

## BAB 6 IMPLEMENTASI

### 6.1 Spesifikasi Pengembangan Sistem

#### 6.1.1 Spesifikasi Perangkat Keras

Pada Tabel 6.1 merupakan spesifikasi perangkat keras yang digunakan dalam pembangunan sistem aplikasi pada PT. Zona Sangangiti Grafika. Beberapa komponen perangkat keras yang digunakan dijabarkan pada tabel ini. Sistem dikembangkan menggunakan sebuah laptop Sony Vaio SVF-14218SGB. Beberapa komponen dari laptop ini yaitu *processor* Intel Core I5 3337U, *hard disk drive* berukuran 500GB, serta RAM 4GB DDR3L.

**Tabel 6.1 Spesifikasi perangkat keras**

Nama Komponen	Spesifikasi
<i>Processor</i>	Intel Core I5 3337U
<i>Hard Disk Drive</i>	500GB
RAM	8GB DDR3L
<i>System Model</i>	Sony Vaio SVF-14218SGB

#### 6.1.2 Spesifikasi Perangkat Lunak

Pada Tabel 6.2 merupakan spesifikasi perangkat lunak yang digunakan dalam pengembangan sistem aplikasi pada PT. Zona Sangangiti Grafika. Sebagai sistem operasi, digunakan Windows 10 Home 64-bit. Bahasa pemrograman yang digunakan yaitu PHP. Sebagai *text editor* digunakan netbeans IDE 8.2 serta untuk *server* disediakan oleh XAMPP.

**Tabel 6.2 Spesifikasi perangkat lunak**

Nama Komponen	Spesifikasi
<i>Operating System</i>	Windows 10 Home 64-bit
<i>Programming Language</i>	PHP
<i>Text Editor</i>	Netbeans IDE 8.2
<i>Server</i>	XAMPP

## 6.2 Implementasi Algoritma

Pada sub bab ini dijelaskan beberapa hasil implementasi algoritma sesuai dengan yang telah dirancang pada perancangan algoritma. Algoritma tersebut yaitu algoritma membuat urutan produksi, algoritma menghitung beban kerja, dan algoritma mencatat status pesanan. Proses kerja dari setiap algoritma akan dijelaskan secara umum. Kode program juga akan dijabarkan pada sub bab ini.

```
1 $kode_pesanan = addslashes(strip_tags(filter_input(INPUT_POST,
2 'kode_pesanan')));
3     $jenis_mesin1 =
4 addslashes(strip_tags(filter_input(INPUT_POST, 'jenis_mesin1')));
5     $kode_mesin1 = $this->model5->getBebanMin($jenis_mesin1);
6     $durasi1 = addslashes(strip_tags(filter_input(INPUT_POST,
7 'durasi1')));
8     $jenis_mesin2 =
9 addslashes(strip_tags(filter_input(INPUT_POST, 'jenis_mesin2')));
10    $kode_mesin2 = $this->model5->getBebanMin($jenis_mesin2);
11    $durasi2 = addslashes(strip_tags(filter_input(INPUT_POST,
12 'durasi2')));
13    $kode_mesin3 = 0;
14    $durasi3 = 0;
15    $urutan_mesin1 = $this->model7-
16 >getUrutanBerdasarkanMesin($kode_mesin1);
17    $urutan_mesin2 = $this->model7-
18 >getUrutanBerdasarkanMesin($kode_mesin2);
19    $get_pekerja = $this->model->bebanPekerjaTerendah(2);
20    $beban_barul = $get_pekerja[0]->beban + $durasi1;
21    $update_beban1 = $this->model->ubahBeban($get_pekerja[0]-
22 >id, $beban_barul);
23    $beban_baru2 = $get_pekerja[1]->beban + $durasi2;
24    $update_beban2 = $this->model->ubahBeban($get_pekerja[1]-
25 >id, $beban_baru2);
26    $num1 = count($urutan_mesin1);
27    $ready = "R";
28    $wait = "W";
29    $proses = "P";
30    if ($num1 == 0) {
31        $cek_P = $this->model7->cekStatusP($kode_mesin1);
32        if ($cek_P == 0) {
33            $insert_urutan = $this->model7-
34 >create($kode_pesanan, $kode_mesin1, $durasi1, 1, 1,
35 $get_pekerja[0]->id, $proses);
36            $update_mesin = $this->model5-
37 >updateKodePesanan($kode_pesanan, $kode_mesin1);
38        } else {
39            $insert_urutan = $this->model7-
40 >create($kode_pesanan, $kode_mesin1, $durasi1, 1, 1,
41 $get_pekerja[0]->id, $ready);
42        }
43    } else if ($durasi1 <= $urutan_mesin1[0]->durasi) { //jika
44 durasi pesanan baru < dari durasi pesanan urutan teratas
45     $urutan = 1;
46     $i = 2;
47     foreach ($urutan_mesin1 as $m1) {
48         $urutanBaru = $this->model7->ubahUrutan($m1->id,
49 $i);
50         $i++;
51     }
52     $insert_urutan = $this->model7->create($kode_pesanan,
53 $kode_mesin1, $durasi1, $urutan, 1, $get_pekerja[0]->id, $ready);
54 } else {
55
```

```

56     $get_urutan = $this->model7-
57 >getUrutanDenganDurasi($kode_mesin1, $durasil);
58     $jumlah = count($get_urutan);
59     if ($jumlah == 0) {
60         $urutan_terakhir = $this->model7-
61 >getUrutanTerakhirMesin($kode_mesin1) + 1;
62         $insert_urutan = $this->model7-
63 >create($kode_pesanan, $kode_mesin1, $durasil, $urutan_terakhir,
64 1, $get_pekerja[0]->id, $ready);
65     } else {
66         $urutan = $get_urutan[0]->urutan;
67         $i = $urutan;
68         foreach ($get_urutan as $m1) {
69             $ubah_urutan = $this->model7->ubahUrutan($m1-
70 >id, $i + 1);
71             $i++;
72         }
73         $insert_urutan = $this->model7-
74 >create($kode_pesanan, $kode_mesin1, $durasil, $urutan, 1,
75 $get_pekerja[0]->id, $ready);
76     }
77 }
78     $beban_mesin1 = $this->model5->getBeban($kode_mesin1);
79     $beban_baru1 = $beban_mesin1 + $durasil;
80     $beban_mesin2 = $this->model5->getBeban($kode_mesin2);
81     $beban_baru2 = $beban_mesin2 + $durasil2;
82     $update_beban_mesin1 = $this->model5-
83 >updateBeban($kode_mesin1, $beban_baru1);
84     $update_beban_mesin2 = $this->model5-
85 >updateBeban($kode_mesin2, $beban_baru2);
86     $status = "produksi";
87     $update_status_pesanan = $this->model3-
88 >updateStatusPesanan($kode_pesanan, $status);
89     $deskripsi = "Pesanan sedang dalam proses produksi";
90     $update_tracking = $this->model9-
91 >inputTracking($kode_pesanan, $deskripsi);
92     $insert = $this->model8->create($kode_pesanan,
93 $kode_mesin1, $kode_mesin2, $kode_mesin3);
94     echo "<script> document.location =
95 'index.php?page=pesanan';</script>";

```

### Kode program 6.1 Implementasi algoritma membuat urutan produksi

Kode program 6.1 merupakan algoritma dalam proses membuat urutan produksi. Pada C\_Direktur, terdapat fungsi yang bertugas mengatur urutan pembuatan pesanan untuk setiap mesin. Kode dari fungsi ini dapat dilihat pada tabel diatas. Fungsi ini menerima masukan dari *user* berupa mesin yang digunakan serta durasi untuk setiap mesin. Algoritma di dalam fungsi ini berjalan berdasarkan durasi pengerjaan. Sesuai dengan metode SJF, maka urutan pesanan untuk setiap mesin akan diurutkan mulai dari durasi terkecil hingga terbesar. Selain itu, juga terdapat proses penentuan pekerja yang menangani pesanan, penentuakn mesin yang menangani pesanan, penambahan beban untuk karyawan, penambahan beban untuk mesin, serta penyimpanan data urutan mesin yang digunakan untuk memproses pesanan.

```

1  $waktu_produkatif = addslashes(strip_tags(filter_input(INPUT_POST,
2  'waktu_produkatif')));
3      $allowance_persen =
4  addslashes(strip_tags(filter_input(INPUT_POST, 'allowance')));
5      $allowance = ($allowance_persen / 100) * $waktu_produkatif;
6      $total_jam_kerja =
7  addslashes(strip_tags(filter_input(INPUT_POST, 'total_jam')));
8
9      $FTE = ($waktu_produkatif + $allowance) / $total_jam_kerja;
10
11     if ($FTE > 5.12) {
12         echo "<script> window.alert('Butuh penambahan 4 orang
13         tenaga kerja');</script>";
14     } else if ($FTE > 3.84 && $FTE <= 5.12) {
15         echo "<script> window.alert('Butuh penambahan 3 orang
16         tenaga kerja');</script>";
17     } else if ($FTE > 2.56 && $FTE <= 3.84) {
18         echo "<script> window.alert('Butuh penambahan 2 orang
19         tenaga kerja');</script>";
20     } else if ($FTE > 1.28 && $FTE <= 2.56) {
21         echo "<script> window.alert('Butuh penambahan 1 orang
22         tenaga kerja');</script>";
23     } else if ($FTE > 1 && FTE <= 1.28) {
24         echo "<script> window.alert('Beban untuk divisi ini
25         NORMAL');</script>";
26     } else if ($FTE > 0 && FTE <= 1) {
27         echo "<script> window.alert('Beban untuk divisi ini
28         UNDERLOAD');</script>";
29     }
30     echo "<script> document.location =
31     'index.php?page=bebankerja';</script>";
32

```

### Kode program 6.2 Implementasi algoritma menghitung beban kerja

Kode program 6.2 merupakan algoritma yang digunakan dalam proses menghitung beban kerja. Pertama, user diminta memasukkan data yang dibutuhkan dalam perhitungan beban kerja. Data yang diminta yaitu total waktu produktif untuk setiap divisi, persentase *allowance* yaitu waktu yang dibutuhkan pekerja secara pribadi seperti beristirahat sejenak, sholat, dan sebagainya serta total jam kerja yang seharusnya. Ketiga data tersebut akan dihitung menggunakan rumus FTE. Hasil perhitungan akan dikonversi menjadi usulan atau pemberitahuan mengenai status beban kerja pada divisi yang bersangkutan.

```

1  $data = $this->modell->viewDetailPesanan($id);
2      $pesanan_id = $data->id;
3      $nama = $_SESSION['nama'];
4      if($_SESSION['role']=="marketing"){
5          $deskripsi = "Pesanan sudah dicatat oleh ". $nama;
6          $update_status_pesanan = $this->modell->
7          >updateStatusPesanan($id, "desainer");
8      }
9      if($_SESSION['role']=="direktur"){
10         $deskripsi = "Pesanan sedang dalam proses desain";
11     }
12     else if($_SESSION['role']=="pelanggan" ){
13         $deskripsi = "Pesanan sudah diterima";
14         $update_status_pesanan = $this->modell->
15         >updateStatusPesanan($id, "selesai");
16     }
17     else if($_SESSION['role']=="desainer"){
18         $deskripsi = "Pesanan sedang diproduksi oleh " . $nama;
19     }

```

```

20         $update_status_pesanan = $this->modell1-
21         >updateStatusPesanan($id, "direktur");
22     }
23     else if($_SESSION['role']=="produksi"){
24         $deskripsi = "Pesanan sedang dalam pengiriman";
25     }
26     $insert = $this->model->inputTracking($id, $deskripsi);
27
28     if ($_SESSION['role'] == "pelanggan") {
29         echo "<script> document.location =
30         'index.php?page=pesanansaya';</script>";
31     }
32     else {
33         echo "<script> document.location =
34         'index.php?page=pesanan';</script>";
35     }

```

### Kode program 6.3 Implementasi algoritma mencatat status pesanan

Kode program 6.3 merupakan implementasi algoritma dari mencatat status pesanan. Data pesanan yang dipilih akan diambil dari *database* untuk digunakan dalam proses mencatat status pesanan. Pencatatan pesanan dilakukan berdasarkan posisi dari aktor didalam sistem. Status pesanan akan tercatat secara otomatis sehingga aktor tidak perlu menuliskannya secara manual. Status pesanan akan disimpan dalam sebuah variabel bernama deskripsi. Isi dari variabel ini berbeda-beda untuk setiap aktor sesuai posisinya masing-masing. Setelah itu, data status pesanan akan disimpan kedalam *database* dan akan ditampilkan pada halaman status pesanan dari pelanggan yang bersangkutan.

## 6.3 Implementasi Antarmuka

Gambar 6.1 Halaman awal

Gambar 6.1 merupakan halaman awal dari sistem. halaman awal ini berisi fitur *login* serta fitur pendaftaran bagi calon pelanggan. Fitur *login* berfungsi untuk menentukan hak akses serta fitur-fitur apa saja yang dapat digunakan oleh setiap pengguna. Terdapat 7 hak akses yang ada dalam sistem, yaitu pelanggan, *guest*, marketing, desainer, akuntan, produksi, dan direktur. Sedangkan fitur pendaftaran digunakan bagi calon pelanggan agar dapat mengakses fitur pesan sehingga dapat berkonsultasi dengan marketing.

Zona Sangangiti Grafika marketing

- Pesan Masuk
- Pesan Keluar
- Pelanggan
- Pesanan**
- Jadwal

### Daftar Pesanan

+ Tambah

Show  entries Search:

Kode	Pesanan	Pelanggan	Harga	Sisa	Pengerjaan	Status	Aksi
ZONA0001	Kalender	Budi Rahayu	Rp 100.000,00	Rp 0,00	1 Hari	Diproses	<span style="color: green;">+ Status</span> <span style="color: blue;">Ubah</span> <span style="color: red;">✖ Hapus</span>
ZONA0002	Nota	Ani Nurhayati	Rp 300.000,00	Rp 25.000,00	5 Hari	Selesai	<span style="color: red;">✖ Hapus</span>
ZONA0003	Banner	Budi Rahayu	Rp 300.000,00	Rp 300.000,00	3 Hari	Selesai	<span style="color: red;">✖ Hapus</span>

Showing 1 to 3 of 3 entries Previous **1** Next

**Gambar 6.2** Fitur mengelola data pesanan

Gambar 6.2 merupakan tampilan dari halaman daftar pesanan. Pada halaman ini marketing dapat melakukan pengelolaan terhadap data pesanan. Marketing dapat membuat, mengubah, dan menghapus data pesanan. Untuk membuat pesanan dapat diakses melalui tombol tambah, mengubah pesanan dapat diakses melalui tombol ubah, dan menghapus pesanan dapat diakses melalui tombol hapus.

Zona Sangangiti Grafika akuntan

- Pesan Masuk
- Pesan Keluar
- Pelanggan
- Pesanan**
- Jadwal
- Keuangan

### Daftar

Apakah anda ingin menambahkan status pesanan?
 Keluar Ya

Show  entries Search:

Kode	Pesanan	Pelanggan	Harga	Sisa	Pengerjaan	Status	Aksi
ZONA0001	Kalender	Budi Rahayu	Rp 100.000,00	Rp 100.000,00	1 Hari	Diproses	<span style="color: green;">+ Status</span> <span style="color: green;">Bayar</span>
ZONA0002	Nota	Ani Nurhayati	Rp 300.000,00	Rp 25.000,00	5 Hari	Diproses	<span style="color: green;">+ Status</span> <span style="color: green;">Bayar</span>
ZONA0003	Banner	Budi Rahayu	Rp 300.000,00	Rp 300.000,00	3 Hari	Diproses	<span style="color: green;">+ Status</span> <span style="color: green;">Bayar</span>

Showing 1 to 3 of 3 entries Previous **1** Next

**Gambar 6.3** Fitur *input* status pesanan

Gambar 6.3 merupakan tampilan untuk mencatat status pesanan. Fitur ini dapat diakses oleh marketing, desainer, produksi, serta akuntan. Fitur ini dapat diakses melalui tombol status yang terdapat pada halaman daftar pesanan. Status pesanan yang telah dicatat akan ditampilkan ke pengguna yang bersangkutan.

Zona Sangangiti Grafika desainer

Pesan Masuk  
Pesan Keluar  
Pesanan  
Jadwal

### Daftar Pesanan

Show 10 entries Search:

Kode	Pesanan	Pelanggan	Harga	Sisa	Pengerjaan	Status	Desain	Aksi
ZONA0001	Kalender	Budi Rahayu	Rp 100.000,00	Rp 0,00	1 Hari	Diproses	 	<a href="#">+ Status</a>
ZONA0002	Nota	Ani Nurhayati	Rp 300.000,00	Rp 25.000,00	5 Hari	Selesai		
ZONA0003	Banner	Budi Rahayu	Rp 300.000,00	Rp 300.000,00	3 Hari	Selesai		

Showing 1 to 3 of 3 entries Previous **1** Next

**Gambar 6.4** Fitur untuk kelola desain pesanan

Gambar 6.4 merupakan fitur untuk kelola desain pesanan. Terdapat dua tombol dalam mengelola desain pesanan, yaitu *download* dan *upload*. Desainer dan produksi dapat mengunduh desain dari pesanan menggunakan fitur *download* yang dapat diakses melalui tombol dengan simbol *download*. Sedangkan untuk fitur *upload*, hanya dapat digunakan oleh desainer dengan menekan tombol bersimbol *upload*.

Zona Sangangiti Grafika akuntan

Pesan Masuk  
Pesan Keluar  
Pelanggan  
Pesanan  
Jadwal  
Keuangan

### Daftar

Show 10 Search:

#### Pembayaran

Kode Pesanan

Pesanan

Kode Pelanggan

Nominal

Kode	Pesanan	Pelanggan	Harga	Sisa	Pengerjaan	Status	Aksi
ZONA0001	Kalender	Budi Rahayu	Rp 100.000,00	Rp 0,00	1 Hari	Diproses	<a href="#">+ Status</a>
ZONA0002	Nota	Ani Nurhayati	Rp 300.000,00	Rp 25.000,00	5 Hari	Selesai	<a href="#">Bayar</a>
ZONA0003	Banner	Budi Rahayu	Rp 300.000,00	Rp 300.000,00	3 Hari	Selesai	<a href="#">Bayar</a>

Showing 1 to 3 of 3 entries Previous **1** Next

**Gambar 6.5** Fitur pembayaran

Gambar 6.5 merupakan tampilan dari fitur pembayaran. Fitur ini dapat diakses oleh akuntan pada halaman daftar pesanan. *Form* akan muncul setelah akuntan menekan tombol bayar. Setelah akuntan mengisi *form* pembayaran, maka halaman daftar pesanan akan kembali ditampilkan. Untuk pesanan yang telah lunas, maka tombol bayar tidak dapat diakses pada pesanan tersebut.



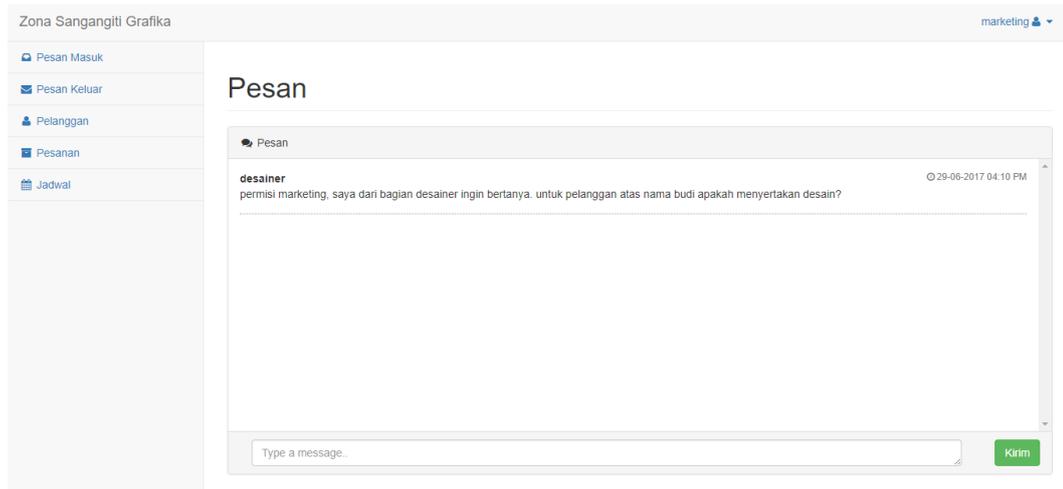
**Gambar 6.6 Tampilan halaman pesan keluar**

Gambar 6.6 merupakan tampilan dari halaman pesan keluar. Halaman ini dapat diakses oleh seluruh pengguna yang sudah masuk ke dalam sistem. Pada halaman ini, pengguna dapat melihat daftar pesan yang telah ia kirimkan kepada pengguna lain. Selain itu, pengguna juga dapat mengirimkan pesan baru, menghapus pesan yang telah dikirimkan, serta melihat detail dari pesan tersebut.



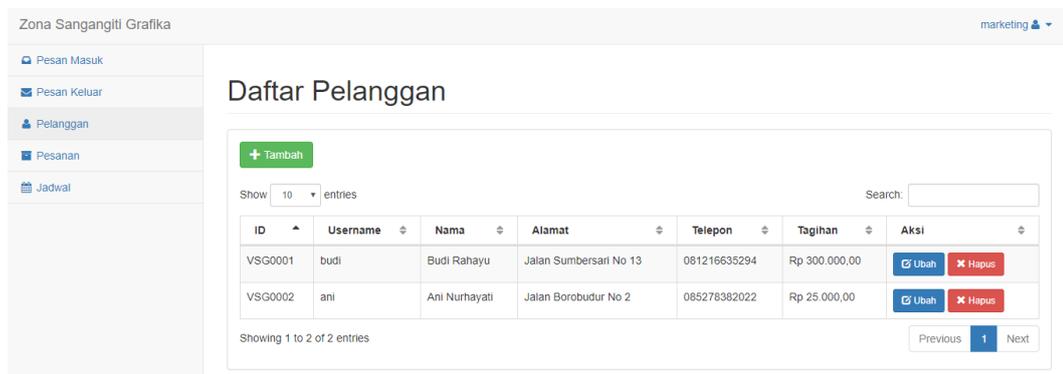
**Gambar 6.7 Tampilan halaman pesan masuk**

Gambar 6.7 merupakan tampilan dari halaman pesan masuk. Pada halaman ini pengguna dapat melihat daftar pesan yang ditujukan kepadanya. Data yang ditampilkan yaitu nama pengirim, isi pesan, tanggal pesan masuk, serta status dari pesan. Status ini disesuaikan terhadap penerima, apakah penerima telah membaca atau belum. Selain itu, pengguna juga dapat melihat detail dari pesan tersebut.



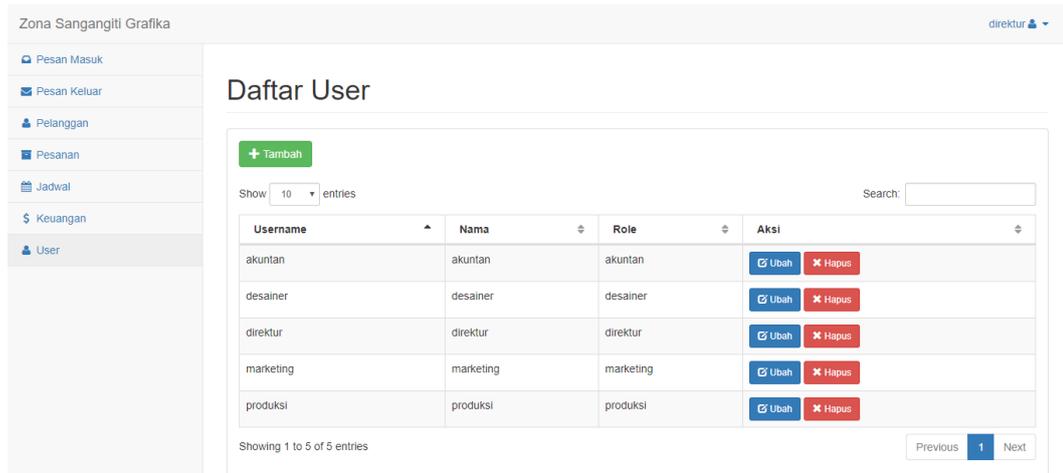
**Gambar 6.8 Tampilan halaman detail pesan**

Gambar 6.8 merupakan tampilan halaman detail pesan. Pada halaman ini, pengguna dapat melihat detail percakapan dari sebuah pesan. Data yang ditampilkan yaitu isi percakapan, pengirim, serta tanggal dari setiap pesan. Selain itu, pengguna juga dapat membalas pesan tersebut pada *form* dibawah detail pesan.



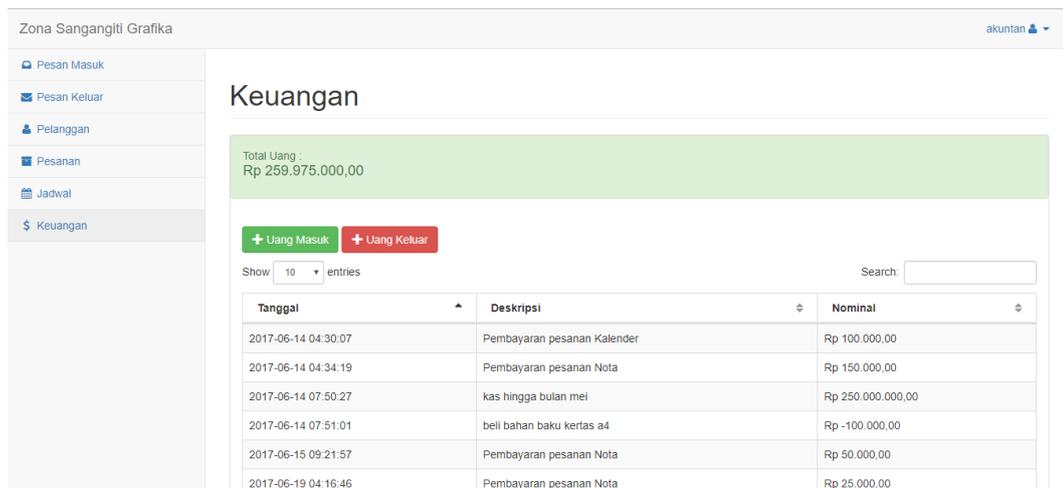
**Gambar 6.9 Fitur kelola pelanggan**

Gambar 6.9 merupakan tampilan dari halaman daftar pelanggan. Pada halaman ini, marketing dapat melakukan pengelolaan terhadap data pelanggan. Pengelolaan data pelanggan dapat dilakukan dengan menekan tombol yang terdapat pada halaman ini. Marketing dapat menambah, menghapus, atau mengubah data pelanggan sesuai dengan tombol yang dipilih.



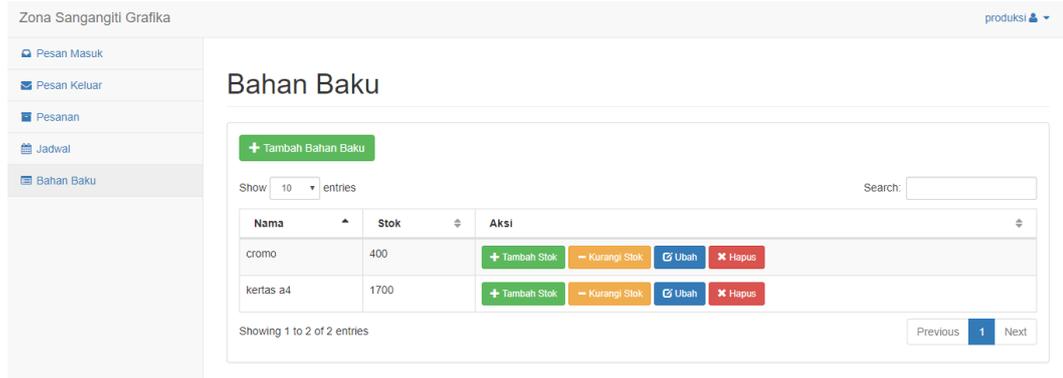
**Gambar 6.10 Tampilan halaman user**

Gambar 6.10 merupakan tampilan dari halaman daftar *user*. Halaman ini hanya dapat diakses oleh direktur. Direktur dapat mengelola *user* yang pada halaman ini, seperti menambah, mengubah, atau menghapus *user*. Setiap fitur dapat diakses melalui tombol yang terdapat pada halaman ini.



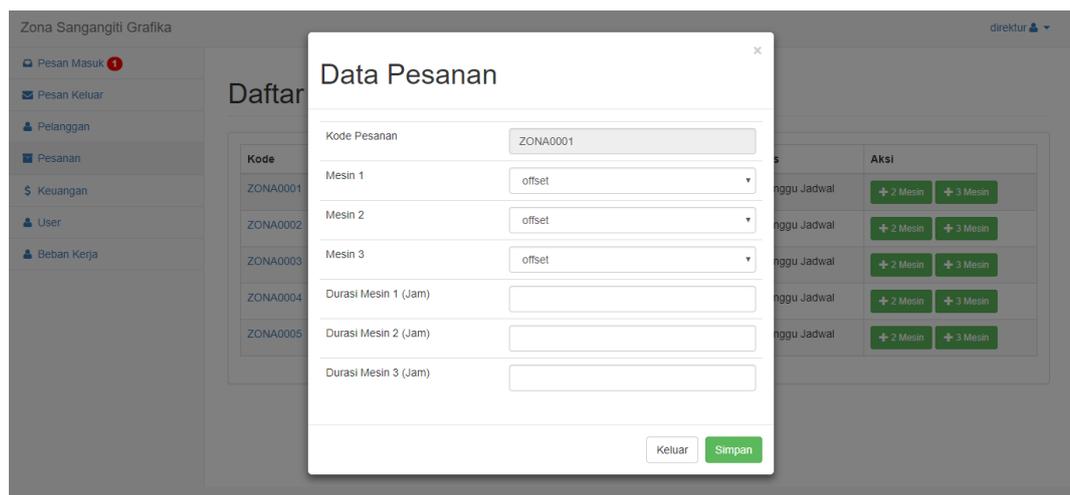
**Gambar 6.11 Tampilan halaman keuangan**

Gambar 6.11 merupakan tampilan dari halaman keuangan. Halaman ini menampilkan seluruh data keuangan mulai dari pembayaran pesanan, uang masuk, hingga pengeluaran perusahaan. Pada halaman ini, akuntan dapat memasukkan data uang masuk melalui tombol uang masuk dan data uang keluar melalui tombol uang keluar.



**Gambar 6.12 Tampilan halaman bahan baku**

Gambar 6.12 merupakan tampilan dari halaman bahan baku. Halaman ini hanya dapat diakses oleh produksi. Pada halaman ini, produksi dapat mengelola data bahan baku seperti menambah, mengubah, mengurangi, serta menghapus. Pengelolaan data bahan baku dapat dilakukan dengan menekan tombol yang terdapat pada halaman ini sesuai tujuannya masing-masing.



**Gambar 6.13 Tampilan form antrian**

Gambar 6.13 merupakan tampilan dari form ketika aktor akan membuat antrian produksi. Form ini hanya dapat diakses oleh direktur. Pada form ini, direktur harus mengisi mesin yang akan digunakan dalam proses pembuatan suatu pesanan. Selain itu, direktur juga harus mengisi durasi penggunaan untuk setiap mesin tersebut sehingga nantinya dapat diurutkan mulai dari durasi terkecil hingga durasi terbesar.

Zona Sangangiti Grafika direktur

Pesanan Masuk 1

Pesanan Keluar

Pelanggan

Pesanan

Keuangan

User

Beban Kerja

### Perhitungan Beban Kerja

Waktu Produktif (Menit)

Allowance (Persen)

Total Jam Kerja (Menit)

**Gambar 6.14** Tampilan form perhitungan beban kerja

Gambar 6.14 merupakan tampilan dari form ketika aktor ingin menghitung beban kerja untuk sebuah divisi. Form ini hanya dapat diakses oleh direktur. Pada form ini, direktur harus mengisi beberapa data yang nantinya akan digunakan untuk perhitungan beban kerja. Data yang harus diisi yaitu waktu produktif kerja dari divisi tersebut, waktu *allowance* yaitu waktu yang dibutuhkan pekerja secara pribadi seperti beristirahat sejenak, sholat, dan sebagainya. Selain itu, data total jam kerja juga harus diisi. Setelah direktur mengisi form dan menekan tombol hitung, maka sistem akan melakukan perhitungan dan menampilkan notifikasi kondisi beban kerja untuk divisi tersebut.