd_bagian`);
```

**5. Tabel Monitoring sebelum peramalan**

Tabel monitoring sebelum akan digunakan untuk menyimpan data monitoring sebelum peramalan. Penjelasannya terdapat pada tabel 4.7 :

**Tabel 4.7 Tabel Monitoring Sebelum Peramalan**

| No | SQL |
|---|---|

```
CREATE TABLE `monitoringsebelum` (
  `kd_mseb` varchar(8) NOT NULL,
  `tanggal` date NOT NULL,
  `jum_katun` int(11) NOT NULL,
  `jum_obatbusa` int(11) NOT NULL,
  `jum_obatwarna` int(11) NOT NULL,
  `jum_obataprek` int(11) NOT NULL,
  `jum_benang` int(11) NOT NULL,
  `jum_size` int(11) NOT NULL,
  `jum_rip` int(11) NOT NULL,
  `jum_plastik` int(11) NOT NULL,
  `jum_handbag` int(11) NOT NULL,
  `jum_label` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--
-- Indexes for dumped tables
--


--
-- Indexes for table `monitoringsebelum`
--
ALTER TABLE `monitoringsebelum`
  ADD PRIMARY KEY (`kd_mseb`);
 (4, 'Operator', 'Operator');

--
-- Indexes for dumped tables
--


--
-- Indexes for table `jabatan`
--
ALTER TABLE `jabatan`
  ADD PRIMARY KEY (`kd_jabatan`);
```

6. **Tabel Monitoring Sesudah Peramalan**

Tabel monitoring sebelum akan digunakan untuk menyimpan data monitoring sesudah eramalan. Penjelasannya terdapat pada tabel 4.8 :

**Tabel 4.8 Tabel Monitoring Sesudah Peramalan**

| No | SQL |
|---|---|
| 1 | `CREATE TABLE `monitoringsesudah` (` |
| 2 | `  `kd_msud` varchar(8) NOT NULL,` |
| 3 | `  `tanggal` date NOT NULL,` |
| 4 | `  `jum_katun` int(11) NOT NULL,` |
| 5 | `  `jum_obatbusa` int(11) NOT NULL,` |
| 6 | `  `jum_obatwarna` int(11) NOT NULL,` |
| 7 | `  `jum_obataprek` int(11) NOT NULL,` |
| 8 | `  `jum_benang` int(11) NOT NULL,` |
| 9 | `  `jum_size` int(11) NOT NULL,` |
| 10 | `  `jum_rip` int(11) NOT NULL,` |
| 11 | `  `jum_plastik` int(11) NOT NULL,` |
| 12 | `  `jum_handbag` int(11) NOT NULL,` |
| 13 | `  `jum_label` int(11) NOT NULL` |
| 14 | `) ENGINE=InnoDB DEFAULT CHARSET=latin1;` |
| 15 | |
| 16 | `--` |
| 17 | `-- Indexes for dumped tables` |
| 18 | `--` |
| 19 | |
| 20 | `--` |
| 21 | `-- Indexes for table `monitoringsesudah`` |
| 22 | `--` |
| 23 | `ALTER TABLE `monitoringsesudah`` |
| 24 | `  ADD PRIMARY KEY (`kd_msud`);` |
| 25 | |
| 26 | |

7. **Tabel Pengiriman**

Tabel jabatan akan digunakan untuk menyimpan data pengiriman yang digunakan. Penjelasannya terdapat pada tabel 4.9 :

**Tabel 4.9 Tabel Pengiriman**

| No | SQL |
|---|---|
| 1 | `CREATE TABLE `pengiriman` (` |
| 2 | `  `kd_pengiriman` int(7) NOT NULL,` |
| 3 | `  `tanggalkirim` date NOT NULL,` |
| 4 | `  `kotatujuan` varchar(30) NOT NULL,` |
| 5 | `  `waktukirim` varchar(12) NOT NULL,` |
| 6 | |

```
  `jenisproduk` varchar(30) NOT NULL,
  `jumlah` int(7) NOT NULL,
  `kendaraan` varchar(30) NOT NULL,
  `supir` varchar(30) NOT NULL,
  `keterangan` varchar(30) NOT NULL,
  `status` varchar(30) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--
-- Dumping data for table `pengiriman`
--

INSERT INTO `pengiriman` (`kd_pengiriman`,
`tanggalkirim`, `kotatujuan`, `waktukirim`,
`jenisproduk`, `jumlah`, `kendaraan`, `supir`,
`keterangan`, `status`) VALUES
(1, '2016-06-06', 'Bandung', '21:31', 'Kaos Oblong',
50, 'D 1270 VA', 'Riki', 'Diantar Ke Gudang', 'Dalam
Pengiriman');

--
-- Indexes for dumped tables
--

--
-- Indexes for table `pengiriman`
--
ALTER TABLE `pengiriman`
  ADD PRIMARY KEY (`kd_pengiriman`);

--
-- AUTO_INCREMENT for dumped tables
--

--
-- AUTO_INCREMENT for table `pengiriman`
--
ALTER TABLE `pengiriman`
  MODIFY `kd_pengiriman` int(7) NOT NULL
AUTO_INCREMENT, AUTO_INCREMENT=2;
```

**8. Tabel Penjualan**

Tabel jabatan akan digunakan untuk menyimpan data penjualan yang digunakan. Penjelasannya terdapat pada tabel 4.10 :

**Tabel 4.10Tabel Penjualan**

| No | SQL |
|---|---|
| 1 | CREATE TABLE `penjualan` ( |
| 2 | `kd_penjualan` int(7) NOT NULL, |
| 3 | `atasnama` varchar(30) NOT NULL, |
| 4 | `jumlah` double NOT NULL, |
| 5 | `tanggal` date NOT NULL, |
| 6 | `keterangan` varchar(30) NOT NULL, |
| 7 | `status` varchar(30) NOT NULL |
| 8 | ) ENGINE=InnoDB DEFAULT CHARSET=latin1; |
| 9 | |
| 10 | -- |
| 11 | -- Dumping data for table `penjualan` |
| 12 | -- |
| 13 | |
| 14 | INSERT INTO `penjualan` (`kd_penjualan`, `atasnama`, |
| 15 | `jumlah`, `tanggal`, `keterangan`, `status`) VALUES |
| 16 | (13, 'Sukijan', 120, '2016-07-21', 'Diantar Ke JNE', |
| 17 | 'Order'), |
| 18 | (14, 'Sukiman', 150, '2016-07-21', 'Diantar Ke Indah |
| 19 | Kargo', 'Ok'), |
| 20 | (15, 'welly', 30, '2016-07-23', 'Diantar Ke Gudang', |
| 21 | 'Order'); |
| 22 | |
| 23 | -- |
| 24 | -- Indexes for dumped tables |
| 25 | -- |
| 26 | |
| 27 | -- |
| 28 | -- Indexes for table `penjualan` |
| 29 | -- |
| 30 | ALTER TABLE `penjualan` |
| 31 | ADD PRIMARY KEY (`kd_penjualan`); |
| 32 | |
| 33 | -- |
| 34 | -- AUTO_INCREMENT for dumped tables |
| 35 | -- |
| 36 | |
| 37 | -- |
| 38 | -- AUTO_INCREMENT for table `penjualan` |
| 39 | -- |
| 40 | ALTER TABLE `penjualan` |
| 41 | MODIFY `kd_penjualan` int(7) NOT NULL |
| 42 | AUTO_INCREMENT, AUTO_INCREMENT=16; |
| 43 | |
| 44 | |
| 45 | |
| 46 | |

**9. Tabel Peramalan**

Tabel jabatan akan digunakan untuk menyimpan data peramalan yang digunakan. Penjelasannya terdapat pada tabel4.11:

**Tabel 4.11Tabel Peramalan**

| No | SQL |
|---|---|
| 1 | CREATE TABLE `peramalan` ( |
| 2 | `kd_peramalan` int(8) NOT NULL, |
| 3 | `tanggal` date NOT NULL, |
| 4 | `bulan` varchar(30) NOT NULL, |
| 5 | `jumlah` int(11) NOT NULL |
| 6 | ) ENGINE=InnoDB DEFAULT CHARSET=latin1; |
| 7 | |
| 8 | -- |
| 9 | -- Dumping data for table `peramalan` |
| 10 | -- |
| 11 | |
| 12 | INSERT INTO `peramalan` (`kd_peramalan`, `tanggal`, |
| 13 | `bulan`, `jumlah`) VALUES |
| 14 | (7, '2016-07-24', 'November 2015', 9946), |
| 15 | (8, '2016-07-24', 'Agustus 2016', 9953); |
| 16 | |
| 17 | -- |
| 18 | -- Indexes for dumped tables |
| 19 | -- |
| 20 | |
| 21 | -- |
| 22 | -- Indexes for table `peramalan` |
| 23 | -- |
| 24 | ALTER TABLE `peramalan` |
| 25 | ADD PRIMARY KEY (`kd_peramalan`); |
| 26 | |
| 27 | -- |
| 28 | -- AUTO_INCREMENT for dumped tables |
| 29 | -- |
| 30 | |
| 31 | -- |
| 32 | -- AUTO_INCREMENT for table `peramalan` |
| 33 | -- |
| 34 | ALTER TABLE `peramalan` |
| 35 | MODIFY `kd_peramalan` int(8) NOT NULL |
| 36 | AUTO_INCREMENT, AUTO_INCREMENT=9; |
| 37 | |
| 38 | |
| 39 | |

**10. Tabel Produksi**

Tabel jabatan akan digunakan untuk menyimpan data peramalan yang digunakan. Penjelasannya terdapat pada tabel4.12 :

**Tabel 4.12 Tabel Produksi**

| No | SQL |
|---|---|
| 1 | CREATE TABLE `produksi` ( |
| 2 | `kd_produksi` int(11) NOT NULL, |
| 3 | `jumlah` double NOT NULL, |
| 4 | `tanggal` date NOT NULL, |
| 5 | `prod_katun` double NOT NULL, |
| 6 | `prod_obatbasa` double NOT NULL, |
| 7 | `prod_obatwarna` double NOT NULL, |
| 8 | `prod_obataprek` double NOT NULL, |
| 9 | `prod_benang` double NOT NULL, |
| 10 | `prod_size` double NOT NULL, |
| 11 | `prod_rip` double NOT NULL, |
| 12 | `prod_plastik` double NOT NULL, |
| 13 | `prod_handbag` double NOT NULL, |
| 14 | `prod_label` double NOT NULL |
| 15 | ) ENGINE=InnoDB DEFAULT CHARSET=latin1; |
| 16 | |
| 17 | -- |
| 18 | -- Dumping data for table `produksi` |
| 19 | -- |
| 20 | |
| 21 | INSERT INTO `produksi` (`kd_produksi`, `jumlah`, |
| 22 | `tanggal`, `prod_katun`, `prod_obatbasa`, |
| 23 | `prod_obatwarna`, `prod_obataprek`, `prod_benang`, |
| 24 | `prod_size`, `prod_rip`, `prod_plastik`, |
| 25 | `prod_handbag`, `prod_label`) VALUES |
| 26 | (15, 100, '2016-07-24', 15, 0.75, 0.5, 0.25, 15, |
| 27 | 100, 100, 100, 100, 100), |
| 28 | (16, 100, '2016-07-24', 15, 0.75, 0.5, 0.25, 15, |
| 29 | 100, 100, 100, 100, 100); |
| 30 | |
| 31 | -- |
| 32 | -- Indexes for dumped tables |
| 33 | -- |
| 34 | |
| 35 | -- |
| 36 | -- Indexes for table `produksi` |
| 37 | -- |
| 38 | ALTER TABLE `produksi` |
| 39 | ADD PRIMARY KEY (`kd_produksi`); |
| 40 | |
| 41 | |
| 42 | |
| 43 | |

```
44
45  --
46  -- AUTO_INCREMENT for dumped tables
47  --
48
49  --
50  -- AUTO_INCREMENT for table `produksi`
51  --
52  ALTER TABLE `produksi`
53    MODIFY `kd_produksi` int(11) NOT NULL
54  AUTO_INCREMENT, AUTO_INCREMENT=17;
```

**11. Tabel User**

Tabel jabatan akan digunakan untuk menyimpan data user yang digunakan. Penjelasannya terdapat pada tabel 4.13:

**Tabel 4.13 Tabel User**

| No | SQL |
|---|---|

```
1   CREATE TABLE `user` (
2     `kd_user` int(7) NOT NULL,
3     `username` varchar(20) NOT NULL,
4     `password` varchar(20) NOT NULL,
5     `nama` varchar(30) NOT NULL,
6     `kd_bagian` varchar(30) NOT NULL,
7     `kd_jabatan` varchar(30) NOT NULL,
8     `hakakses` varchar(20) NOT NULL
9   ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
10
11  --
12  -- Dumping data for table `user`
13  --
14
15  INSERT INTO `user` (`kd_user`, `username`,
16  `password`, `nama`, `kd_bagian`, `kd_jabatan`,
17  `hakakses`) VALUES
18  (1, 'admin', 'admin', 'admin', 'Gudang', 'Manager',
19  'admin'),
20  (3, 'sarah', 'sarah', 'Sarah', 'Staff', 'Manager',
21  'generlgm'),
22  (4, 'rizal', 'rizal', 'Rizal', 'Gudang', 'Kepala
23  Bagian', 'kepgudang'),
24  (5, 'agung', 'agung', 'Agung', 'Pemasaran', 'Kepala
25  Bagian', 'keppemasaran'),
26  (6, 'rukiah', 'rukiah', 'Rukiah', 'Produksi',
27
28
```

```
'Staff', 'kepproduksi'),
(7, 'riki', 'riki', 'Riki', 'Staff', 'Kepala
Bagian', 'admin');

--
-- Indexes for dumped tables
--

--
-- Indexes for table `user`
--
ALTER TABLE `user`
  ADD PRIMARY KEY (`kd_user`);

--
-- AUTO_INCREMENT for dumped tables
--

--
-- AUTO_INCREMENT for table `user`
--
ALTER TABLE `user`
```

### 4.1.4. Implementasi*Class Diagram*

Adapun implementasi *class diagram*untuk pemodelan *Model*, *View*, *Control (MVC)*pada aplikasi*supply chain management*yang dibangun adalah sebagai berikut:

**A. Class Controller**

1. Kelas Beranda

   Implementasi *Class Diagram* pada Kelas Beranda dapat dilihat pada Tabel 4.14.

**Tabel 4.14 Implementasi Kelas Beranda**

| File : Beranda | |
|---|---|
| **Nama Kelas** | |
| **Rancangan** | **Implementasi** |
| beranda | class beranda extends CI_Controller {} |
| **Atribut** | |
| **Rancangan** | **Implementasi** |

| | |
|---|---|
| **Method** | |
| **Rancangan** | **Implementasi** |
| +__construct()<br>+index() | public function __construct() {}<br>public function index() {} |

2. Kelas Login

   Implementasi *Class Diagram* pada Kelas Login dapat dilihat pada Tabel 4.15.

**Tabel 4.15 Implementasi Kelas Login**

| File : Login | |
|---|---|
| **Nama Kelas** | |
| **Rancangan** | **Implementasi** |
| login | class login extends CI_Controller {} |
| **Atribut** | |
| **Rancangan** | **Implementasi** |
| | |
| **Method** | |
| **Rancangan** | **Implementasi** |
| +__construct()<br>+index()<br>+cek_login()<br>+logout() | public function __construct() {}<br>public function index() {}<br>public function cek_login() {<br>public function logout() {} |

3. Kelas Kelola_User

   Implementasi *Class Diagram* pada Kelas Kelola_User dapat dilihat pada Tabel 4.16.

**Tabel 4.16 Implementasi Kelas Kelola_User**

| File : Kelola_User | |
|---|---|
| **Nama Kelas** | |
| **Rancangan** | **Implementasi** |
| Kelola_user | class kelola_user extends CI_Controller {} |
| **Atribut** | |
| **Rancangan** | **Implementasi** |

| | |
|---|---|
| **Method** | |
| **Rancangan** | **Implementasi** |
| +__construct()<br>+index()<br>+tambah()<br>+edit()<br>+hapus() | public function __construct() {}<br>public function index() {}<br>public function tambah() {}<br>public function edit() {}<br>public function hapus() {} |

4. Kelas Kelola_Bahanbaku

   Implementasi *Class Diagram* pada Kelas Kelola_Bahanbaku dapat dilihat pada Tabel 4.17.

**Tabel 4.17 Implementasi Kelas Kelola_Bahanbaku**

| File : Kelola_Bahanbaku | |
|---|---|
| **Nama Kelas** | |
| **Rancangan** | **Implementasi** |
| kelola_bahanbaku | class kelola_bahanbaku extends CI_Controller {} |
| **Atribut** | |
| **Rancangan** | **Implementasi** |
| | |
| **Method** | |
| **Rancangan** | **Implementasi** |
| +__construct()<br>+index()<br>+tambah_bb() | public function __construct() {}<br>public function index() {}<br>public function tambah_bb() {} |

5. Kelas Kelola_Barang

   Implementasi *Class Diagram* pada Kelas Kelola_Barang dapat dilihat pada Tabel 4.18.

**Tabel 4.18 Implementasi Kelas Kelola_Barang**

| File : Kelola_Barang | |
|---|---|
| **Nama Kelas** | |
| **Rancangan** | **Implementasi** |
| kelola_barang | class kelola_barang extends |

| | |
|---|---|
| | CI_Controller {} |
| **Atribut** | |
| **Rancangan** | **Implementasi** |
| | |
| **Method** | |
| **Rancangan** | **Implementasi** |
| +__construct()<br>+index()<br>+tambah_barang() | public function __construct() {}<br>public function index() {}<br>public function tambah_barang() {} |

6. Kelas Kelola_Distribusibarang

   Implementasi *Class Diagram* pada Kelas Kelola_Distribusibarang dapat dilihat pada Tabel 4.19.

**Tabel 4.19 Implementasi Kelas Kelola_Distribusibarang**

| File : Kelola_Distribusibarang | |
|---|---|
| **Nama Kelas** | |
| **Rancangan** | **Implementasi** |
| kelola_distribusibarang | class kelola_distribusibarang extends CI_Controller {} |
| **Atribut** | |
| **Rancangan** | **Implementasi** |
| | |
| **Method** | |
| **Rancangan** | **Implementasi** |
| +__construct()<br>+index()<br>+tambah_data()<br>+edit_data()<br>+hapus_data() | public function __construct() {}<br>public function index() {}<br>public function tambah_data() {}<br>public function edit_data() {}<br>public function hapus_data() {} |

7. Kelas ApproveBahanBaku

   Implementasi *Class Diagram* pada KelasApproveBahanBaku dapat dilihat pada Tabel 4.20.

**Tabel 4.20Implementasi KelasApproveBahanBaku**

| File : ApproveBahanBaku | |
|---|---|
| **Nama Kelas** | |
| **Rancangan** | **Implementasi** |
| approvebahanbaku | class approvebahanbaku extends CI_Controller {} |
| **Atribut** | |
| **Rancangan** | **Implementasi** |
| | |
| **Method** | |
| **Rancangan** | **Implementasi** |
| +__construct()<br>+index() | public function __construct() {}<br>public function index() {} |

8. Kelas ApprovePenjualan

   Implementasi *Class Diagram* pada KelasApprovePenjualandapat dilihat pada Tabel 4.21.

**Tabel 4.21 Implementasi Kelas ApprovePenjualan**

| File : ApprovePenjualan | |
|---|---|
| **Nama Kelas** | |
| **Rancangan** | **Implementasi** |
| Approvepenjualan | class approvepenjualanextends CI_Controller {} |
| **Atribut** | |
| **Rancangan** | **Implementasi** |
| | |
| **Method** | |
| **Rancangan** | **Implementasi** |
| +__construct()<br>+index() | public function __construct() {}<br>public function index() {} |

9. Kelas Penjualan

   Implementasi *Class Diagram* pada KelasPenjualandapat dilihat pada Tabel 4.22.

**Tabel 4.22 Implementasi Kelas Penjualan**

| File : Penjualan | |
|---|---|
| **Nama Kelas** | |
| **Rancangan** | **Implementasi** |
| Penjualan | class penjualan extends CI_Controller {} |
| **Atribut** | |
| **Rancangan** | **Implementasi** |
| | |
| **Method** | |
| **Rancangan** | **Implementasi** |
| +__construct()<br>+index()<br>+tambah_penjualan() | public function __construct() {}<br>public function index() {}<br>public function tambah_penjualan() {} |

10. Kelas Peramalan

    Implementasi *Class Diagram* pada Kelas Peramalandapat dilihat pada Tabel 4.23.

**Tabel 4.23 Implementasi KelasPeramalan**

| File : Peramalan | |
|---|---|
| **Nama Kelas** | |
| **Rancangan** | **Implementasi** |
| peramalan | class peramalan extends CI_Controller {} |
| **Atribut** | |
| **Rancangan** | **Implementasi** |
| | |
| **Method** | |
| **Rancangan** | **Implementasi** |
| +__construct()<br>+index()<br>+hitung_peramalan() | public function __construct() {}<br>public function index() {}<br>public function hitung_peramalan() {} |

11. Kelas Lihat_Distribusi_Barang

    Implementasi *Class Diagram* pada Kelas Lihat_Distribusi_Barangdapat dilihat pada Tabel 4.24.

**Tabel 4.24 Implementasi Kelas Lihat_Distribusi_Barang**

| File : LihatDistribusiBarang | |
|---|---|
| **Nama Kelas** | |
| **Rancangan** | **Implementasi** |
| Lihatdistribusibarang | class lihatdistribusibarangS extends CI_Controller {} |
| **Atribut** | |
| **Rancangan** | **Implementasi** |
| | |
| **Method** | |
| **Rancangan** | **Implementasi** |
| +__construct()<br>+index() | public function __construct() {}<br>public function index() {} |

**B. Class Model**

1. Kelas M_Beranda

   Implementasi *Class Diagram* pada Kelas M_Beranda dapat dilihat pada Tabel 4.25.

**Tabel 4.25 Implementasi Kelas M_Beranda**

| File : M_Beranda | |
|---|---|
| **Nama Kelas** | |
| **Rancangan** | **Implementasi** |
| m_beranda | class m_beranda extends CI_Model {} |
| **Atribut** | |
| **Rancangan** | **Implementasi** |
| | |
| **Method** | |
| **Rancangan** | **Implementasi** |
| +__construct()<br>+index() | public function __construct() {}<br>public function index() {} |

2. Kelas M_Login

   Implementasi *Class Diagram* pada Kelas M_Login dapat dilihat pada Tabel 4.26.

**Tabel 4.26 Implementasi Kelas M_Login**

| File : M_Login | |
|---|---|
| **Nama Kelas** | |
| **Rancangan** | **Implementasi** |
| m_login | class m_login extends CI_Model {} |
| **Atribut** | |
| **Rancangan** | **Implementasi** |
| | |
| **Method** | |
| **Rancangan** | **Implementasi** |
| +__construct()<br>+index()<br>+cek_login()<br>+logout() | public function __construct() {}<br>public function index() {}<br>public function cek_login() {<br>public function logout() {} |

3. Kelas M_Kelola_User

   Implementasi *Class Diagram* pada Kelas M_Kelola_User dapat dilihat pada Tabel 4.27.

**Tabel 4.27 Implementasi Kelas M_Kelola_User**

| File : M_Kelola_User | |
|---|---|
| **Nama Kelas** | |
| **Rancangan** | **Implementasi** |
| m_kelola_user | class m_kelola_user extends CI_Model {} |
| **Atribut** | |
| **Rancangan** | **Implementasi** |
| | |
| **Method** | |
| **Rancangan** | **Implementasi** |
| +__construct()<br>+index()<br>+get_all_user()<br>+tambah()<br>+edit()<br>+hapus() | public function __construct() {}<br>public function index() {}<br>public function get_all_user() {}<br>public function tambah() {}<br>public function edit() {}<br>public function hapus() {} |

4. Kelas M_Kelola_Bahanbaku

   Implementasi *Class Diagram* pada Kelas M_Kelola_Bahanbaku dapat dilihat pada Tabel 4.28.

**Tabel 4.28 Implementasi Kelas M_Kelola_Bahanbaku**

| File : M_Kelola_Bahanbaku | |
|---|---|
| **Nama Kelas** | |
| **Rancangan** | **Implementasi** |
| m_kelola_bahanbaku | class m_kelola_bahanbaku extends CI_Model {} |
| **Atribut** | |
| **Rancangan** | **Implementasi** |
| | |
| **Method** | |
| **Rancangan** | **Implementasi** |
| +__construct()<br>+index()<br>+get_all_bb()<br>+tambah() | public function __construct() {}<br>public function index() {}<br>public function get_all_bb() {}<br>public function tambah() {} |

5. Kelas M_Kelola_Barang

   Implementasi *Class Diagram* pada Kelas M_Kelola_Barang dapat dilihat pada Tabel 4.29.

**Tabel 4.29 Implementasi Kelas M_Kelola_Barang**

| File : M_Kelola_Barang | |
|---|---|
| **Nama Kelas** | |
| **Rancangan** | **Implementasi** |
| m_kelola_barang | class m_kelola_barang extends CI_Model {} |
| **Atribut** | |
| **Rancangan** | **Implementasi** |
| | |
| **Method** | |
| **Rancangan** | **Implementasi** |
| +__construct()<br>+index()<br>+get_all_barang() | public function __construct() {}<br>public function index() {}<br>public function get_all_barang() {} |

| | |
|---|---|
| +tambah_barang() | public function tambah_barang() {} |

6. Kelas M_Kelola_Distribusibarang

   Implementasi *Class Diagram* pada Kelas M_Kelola_Distribusibarang dapat dilihat pada Tabel 4.30.

**Tabel 4.30 Implementasi Kelas M_Kelola_Distribusibarang**

| File : M_Kelola_Distribusibarang | |
|---|---|
| **Nama Kelas** | |
| **Rancangan** | **Implementasi** |
| m_kelola_distribusibarang | class m_kelola_distribusibarang extends CI_Model {} |
| **Atribut** | |
| **Rancangan** | **Implementasi** |
| | |
| **Method** | |
| **Rancangan** | **Implementasi** |
| +__construct()<br>+index()<br>+get_all_data()<br>+tambah_data()<br>+edit_data()<br>+hapus_data() | public function __construct() {}<br>public function index() {}<br>public function get_all_data() {}<br>public function tambah_data() {}<br>public function edit_data() {}<br>public function hapus_data() {} |

7. Kelas M_ApproveBahanBaku

   Implementasi *Class Diagram* pada Kelas M_ApproveBahanBaku dapat dilihat pada Tabel 4.31.

**Tabel 4.31 Implementasi Kelas M_ApproveBahanBaku**

| File : M_ApproveBahanBaku | |
|---|---|
| **Nama Kelas** | |
| **Rancangan** | **Implementasi** |
| m_approvebahanbaku | class m_approvebahanbaku extends CI_Model {} |
| **Atribut** | |
| **Rancangan** | **Implementasi** |
| | |

| **Method** | |
|---|---|
| **Rancangan** | **Implementasi** |
| +__construct()<br>+index()<br>+get_all_data() | public function __construct() {}<br>public function index() {}<br>public function get_all_data() {} |

8. Kelas M_ApprovePenjualan

Implementasi *Class Diagram* pada Kelas M_ApprovePenjualandapat dilihat pada Tabel 4.32.

**Tabel 4.32 Implementasi Kelas M_ApprovePenjualan**

| File : M_ApprovePenjualan | |
|---|---|
| **Nama Kelas** | |
| **Rancangan** | **Implementasi** |
| m_approvepenjualan | class m_approvepenjualan extends CI_Model {} |
| **Atribut** | |
| **Rancangan** | **Implementasi** |
| | |
| **Method** | |
| **Rancangan** | **Implementasi** |
| +__construct()<br>+index()<br>+get_all_penjualan | public function __construct() {}<br>public function index() {}<br>public function get_all_penjualan() {} |

9. Kelas M_Penjualan

Implementasi *Class Diagram* pada Kelas M_Penjualandapat dilihat pada Tabel 4.33.

**Tabel 4.33 Implementasi Kelas M_Penjualan**

| File : M_Penjualan | |
|---|---|
| **Nama Kelas** | |
| **Rancangan** | **Implementasi** |
| m_penjualan | class m_penjualan extends CI_Model {} |
| **Atribut** | |
| **Rancangan** | **Implementasi** |
| | |

| **Method** | |
|---|---|
| **Rancangan** | **Implementasi** |
| +__construct()<br>+index()<br>+tambah_penjualan()<br>+get_all_data() | public function __construct() {}<br>public function index() {}<br>public function tambah_penjualan() {}<br>public function get_all_data() {} |

10. Kelas M_Peramalan

Implementasi *Class Diagram* pada KelasM_Peramalandapat dilihat pada Tabel 4.34.

**Tabel 4.34 Implementasi Kelas M_Peramalan**

| File : M_Peramalan | |
|---|---|
| **Nama Kelas** | |
| **Rancangan** | **Implementasi** |
| m_peramalan | class m_peramalan extends CI_Model {} |
| **Atribut** | |
| **Rancangan** | **Implementasi** |
| | |
| **Method** | |
| **Rancangan** | **Implementasi** |
| +__construct()<br>+index()<br>+hitung_peramalan()<br>+get_all_data()<br>+get_data_peramalan() | public function __construct() {}<br>public function index() {}<br>public function hitung_peramalan() {}<br>public function get_all_data() {}<br>public function get_data_peramalan() {} |

11. Kelas M_Lihat_Distribusi_Barang

Implementasi *Class Diagram* pada Kelas M_Lihat_Distribusi_Barangdapat dilihat pada Tabel 4.35.

**Tabel 4.35 Implementasi Kelas M_Lihat_Distribusi_Barang**

| File : M_LihatDistribusiBarang | |
|---|---|
| **Nama Kelas** | |
| **Rancangan** | **Implementasi** |

| m_lihatdistribusibarang | class m_lihatdistribusibarang extends CI_Model {} |
|---|---|
| **Atribut** | |
| **Rancangan** | **Implementasi** |
| | |
| **Method** | |
| **Rancangan** | **Implementasi** |
| +__construct()<br>+index()<br>+get_all_dist() | public function __construct() {}<br>public function index() {}<br>public function get_all_dist() {} |

**C. Class View**

1. Kelas V_Beranda

   Implementasi *Class Diagram* pada Kelas V_Beranda dapat dilihat pada Tabel 4.36.

**Tabel 4.36 Implementasi Kelas V_Beranda**

| File : V_Beranda | |
|---|---|
| **Nama Kelas** | |
| **Rancangan** | **Implementasi** |
| v_beranda | |
| **Atribut** | |
| **Rancangan** | **Implementasi** |
| | |
| **Method** | |
| **Rancangan** | **Implementasi** |
| | |

2. Kelas V_Login

   Implementasi *Class Diagram* pada Kelas V_Login dapat dilihat pada Tabel 4.37.

**Tabel 4.37 Implementasi Kelas V_Login**

| File : V_Login | |
|---|---|
| **Nama Kelas** | |
| **Rancangan** | **Implementasi** |
| v_login | |

| **Atribut** | |
|---|---|
| **Rancangan** | **Implementasi** |
| | |
| **Method** | |
| **Rancangan** | **Implementasi** |
| | |

3. Kelas V_Kelola_User

Implementasi *Class Diagram* pada Kelas V_Kelola_User dapat dilihat pada Tabel 4.38.

**Tabel 4.38 Implementasi Kelas V_Kelola_User**

| File : V_Kelola_User | |
|---|---|
| **Nama Kelas** | |
| **Rancangan** | **Implementasi** |
| v_kelola_user | |
| **Atribut** | |
| **Rancangan** | **Implementasi** |
| | |
| **Method** | |
| **Rancangan** | **Implementasi** |
| | |

4. Kelas V_Kelola_Bahanbaku

Implementasi *Class Diagram* pada Kelas V_Kelola_Bahanbaku dapat dilihat pada Tabel 4.39.

**Tabel 4.39 Implementasi Kelas V_Kelola_Bahanbaku**

| File : V_Kelola_Bahanbaku | |
|---|---|
| **Nama Kelas** | |
| **Rancangan** | **Implementasi** |
| v_kelola_bahanbaku | |
| **Atribut** | |
| **Rancangan** | **Implementasi** |
| | |
| **Method** | |
| **Rancangan** | **Implementasi** |

| | |
|---|---|
| | |

5. Kelas V_Kelola_Barang

Implementasi *Class Diagram* pada Kelas V_Kelola_Barang dapat dilihat pada Tabel 4.40.

**Tabel 4.40 Implementasi Kelas V_Kelola_Barang**

| File : V_Kelola_Barang | |
|---|---|
| **Nama Kelas** | |
| **Rancangan** | **Implementasi** |
| v_kelola_barang | |
| **Atribut** | |
| **Rancangan** | **Implementasi** |
| | |
| **Method** | |
| **Rancangan** | **Implementasi** |
| | |

6. Kelas V_Kelola_Distribusibarang

Implementasi *Class Diagram* pada Kelas V_Kelola_Distribusibarang dapat dilihat pada Tabel 4.41.

**Tabel 4.41 Implementasi Kelas V_Kelola_Distribusibarang**

| File : V_Kelola_Distribusibarang | |
|---|---|
| **Nama Kelas** | |
| **Rancangan** | **Implementasi** |
| v_kelola_distribusibarang | |
| **Atribut** | |
| **Rancangan** | **Implementasi** |
| | |
| **Method** | |
| **Rancangan** | **Implementasi** |
| | |

7. Kelas V_ApproveBahanBaku

Implementasi *Class Diagram* pada Kelas V_ApproveBahanBaku dapat dilihat pada Tabel 4.42.

**Tabel 4.42 Implementasi Kelas Kelola V_ApproveBahanBaku**

| File : V_ApproveBahanBaku | |
|---|---|
| **Nama Kelas** | |
| **Rancangan** | **Implementasi** |
| v_approvebahanbaku | |
| **Atribut** | |
| **Rancangan** | **Implementasi** |
| | |
| **Method** | |
| **Rancangan** | **Implementasi** |
| | |

8. Kelas V_ApprovePenjualan

Implementasi *Class Diagram* pada Kelas V_ApprovePenjualandapat dilihat pada Tabel 4.43.

**Tabel 4.43 Implementasi Kelas V_ApprovePenjualan**

| File : V_ApprovePenjualan | |
|---|---|
| **Nama Kelas** | |
| **Rancangan** | **Implementasi** |
| v_approvepenjualan | |
| **Atribut** | |
| **Rancangan** | **Implementasi** |
| | |
| **Method** | |
| **Rancangan** | **Implementasi** |
| | |

9. Kelas V_Penjualan

Implementasi *Class Diagram* pada Kelas V_Penjualandapat dilihat pada Tabel 4.44.

**Tabel 4.44 Implementasi Kelas V_Penjualan**

| File : V_Penjualan |
|---|
| **Nama Kelas** |

| **Rancangan** | **Implementasi** |
|---|---|
| v_penjualan | |
| **Atribut** | |
| **Rancangan** | **Implementasi** |
| | |
| **Method** | |
| **Rancangan** | **Implementasi** |
| | |

10. Kelas V_Peramalan

Implementasi *Class Diagram* pada KelasV_Peramalandapat dilihat pada Tabel 4.45.

**Tabel 4.45 Implementasi Kelas V_Peramalan**

| File : V_Peramalan | |
|---|---|
| **Nama Kelas** | |
| **Rancangan** | **Implementasi** |
| v_peramalan | |
| **Atribut** | |
| **Rancangan** | **Implementasi** |
| | |
| **Method** | |
| **Rancangan** | **Implementasi** |
| | |

11. Kelas V_Lihat_Distribusi_Barang

Implementasi *Class Diagram* pada Kelas V_Lihat_Distribusi_Barangdapat dilihat pada Tabel 4.46.

**Tabel 4.46 Implementasi Kelas V_Lihat_Distribusi_Barang**

| File : V_LihatDistribusiBarang | |
|---|---|
| **Nama Kelas** | |
| **Rancangan** | **Implementasi** |
| v_lihatdistribusibarang | |
| **Atribut** | |
| **Rancangan** | **Implementasi** |
| | |

| **Method** | |
|---|---|
| **Rancangan** | **Implementasi** |
| | |

### 4.1.4. Implementasi Antar Muka

Implementasi antar muka pada aplikasi *supply chain management* ini adalah sebagai berikut :

**Tabel 4.47 Implementasi Antar Muka Admin**

| Menu | Deskripsi | Nama File |
|---|---|---|
| Form login | Tampilan login pengguna | v_login.php |
| Form Beranda Admin | Tampilan untuk beranda admin | v_admin.php |
| Form Kelola User | Tampilan untuk mengelola data user | v_kelola_user.php |

**Tabel 4.48 Implementasi Antar Muka General Manager**

| Menu | Deskripsi | Nama File |
|---|---|---|
| Form login | Tampilan login pengguna | v_login.php |
| Form Beranda General Manager | Tampilan untuk beranda general manager | v_admin.php |
| Form Approve Bahan Baku | Tampilan untuk mengelola approving data bahan baku | v_approvebahanbaku.php |
| Form | Tampilan untuk mengelola | v_approvepenjualan.php |

| | | |
|---|---|---|
| Approve Penjualan | approving data penjualan | |
| Form Lihat Distribusi | Tampilan untuk melihat data distribusi | v_lihatdistbarang.php |

**Tabel 4.49 Implementasi Antar Muka Kepala Gudang**

| Menu | Deskripsi | Nama File |
|---|---|---|
| Form login | Tampilan login pengguna | v_login.php |
| Form Beranda Kepala Gudang | Tampilan untuk beranda kepala gudang | v_kepgudang.php |
| Form Kelola Bahan Baku | Tampilan untuk mengelola bahan baku | v_kelola_bahanbaku.php |
| Form Peramalan | Tampilan untuk menghitung peramalan | v_peramalan.php |
| Form Hasil Peramalan | Tampilan untuk melihat hasil dari perhitungan peramalan | v_hasil_peramalan.php |

**Tabel 4.50 Implementasi Antar Muka Kepala Pemasaran**

| Menu | Deskripsi | Nama File |
|---|---|---|
| Form login | Tampilan login pengguna | v_login.php |
| Form Beranda Kepala Pemasaran | Tampilan untuk beranda kepala pemasaran | v_keppemasaran.php |

| | | |
|---|---|---|
| Form Kelola Distribusi Barang | Tampilan untuk mengelola data distribusi barang | v_kelola_distribusibarang.php |
| Form Penjualan | Tampilan untuk mengelola penjualan | v_penjualan.php |

**Tabel 4.51 Implementasi Antar Muka Kepala Produksi**

| Menu | Deskripsi | Nama File |
|---|---|---|
| Form login | Tampilan login pengguna | v_login.php |
| Form Beranda Kepala Produksi | Tampilan untuk beranda kepala produksi | v_kepproduksi.php |
| Form Kelola Barang | Tampilan untuk mengelola data barang | v_kelola_barang.php |

## 4.2. Pengujian

Pengujian system dilakukan untuk menguji system dimana pada tahap ini dapat diketahui kesalahan – kesalahan sehingga dapat dilakukan perbaikan dan pengembangan system dan dapat diketahui kesesuaian system dengan kebutuhannya.

Pengujian yang digunakan untuk menguji system yang baru dalah dengan menggunakan metode pengujian *blackbox.*Pada pengujian *blackbox* difokuskan pada persyaratan fungsional perangkat lunak dan pengujian ini berusahan menemukan kesalahan dalam kategori sebagai berikut :

1. Kesalahn *interface*
2. Fungsi – fungsi yang tidak sesuai dengan prosedur
3. Kesalahan dalam struktur data
4. Kesalahan kinerja system

### 4.2.1. Rencana Pengujian Blackbox

Pengujian blackbox yang akan dilakukan pada pengguna system informasi yang dibangun adalah sebagai berikut :

**Tabel 4.52 Rencana Pengujian**

| Kasus Uji | Detail Pengujian | Jenis Pengujian |
|---|---|---|
| Login | Verifikasi data login | Black Box |
| Kelola*user* | Tambah *user* | Black Box |
| | Ubah *user* | Black Box |
| | Hapus *user* | Black Box |
| Approve Bahan Baku | Menampilkan form approve bahan baku | Black Box |
| Approve Penjualan | Menampilkan form approve Penjualan | Black Box |
| Lihat Distribusi Barang | Menampilkan form lihatdistribusi barang | Black Box |
| Kelola Bahan Baku | Tambah Bahan Baku | Black Box |
| Peramalan | Menampilkan form peramalan dan melakukan peramalan data | Black Box |
| Kelola Distribusi Barang | Tambah Distribusi Barang | Black Box |
| | Ubah Distribusi Barang | Black Box |
| | Hapus Distribusi Barang | Black Box |
| Penjualan | Tambah Penjualan | Black Box |
| Kelola Barang | Tambah Barang | Black Box |

### 4.2.2. Kasus dan Hasil Pengujian Blackbox

Berdasarkan pada rencana pengujian blackbox maka dilakukan pengujian dan didapatkan hasil dari pengujian blackbox sebagai berikut:

**1. Pengujian Login**

Berikut dapat dilihat pengujian dari login yang dapat dilihat pada tabel 4.53 :

**Tabel 4.53 Pengujian Login**

| **Kasus dan Hasil Uji (Data Benar)** | | | |
|---|---|---|---|
| **Data Masukan** | **Data yang diharapkan** | **Pengamatan** | **Kesimpulan** |
| Username: sarah<br>Password : sarah | Mengisi data login dan *field* yang tersedia dan berhasil masuk kedalam halaman beranda | Dapat mengisi data login dan dapat masuk ke halaman beranda | [ √ ] Diterima<br>[ ] Ditolak |
| **Kasus dan Hasil Uji (Data Salah)** | | | |
| **Data Masukan** | **Data yang diharapkan** | **Pengamatan** | **Kesimpulan** |
| Username : (kosong)<br>Password : (kosong) | Tampil pesan kesalahan "*please fill out this field*" | Dapat menampilkan pesan kesalahan "*please fill out this field*" | [ √ ] Diterima<br>[ ] Ditolak |

**2. Pengujian Tambah *User***

Berikut dapat dilihat pengujian dari tambah *user* yang dapat dilihat pada tabel 4.54 :

**Tabel 4.54 Pengujian Tambah User**

| **Kasus dan Hasil Uji (Data Benar)** | | | |
|---|---|---|---|
| **Data Masukan** | **Data yang diharapkan** | **Pengamatan** | **Kesimpulan** |
| Username: riki<br>Password : riki | Mengisi data tambah user dan *field* yang tersedia dan berhasil | Dapat mengisi data tambah user dan *field* | [ √ ] Diterima<br>[ ] Ditolak |

| | | | |
|---|---|---|---|
| Nama : riki<br>Bagian : staff<br>Jabatan : staff<br>Hak Akses : admin | memasukkan ke dalam basis data | yang tersedia dan dapat masuk ke halaman beranda | |
| **Kasus dan Hasil Uji (Data Salah)** | | | |
| **Data Masukan** | **Data yang diharapkan** | **Pengamatan** | **Kesimpulan** |
| Username: (kosong)<br>Password : (kosong)<br>Nama : (kosong)<br>Bagian : (kosong)<br>Jabatan : (kosong)<br>Hak Akses : (kosong) | Tampil pesan kesalahan "*please fill out this field*" | Dapat menampilkan pesan kesalahan "*please fill out this field*" | [ √ ] Diterima<br>[ ] Ditolak |

**3. Pengujian Ubah *User***

Berikut dapat dilihat pengujian dari ubah *user* yang dapat dilihat pada tabel 4.55 :

**Tabel 4.55 Pengujian Ubah User**

| **Kasus dan Hasil Uji (Data Benar)** | | | |
|---|---|---|---|
| **Data Masukan** | **Data yang diharapkan** | **Pengamatan** | **Kesimpulan** |
| Username: riki<br>Password : riki<br>Nama : riki<br>Bagian : staff<br>Jabatan : kep.bagian<br>Hak Akses : admin | Mengisi data ubah user dan *field* yang tersedia dan berhasil mengubah data dan memasukkan ke dalam basis data | Dapat mengisi data tambah user dan *field* yang tersedia dan dapat masuk ke halaman beranda | [ √ ] Diterima<br>[ ] Ditolak |
| **Kasus dan Hasil Uji (Data Salah)** | | | |
| **Data Masukan** | **Data yang diharapkan** | **Pengamatan** | **Kesimpulan** |

| | | | |
|---|---|---|---|
| Username: (kosong)<br>Password : (kosong)<br>Nama : (kosong)<br>Bagian : (kosong)<br>Jabatan : (kosong)<br>Hak Akses : (kosong) | Tampil pesan kesalahan "*please fill out this field*" | Dapat menampilkan pesan kesalahan "*please fill out this field*" | [ √ ] Diterima<br>[ ] Ditolak |

**4. Pengujian Hapus *User***

Berikut dapat dilihat pengujian dari hapus*user* yang dapat dilihat pada tabel 4.56 :

**Tabel 4.56 Pengujian Hapus User**

| Kasus dan Hasil Uji (Data Benar) | | | |
|---|---|---|---|
| **Data masukkan** | **Data yang diharapkan** | **Pengamatan** | **Kesimpulan** |
| Memilih tombol "delete" | Memilih tombol hapus, kemudian tampil pertanyaan : "anda yakin menghapus data ini?" jika memilih tombol Hapus, sistem akan menghapus checklist dari basis data | Dapat melakukan proses hapus | [ √ ] Diterima<br>[ ] Ditolak |

**5. Pengujian *Approve* Bahan Baku**

Berikut dapat dilihat pengujian dari approve bahan baku yang dapat dilihat pada tabel 4.57 :

**Tabel 4.57 Pengujian *Approve* Bahan Baku**

| Kasus dan Hasil Uji (Data Benar) | | | |
|---|---|---|---|
| **Data masukkan** | **Data yang diharapkan** | **Pengamatan** | **Kesimpulan** |
| Pengadaan data bahan baku yang akan | Memilih tombol approve, kemudian tampil pertanyaan :"anda yakin | Dapat melakukan proses | [ √ ] Diterima<br>[ ] Ditolak |

| | | | |
|---|---|---|---|
| diadakan | approve data ini?" jika memilih approve, system akan menambahkan data ke basis data | approve dan menyimpan data ke dalam basis data | |

**6. Pengujian *Approve* Penjualan**

Berikut dapat dilihat pengujian dari approve penjualan yang dapat dilihat pada tabel 4.58:

**Tabel 4.58 Pengujian *Approve* Penjualan**

| Kasus dan Hasil Uji (Data Benar) | | | |
|---|---|---|---|
| **Data masukkan** | **Data yang diharapkan** | **Pengamatan** | **Kesimpulan** |
| Data pelanggan yang memesan barang | Memilih tombol approve, kemudian tampil pertanyaan :"anda yakin approve data ini?" jika memilih approve, system akan menambahkan data ke basis data | Dapat melakukan proses approve dan menyimpan data ke dalam basis data | [ √ ] Diterima<br>[ ] Ditolak |

**7. Pengujian Lihat Distribusi**

Berikut dapat dilihat pengujian dari lihat distribusi yang dapat dilihat pada tabel 4.59 :

**Tabel 4.59 Pengujian Lihat Distribusi**

| Kasus dan Hasil Uji (Data Benar) | | | |
|---|---|---|---|
| **Data masukkan** | **Data yang diharapkan** | **Pengamatan** | **Kesimpulan** |
| Data distribusi barang | Menampilkan halaman lihat distribusi berupa report hasil penjualan barang | Dapat menampilkan halaman lihat distribusi | [ √ ] Diterima<br>[ ] Ditolak |

**8. Pengujian Kelola Bahan Baku**

Berikut dapat dilihat pengujian dari kelola bahan baku yang dapat dilihat pada tabel 4.60 :

**Tabel 4.60 Pengujian Kelola Bahan Baku**

| **Kasus dan Hasil Uji (Data Benar)** | | | |
|---|---|---|---|
| **Data Masukan** | **Data yang diharapkan** | **Pengamatan** | **Kesimpulan** |
| Tanggal : 23/07/2016<br>Katun : (kosong)<br>Obat Busa : (2kg)<br>Obat warna: (kosong)<br>Obat aprek: (kosong)<br>Benang : (kosong)<br>Size : (kosong)<br>Rip : (kosong)<br>Plastic : (kosong)<br>Handtag : (kosong) | Mengisi data tambah bahan baku dan field yang tersedia dan berhasil memasukkan data ke dalam basis data. | Dapat mengisi data tambah bahan baku dan field yang tersedia dan berhasil memasukkan data ke dalam basis data. | [ √ ] Diterima<br>[ ] Ditolak |
| **Kasus dan Hasil Uji (Data Salah)** | | | |
| **Data Masukan** | **Data yang diharapkan** | **Pengamatan** | **Kesimpulan** |
| Tanggal : (kosong)<br>Katun : (kosong)<br>Obat Busa : (2kg)<br>Obat warna: (kosong)<br>Obat aprek: (kosong)<br>Benang : (kosong)<br>Size : (kosong)<br>Rip : (kosong) | Tampil pesan kesalahan "*please fill out this field*" | Dapat menampilkan pesan kesalahan "*please fill out this field*" | [ √ ] Diterima<br>[ ] Ditolak |

| | | | |
|---|---|---|---|
| Plastic : | | | |

**9. Pengujian Peramalan**

Berikut dapat dilihat pengujian dari peramalan yang dapat dilihat pada tabel 4.61 :

**Tabel 4.61 Pengujian Peramalan**

| Kasus dan Hasil Uji (Data Benar) | | | |
|---|---|---|---|
| **Data masukkan** | **Data yang diharapkan** | **Pengamatan** | **Kesimpulan** |
| bulan yang akan diramalkan | Melakukan perhitungan peramalan dan menghitung data stock bahan baku untuk hasil peramalan | Dapat melakukan proses peramalan dan menyimpan data ke dalam basis data | [ √ ] Diterima<br>[ ] Ditolak |

**10. Pengujian Tambah Distribusi Barang**

Berikut dapat dilihat pengujian dari tambah distribusi barang yang dapat dilihat pada tabel 4.62 :

**Tabel 4.62Tambah Distribusi Barang**

| Kasus dan Hasil Uji (Data Benar) | | | |
|---|---|---|---|
| **Data Masukan** | **Data yang diharapkan** | **Pengamatan** | **Kesimpulan** |
| Tanggal : 23/07/2016<br>Kota/Tujuan : Cimahi<br>Waktu Pengiriman : 20:00<br>Jenis Produk : Kaos Oblong<br>Jumlah(Lusin) : 20<br>Kendaraan : D | Mengisi data tambah distribusi barang dan field yang tersedia dan berhasil memasukkan data ke dalam basis data. | Dapat mengisi data tambah distribusi barang dan field yang tersedia dan berhasil memasukkan data ke dalam basis data. | [ √ ] Diterima<br>[ ] Ditolak |

| | | | |
|---|---|---|---|
| 1270 VA<br>Supir : Riki<br>Keterangan : Diantar Ke gudang<br>Status : Diterima | | | |
| **Kasus dan Hasil Uji (Data Salah)** | | | |
| **Data Masukan** | **Data yang diharapkan** | **Pengamatan** | **Kesimpulan** |
| Tanggal : 23/07/2016<br>Kota/Tujuan : (Kosong)<br>Waktu Pengiriman : 20:00<br>Jenis Produk : Kaos Oblong<br>Jumlah(Lusin) : 20<br>Kendaraan : D 1270 VA<br>Supir : Riki<br>Keterangan : (Kosong)<br>Status : Diterima | Tampil pesan kesalahan "*please fill out this field*" | Dapat menampilkan pesan kesalahan "*please fill out this field*" | [ √ ] Diterima<br>[ ] Ditolak |

**11. Pengujian Edit Distribusi Barang**

Berikut dapat dilihat pengujian dari edit distribusi barang yang dapat dilihat pada tabel 4.63 :

**Tabel 4.63Edit Distribusi Barang**

| **Kasus dan Hasil Uji (Data Benar)** | | | |
|---|---|---|---|
| **Data Masukan** | **Data yang diharapkan** | **Pengamatan** | **Kesimpulan** |
| Tanggal : 23/07/2016<br>Kota/Tujuan : Bandung<br>Waktu Pengiriman : | Mengubah data distribusi barang dan field yang tersedia dan berhasil memasukkan data ke dalam basis data. | Dapat mengisi data distribusi barang dan field yang tersedia dan berhasil memasukkan | [ √ ] Diterima<br>[ ] Ditolak |

| | | | |
|---|---|---|---|
| 20:00<br>Jenis Produk : Kaos Oblong<br>Jumlah(Lusin) : 20<br>Kendaraan : D 1270 VA<br>Supir : Riki<br>Keterangan : Diantar Ke gudang<br>Status : Diterima | | data ke dalam basis data. | |
| **Kasus dan Hasil Uji (Data Salah)** | | | |
| **Data Masukan** | **Data yang diharapkan** | **Pengamatan** | **Kesimpulan** |
| Tanggal : 23/07/2016<br>Kota/Tujuan : bandung<br>Waktu Pengiriman : 20:00<br>Jenis Produk : Kaos Oblong<br>Jumlah(Lusin) : (kosong)<br>Kendaraan : D 1270 VA<br>Supir : Riki<br>Keterangan : (Kosong)<br>Status : Diterima | Tampil pesan kesalahan "*please fill out this field*" | Dapat menampilkan pesan kesalahan "*please fill out this field*" | [ √ ] Diterima<br>[ ] Ditolak |

**12. Pengujian Penjualan**

Berikut dapat dilihat pengujian penjualan yang dapat dilihat pada tabel 4.64 :

**Tabel 4.64Penjualan Barang**

| **Kasus dan Hasil Uji (Data Benar)** | | | |
|---|---|---|---|
| **Data Masukan** | **Data yang diharapkan** | **Pengamatan** | **Kesimpulan** |
| Atas Nama : Welly | Mengisi data penjualan barang | Dapat mengisi data penjualan | [ √ ] Diterima |

| | | | |
|---|---|---|---|
| Jumlah : 30<br>Tanggal : 23/07/2016<br>Keterangan : Diantar ke gudang | dan field yang tersedia dan berhasil memasukkan data ke dalam basis data. | barang dan field yang tersedia dan berhasil memasukkan data ke dalam basis data. | [ ] Ditolak |
| **Kasus dan Hasil Uji (Data Salah)** | | | |
| **Data Masukan** | **Data yang diharapkan** | **Pengamatan** | **Kesimpulan** |
| Atas Nama : (kosong)<br>Jumlah : 30<br>Tanggal : 23/07/2016<br>Keterangan : Diantar ke gudang | Tampil pesan kesalahan "*please fill out this field*" | Dapat menampilkan pesan kesalahan "*please fill out this field*" | [ √ ] Diterima<br>[ ] Ditolak |

**13. Pengujian Kelola Barang**

Berikut dapat dilihat pengujian kelola barang yang dapat dilihat pada tabel 4.65 :

**Tabel 4.65 Kelola Barang**

| **Kasus dan Hasil Uji (Data Benar)** | | | |
|---|---|---|---|
| **Data Masukan** | **Data yang diharapkan** | **Pengamatan** | **Kesimpulan** |
| Tanggal : 23/07/2016<br>Jumlah : 400Pcs | Mengisi data penjualan barang dan berhasil memasukkan data ke dalam basis data. | Dapat mengisi data penjualan barang dan field yang tersedia dan berhasil memasukkan data ke dalam basis data. | [ √ ] Diterima<br>[ ] Ditolak |

### 4.2.3. Kesimpulan Pengujian Blackbox

Berdasarkan hasil pengujian pada kasus uji yang telah dilakukan maka dapat disimpulkan bahwa secara umum system informasi supply chain management di UD. Next Label Clothing Company dapat berjalan sesuai dengan yang diharapkan.

### 4.2.4. Pengujian Beta

Setelah melakukan pengujian blackbox maka diperlukan juga pengujian beta dimana pengujian beta adalah pengujian yang dilakukan secara langsung ke lapangan dan hasilnya bersifat objektif.

Pengujian beta dilakukan dengan cara interview atau wawancara dengan menggunakan teknik wawancara terstruktur mengenai kepuasan pengguna, dengan menggunakan teknik jawaban terbuka yaitu peneliti memberikan jawaban sepenuhnya kepada orang yang menggunakan system.

Wawancara hanya dilakukan dengan 2 pengguna saja yaitu bagian kepala gudang dan kepala pemasaran tentang kelebihan dan kekurangan system yang dibangun yakni peramalan, pengadaan bahan baku, pendistribusian barang.

Adapun pertanyaaan – pertanyaan yang diajukan dalam proses wawancara pengujian beta system informasi supply chain management di UD. Next Label Clothing Company secara terperinci dapat dilihat pada tabel berikut :

**Tabel 4.66 Pertanyaan pengujian beta**

| Level Pengguna | Pertanyaan |
|---|---|
| Kepala Gudang | 1. Apakah sistem informasi *supply chain management* ini mudah digunakan?<br>Jawaban:<br>Menurut bagian kepala gudang sistem informasi *supply chain management* yang dibangun sederhana dan mudah untuk digunakan.<br><br>2. Apakah tampilan antar muka sistem informasi *supply chain management* ini terlihat menarik?<br>Jawaban:<br>Menurut bagian kepala gudangbahwa sistem informasi *supply chain management* yang dibangun tampilan |

| | |
|---|---|
| | antarmuka sederhana dan menarik.<br><br>3. Apakah sistem informasi *supply chain management* ini memberikan kemudahan pada bagian kepala gudang?<br>Jawaban:<br>Menurut bagian kepala gudang bahwa sistem informasi *supply chain management* yang dibangun sangat memberikan kemudahan untuk bagian kepala gudang dikarenakan system yang dibangun dapat digunakan sebagai acuan dalam pembelian dan pengadaan bahan baku.<br><br>4. Menurut anda, apakah sistem informasi *supply chain management* ini menggunakan bahasa yang mudah untuk dimengerti?<br>Jawaban:<br>Menurut bagian kepala gudang bahwa sistem informasi *supply chain management* yang dibangunsangat mudah untuk dimengerti dikarenakan sistem yang dibangun sederhana dan menggunakan bahasa indonesia sehingga.<br><br>5. Apakah sistem informasi *supply chain management* ini memberikan kemudahan dalam meramalkan pembelian bahan baku bagian gudang?<br>Jawaban:<br>Menurut bagian kepala gudangbahwa sistem informasi *supply chain management* yang dibangun sangat membantu bagian gudang dalam meramalkan pembelian bahan baku. |

| | |
|---|---|
| | 6. Apakah sistem informasi *supply chain management* ini memberikan kemudahan dalam memantau stok bahan baku pada gudang bahan baku?<br>Jawaban:<br>Menurut bagian kepala gudangbahwa sistem informasi *supply chain management* yang dibangun sangat membantu kepala gudangdalam memantau stok bahan baku pada gudang material.<br><br>7. Apakah sistem informasi *supply chain management* yang dibangun mengenai peramalan persediaan bahan baku agar tidak kekurangan maupun kelebihan bahan baku sudah sesuai dengan yang dibutuhkan?<br>Jawaban:<br>Menurut bagian kepala gudangbahwa sistem informasi *supply chain management* yang dibangun mengenai peramalan persediaan bahan baku pada periode sebelumnya yang telah sesuai sistem yang dibangun, maka dapat dipastikan sistem yang dibangun telah cukup akurat dan sesuai dengan kebutuhan kepala gudang. |
| Kepala Pemasaran | 1. Apakah sistem informasi *supply chain management* ini mudah digunakan?<br>Jawaban:<br>Menurut bagian Kepala Pemasaranbahwa sistem informasi *supply chain management* yang dibangun sangat sederhana sehingga mudah untuk digunakan.<br><br>2. Apakah tampilan antar muka sistem informasi *supply chain management* ini terlihat menarik? |

| | |
|---|---|
| | Jawaban:<br>Menurut bagian Kepala Pemasaranbahwa sistem informasi *supply chain management* yang dibangun tampilan antarmuka sistem ini sederhana dan cukup menarik lagi.<br><br>3. Apakah sistem informasi *supply chain management* ini memberikan kemudahan pada bagian pemasaran?<br>Jawaban:<br>Menurut bagian Kepala Pemasaranbahwa sistem informasi *supply chain management* yang dibangun sangat memberikan kemudahan untuk bagian pemasaran dikarenakan data pendistribusian ke pelanggan lebih terstruktur dan jelas pendistribusiannya.<br><br>4. Menurut anda, apakah sistem informasi *supply chain management* ini menggunakan bahasa yang mudah untuk dimengerti?<br>Jawaban:<br>Menurut bagian Kepala Pemasaranbahwa sistem informasi *supply chain management* yang dibangun cukup sederhana sehingga sangat mudah untuk dimengerti.<br><br>5. Apakah Sistem Informasi *supply chain management* ini memberikan kemudahan dalam mendistribusikan barang ?<br><br>Jawaban:<br>Menurut bagian Kepala Pemasaranbahwa sistem |

| | |
|---|---|
| | informasi *supply chain management* yang dibangun cukup membantu dalam proses pendistribusian barang.<br><br>6. Apakah Sistem informasi *supply chain management*ini memberikan kemudahan dalam memantau stok barang di gudang barang?<br>Jawaban:<br>Menurut bagian Kepala Pemasaranbahwa sistem informasi *supply chain management* yang dibangun cukup membantu dalam memantau stok barang yang tersedia di dalam gudang barang. |

Hasil wawancara yang dilakukan terhadap pengguna bagian kepala gudang dan bagian kepala pemasaran adalah bahwa system informasi supply chain management yang dibangun mengenai peramalan pembelian bahan baku sudah cukup benar dan sesuai dengan yang dibutuhkan. Untuk pendistribusian barang kaos oblong sendiri sudah cukup membantu dalam menentukan pengiriman barang ke setiap pelanggan sesuai dengan permintaan dan target produksi. Sedangkan untuk tampilan system yang dibangun sudah cukup menarik dan saran untuk kedepannya aplikasi yang dibangun harus ditambahkan lebih banyak fitur.

### 4.2.5. Kesimpulan Hasil Pengujian Beta

Berdasarkan pengujian beta yang telah dilakukan maka dapat ditarik sebuah kesimpulan bahwa system informasi supply chain management yang dibangun telah memenuhi kebutuhan pengguna yakni bagian kepala gudang dapat terbantu dalam melakukan perhitungan pembelian bahan baku yang harus dibeli oleh bagian gudang agar kegiatan produksi terus berlanjut. Untuk bagian kepala pemasaran terbantu

dalam mendistribusikan barang ke setiap pelanggan sesuai dengan banyaknya permintaan dan pemesanan yang telah ditentukan oleh pelanggan.