

Laporan Project Akhir Semester
Struktur Data
Antrian Bazaar Buku dengan Priority Queue
Dan Searching Array



Oleh:

Abdul Aziz Fakhrul Soleh - 1855301079

Farahdilla Anisa Putri – 1855301043

Jean - 1855301068

M. Hasbi Assidiq - 1855301084

Pengampu :

Maksum Ro'is Adin Saf, S.Kom., M.Eng

Susiyanti., S.ST

Program Studi Teknik Informatika

Jurusan Teknologi Informasi

Politeknik Caltex Riau

Tahun Ajaran : 2019/2020

1. Deskripsi Sistem dan Implementasi Materi SDA

Sistem ini adalah sebagai bentuk simulasi dari Antrian Bazaar Buku menggunakan prinsip priority Queue dan Searching array, **Priority queue** ini dalam implementasinya adalah untuk fungsi antrian antara pelanggan non-member dan member, maka pelanggan member memiliki prioritas lebih dibanding non-member , maka apabila jika ada pelanggan non-member yang lebih dahulu terdaftar dalam registrasi , maka jika ada pelanggan member akan dijalankan terlebih dahulu. Pada antrian ini juga menggunakan timer dengan inputan waktu pada saat registrasi. Lalu untuk implementasi searching array digunakan untuk mencari buku bazaar pada list buku yang sudah tersedia, list buku menggunakan array list karena menyimpan lebih dari satu tipe data. Jadi ketika registrasi sudah dilakukan sekaligus menginputkan buku yang dicari maka ketika sudah masuk ke antrian bazaar , akan dilakukan **searching array** menggunakan metode sequential search dan akan menampilkan data buku ditemukan atau tidak. Pada sistem ini juga terdapat fitur input buku untuk menambah daftar buku pada list buku bazaar.

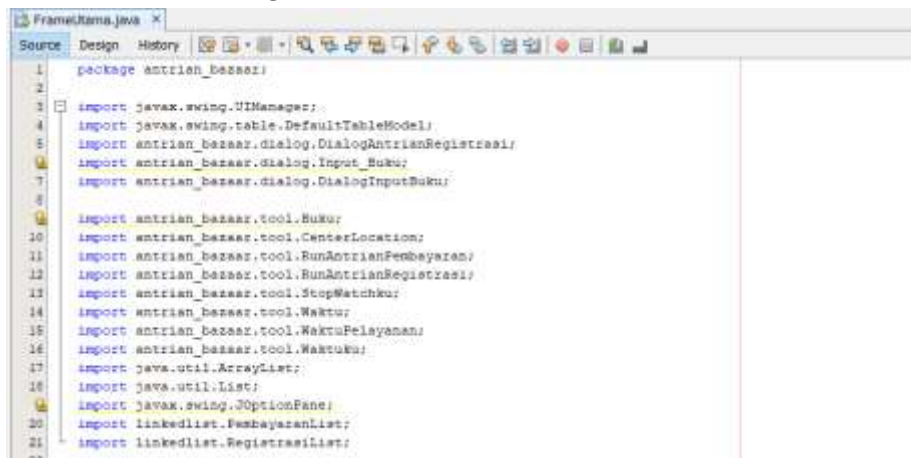
2. Bedah Kode Program

Berikut adalah bedah program untuk metode priority Queue

A. Package antrian_bazaar

1. Class FrameUtama

1) Koding



```
1 package antrian_bazaar;
2
3 import javax.swing.UIManager;
4 import javax.swing.table.DefaultTableModel;
5 import antrian_bazaar.dialog.DialogAntrianRegistrasi;
6 import antrian_bazaar.dialog.Input_Buku;
7 import antrian_bazaar.dialog.DialogInputBuku;
8
9 import antrian_bazaar.tool.Buku;
10 import antrian_bazaar.tool.CenterLocation;
11 import antrian_bazaar.tool.RunAntrianPembayaran;
12 import antrian_bazaar.tool.RunAntrianRegistrasi;
13 import antrian_bazaar.tool.StopWatchku;
14 import antrian_bazaar.tool.Waktu;
15 import antrian_bazaar.tool.WaktuPelayanan;
16 import antrian_bazaar.tool.WaktuKu;
17 import java.util.ArrayList;
18 import java.util.List;
19 import javax.swing.JOptionPane;
20 import linkedlist.PembayaranList;
21 import linkedlist.RegistrasiList;
```

```

FrameUtama.java x
Source Design History
23 public class FrameUtama extends javax.swing.JFrame {
24
25     Stopwatchku stopWatch;
26     Waktu w[];
27     public static boolean runOto = true;
28     public static int jumlahAntrianTeller = 0;
29     public static int waktuMulaiTeller = 10;
30     int flag = 0;
31
32     public FrameUtama() {
33         initComponents();
34         //ini ubah jTable1
35         RunAntrianRegistrasi run = new RunAntrianRegistrasi(this.jTable2, this.jTable3, this.jTable4);
36         RunAntrianPembayaran run1 = new RunAntrianPembayaran(this.jTable3, this.jTable4, this.jTable1);
37         // RunAntrianCS run1 = new RunAntrianCS(this.jTable4, this.jTable1);
38
39         DefaultTableModel m = (DefaultTableModel) this.jTable4.getModel();
40         List<Object> list = new ArrayList<>();
41         list.add(new Object[]{"B-001", "SDA", 52000});
42         list.add(new Object[]{"B-002", "RPL", 60000});
43         list.add(new Object[]{"B-003", "DW", 53500});
44         list.add(new Object[]{"B-004", "WEB", 40000});
45         list.add(new Object[]{"B-005", "JKD", 35000});
46         list.add(new Object[]{"B-006", "MTK", 30000});
47
48         for (Object obj : list) {
49             m.addRow((Object[]) obj);
50         }
51
52         DefaultTableModel m1 = (DefaultTableModel) this.jTable1.getModel();
53         List<Object> list1 = new ArrayList<>();
54         list1.add(new Object[]{"Abdul Aziz Fakhru", "1855301079"});
55         list1.add(new Object[]{"Farahdilla Anisa", "1855301043"});
56         list1.add(new Object[]{"Jean", "1855301068"});
57         list1.add(new Object[]{"M. Hasbi", "1855301084"});
58
59         for (Object obj1 : list1) {
60             m1.addRow((Object[]) obj1);
61         }
62
63     }

```

Pada Class Frame Utama ini di panggil class RunAntrianRegistrasi (variable run) dengan parameter jTable2, jTable3, jTable4, akan dijelaskan pada bagian class RunAntrianRegistrasi. runAntrianPembayaran pada proyek ini tidak dipergunakan. Kemudian isi dari jTable4 adalah list buku yang ada dibazar, list ini menggunakan arraylist untuk menyimpan data buku yang terdiri dari tipe data yang berbeda. Bagian ini dibuat secara manual karena data tidak dari database, sehingga data table harus diinput dahulu. Untuk menambahkan data pada arraylist cukup memanggil method add dengan new object. Untuk menampilkan data buku akan dilakukan berulang menggunakan for.

```

285 private void tombolRegisActionPerformed(java.awt.event.ActionEvent evt) {
286     DialogAntrianRegistrasi reg = new DialogAntrianRegistrasi(this, true, this.jTable2);
287     CenterLocation center = new CenterLocation(reg);
288     reg.setVisible(true);
289 }
290
291 private void tombolInputBukuActionPerformed(java.awt.event.ActionEvent evt) {
292     DialogInputBuku input = new DialogInputBuku(this, true, this.jTable4);
293     CenterLocation center = new CenterLocation(input);
294     input.setVisible(true);
295 }
296
297 private void ExitActionPerformed(java.awt.event.ActionEvent evt) {
298     System.exit(0);
299 }
300

```

Bagian action tombol regis adalah aksi yang akan dilakukan jika tombol Antrian Registrasi ditekan , maka akan memanggil class DialogAntrianRegistrasi, dan tombol input buku action akan memanggil DialogInputBuku, dan Exit Action akan keluar dari sistem.

```

FrameUtama.java x
Source Design History
302 public void runperKlik(int cek) {
303     String no_antrian = "";
304     String waktu_antrian = "";
305     String nama = "";
306     String buku = "";
307
308     int[] value = RegistrasiList.cariPrioritas();
309     DefaultTableModel m = (DefaultTableModel) this.jTable2.getModel();
310
311     for (int i = 0; i < m.getRowCount(); i++) {
312         if (value[1] == Integer.parseInt(
313             m.getValueAt(i, 0).toString())) {
314             no_antrian = m.getValueAt(i, 0).toString();
315             waktu_antrian = m.getValueAt(i, 2).toString();
316             nama = m.getValueAt(i, 3).toString();
317             buku = m.getValueAt(i, 4).toString();
318
319             DefaultTableModel m3 = (DefaultTableModel) this.jTable3.getModel();
320             m3.setValueAt(no_antrian, cek, 1);
321             m3.setValueAt(waktu_antrian, cek, 2);
322
323             m.removeRow(i);
324             RegistrasiList.deleteAntrianPrioritas(value[0]);
325             RegistrasiList.cetakAntrianRegistrasi();
326         }
327     }
}

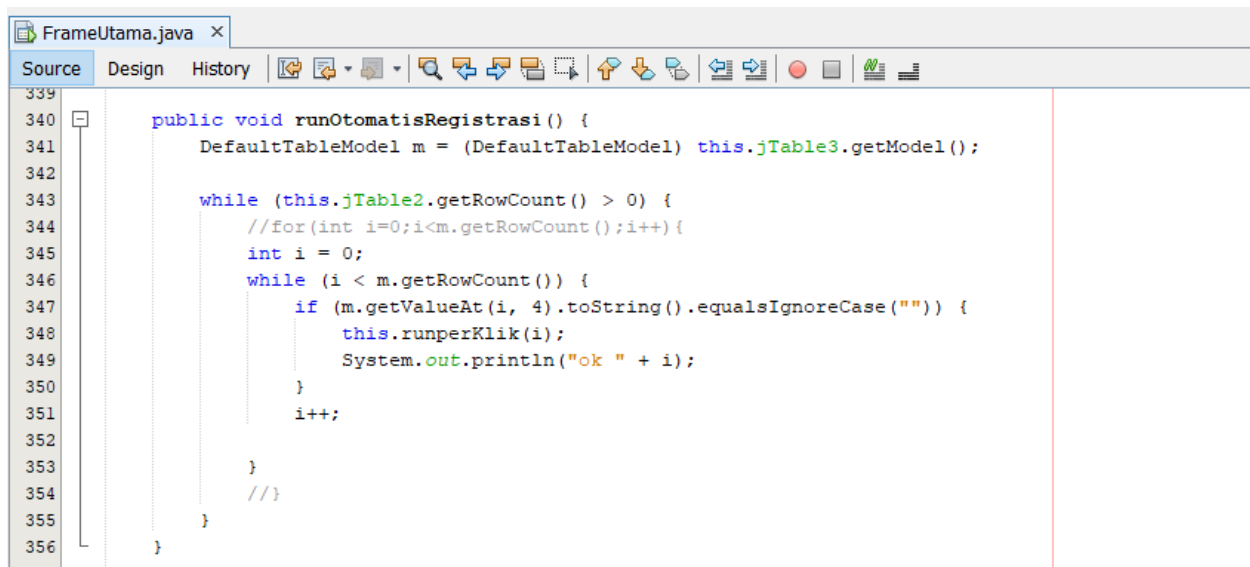
```

Pada method runperKlik dengan variable cek berisi variable value yang digunakan untuk mengecek prioritas dari antrian,dan akan mereturn nilai max dan nomor urut dari antrian. Kemudian Jtable2 diinisialisasikan sebagai m , jTable2 berisi Antrian Registrasi Bazaar. Isi dari table2 ini akan dimasukkan dalam table3 , sehingga untuk mengakses setiap data pada row jTable2 menggunakan for dengan batas m.getRowCount (banyaknya row pada jTable2), kemudian no_antrian diambil dari kolom ke 0 yaitu kolom no urut, waktu antrian diambil dari kolom 2 yaitu waktu, nama diambil dari kolom ke 3 dan buku diambil dari kolom ke 4. Kemudian m3 adalah jTable3 yang akan menampung data dari table2, naik no_antrian,

waktu_antrian nama dan buku sesuai kolom di jTable3, kemudian data di jTable 2 akan dihapus(remove row).

```
328 Stopwatchku sto = new Stopwatchku(new Waktuku(0, 0, 0));
329
330 WaktuPelayanan w = new WaktuPelayanan(sto,
331     jTable3,
332     Integer.parseInt(waktu_antrian), true);
333
334 DefaultTableModel m3 = (DefaultTableModel) this.jTable3.getModel();
335 m3.setValueAt(nama, cek, 5);
336 m3.setValueAt(buku, cek, 6);
337
338 }
```

Waktu layanan diambil dari class stopwatch dan waktu pelayanan berdasarkan waktu yang diinputkan.



```
FrameUtama.java x
Source Design History
339
340 public void runOtomatisRegistrasi() {
341     DefaultTableModel m = (DefaultTableModel) this.jTable3.getModel();
342
343     while (this.jTable2.getRowCount() > 0) {
344         //for(int i=0;i<m.getRowCount();i++){
345         int i = 0;
346         while (i < m.getRowCount()) {
347             if (m.getValueAt(i, 4).toString().equalsIgnoreCase("")) {
348                 this.runperKlik(i);
349                 System.out.println("ok " + i);
350             }
351             i++;
352         }
353         //}
354     }
355 }
356 }
```

Pada method ini akan menjalankan antrian setiap baris pada jTable2 sehingga akan dipanggil method runperklik dengan parameter cek sesuai dengan baris yang dijalankan.

```

419 public static void main(String args[]) {
420     java.awt.EventQueue.invokeLater(new Runnable() {
421
422         public void run() {
423             try {
424                 UIManager.setLookAndFeel(
425                     UIManager.getSystemLookAndFeelClassName());
426             } catch (Exception e) {
427             }
428             FrameUtama f = new FrameUtama();
429             f.setVisible(true);
430         }
431         public void run2() {
432             try {
433                 UIManager.setLookAndFeel(
434                     UIManager.getSystemLookAndFeelClassName());
435             } catch (Exception e) {
436             }
437             FrameUtama f = new FrameUtama();
438             f.setVisible(true);
439         }
440     });
441 }

```

2) GUI

The screenshot shows a Java Swing application window titled "Sistem Antrian Bazaar". The interface is divided into several sections:

- Sidebar (Left):** Contains three buttons: "Antrian Registrasi" (highlighted), "Input Data Buku", and "Keluar".
- Main Area (Top):** A large table titled "Antrian Bazaar" with columns: "No Urut", "Status", "Waktu", "Waktu Pelayanan", "Status", "Nama", and "Buku". It is currently empty.
- Main Area (Bottom Left):** A table titled "Kelompok 2" with columns "Nama" and "IDM". It contains the following data:

Nama	IDM
Abdul Aziz Fakhriul	1855301029
Farahdila Anisa	1855301043
Jean	1855301068
M. Hasbi	1855301084
- Main Area (Bottom Center):** A table titled "List Buku" with columns "ID Buku", "Judul Buku", and "Harga". It contains the following data:

ID Buku	Judul Buku	Harga
B-001	SDA	52000
B-002	RPL	60000
B-003	DW	53500
B-004	WEB	40000
B-005	JKD	35000
B-006	MTK	30000
- Main Area (Bottom Right):** A table titled "Antrian Registrasi Bazaar" with columns: "No Urut", "Status", "Waktu", "Nama", and "Buku". It is currently empty.

B. Package antrian_bazaar.dialog
1. Class DialogAntrianRegistrasi
1) Kodng

```
FrameUtama.java x DialogAntrianRegistrasi.java x
Source Design History
1 package antrian_bazaar.dialog;
2
3 import java.util.Vector;
4 import javax.swing.JTable;
5 import javax.swing.table.DefaultTableModel;
6 import javax.swing.text.Position;
7 import antrian_bazaar.FrameUtama;
8 import antrian_bazaar.tool.GenerateAntrian;
9 import linkedlist.RegistrasiList;
10
11 public class DialogAntrianRegistrasi extends javax.swing.JDialog {
12     JTable table;
13     /** Creates new form DialogTeller */
14     public DialogAntrianRegistrasi(java.awt.Frame parent, boolean modal, JTable tableAntrianTeller) {
15         super(parent, modal);
16         initComponents();
17         this.table=tableAntrianTeller;
18         int jum=FrameUtama.jumlahAntrianTeller+1;
19         this.jTextField1.setText(String.valueOf(jum));
20     }
21 }
```

Pada method ini menggunakan parameter table yang akan diisi dengan table2 yaitu table untuk antrian bazaar, kemudian untuk jTextField1 yaitu urutan mengambil data jumlahAntrianTeller yang semula bernilai 0 jadi ditambah 1 begitu seterusnya untuk urutan antrian. Urutan ini disimpan pada variable jum.

```
FrameUtama.java x DialogAntrianRegistrasi.java x
Source Design History
147 private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
148     DefaultTableModel m=(DefaultTableModel)this.table.getModel();
149     int no_urut=0;
150     if(m.getRowCount()==0){
151         no_urut=0;
152     }else{
153         no_urut=m.getRowCount()+1;
154     }
155     //penambahan pada tabel
156     Object[] o;
157     o=new Object[5];
158     FrameUtama.jumlahAntrianTeller++;
159     o[0]=FrameUtama.jumlahAntrianTeller;
160     o[1]=this.jComboBox1.getSelectedItem().toString();
161     o[2]=this.jComboBox2.getSelectedItem().toString();
162     o[3]=this.nama.getText();
163     o[4]=this.buku.getText();
164     m.addRow(o);
165
166     GenerateAntrian g=new GenerateAntrian(FrameUtama.jumlahAntrianTeller,
167         this.jComboBox1.getSelectedItem().toString(),
168         Integer.parseInt(this.jComboBox2.getSelectedItem().toString()),
169         this.nama.getText(), this.buku.getText());
170     RegistrasiList.insertAntrianAval(FrameUtama.jumlahAntrianTeller, g.getAntrian());
171     RegistrasiList.cetakAntrianRegistrasi();
172     int jum=FrameUtama.jumlahAntrianTeller+1;
173     this.jTextField1.setText(String.valueOf(jum));
174
175 }
```

Pada action ini akan dikerjakan jika kita menekan tombol ok, kemudian isi dari form registrasi ini akan dimasukkan ke jTable2 , sebelumnya disimpan pada object o karena pada jTable semua tipe data object. JComboBox1 adalah status baik member dan non member, jCombobox2 adalah waktu. Dan kemudian akan dimasukkan ke table2 dengan method addRow.

Selanjutnya memanggil class GenerateAntrian dengan parameter nomor, urutan, waktu, nama dan buku. Maka kemudian akan dijalankan antrianregistrasi.

2) GUI



Form Registrasi

Urutan

Nama

Status

Buku

Waktu detik

OK

2. Class DialogInputBuku

1) Koding

```
DialogInputBuku.java
Source Design History
1 package antrian_bazaar.dialog;
2
3 import java.util.Vector;
4 import javax.swing.JTable;
5 import javax.swing.table.DefaultTableModel;
6 import javax.swing.text.Position;
7 import antrian_bazaar.FrameUtama;
8 import antrian_bazaar.tool.GenerateAntrian;
9
10 import linkedlist.BukuList;
11 import linkedlist.RegistrasiList;
12
13 public class DialogInputBuku extends javax.swing.JDialog {
14     JTable table;
15     /** Creates new form DialogTeller */
16     public DialogInputBuku(java.awt.Frame parent, boolean modal, JTable tableAntrianTeller) {
17         super(parent, modal);
18         initComponents();
19         this.table = tableAntrianTeller;
20     }
21
22     @SuppressWarnings("unchecked")
23     // Generated Code
24
25     @SuppressWarnings("empty-statement")
26     private void simpanActionPerformed(java.awt.event.ActionEvent evt) {
27         DefaultTableModel m = (DefaultTableModel) this.table.getModel();
28
29         //penambahan pada tabel
30         Object[] o;
31         o = new Object[3];
32         o[0] = this.id.getText();
33         o[1] = this.judul.getText();
34         o[2] = Integer.parseInt(this.harga.getText());
35         m.addRow(o);
36     }
37
38     private void idActionPerformed(java.awt.event.ActionEvent evt) { ... 3 lines ... }
39     private void resetActionPerformed(java.awt.event.ActionEvent evt) {
40         id.setText("");
41         judul.setText("");
42         harga.setText("");
43     }
44 }
```

DialogInputBuku ini berkaitan dengan table4 yang akan digunakan untuk menampilkan list buku yang ada pada bazaar. Table4 diinisialisasikan sebagai table. Pada action simpan, nilai id buku, judul buku dan harga akan disimpan pada object o, kemudian untuk menambahkan setiap row data buku menggunakan method addRow, pada action reset hanya digunakan untuk mereset inputan pada form input buku

2) GUI

INPUT BUKU BAZAAR

ID

Judul Buku

Harga

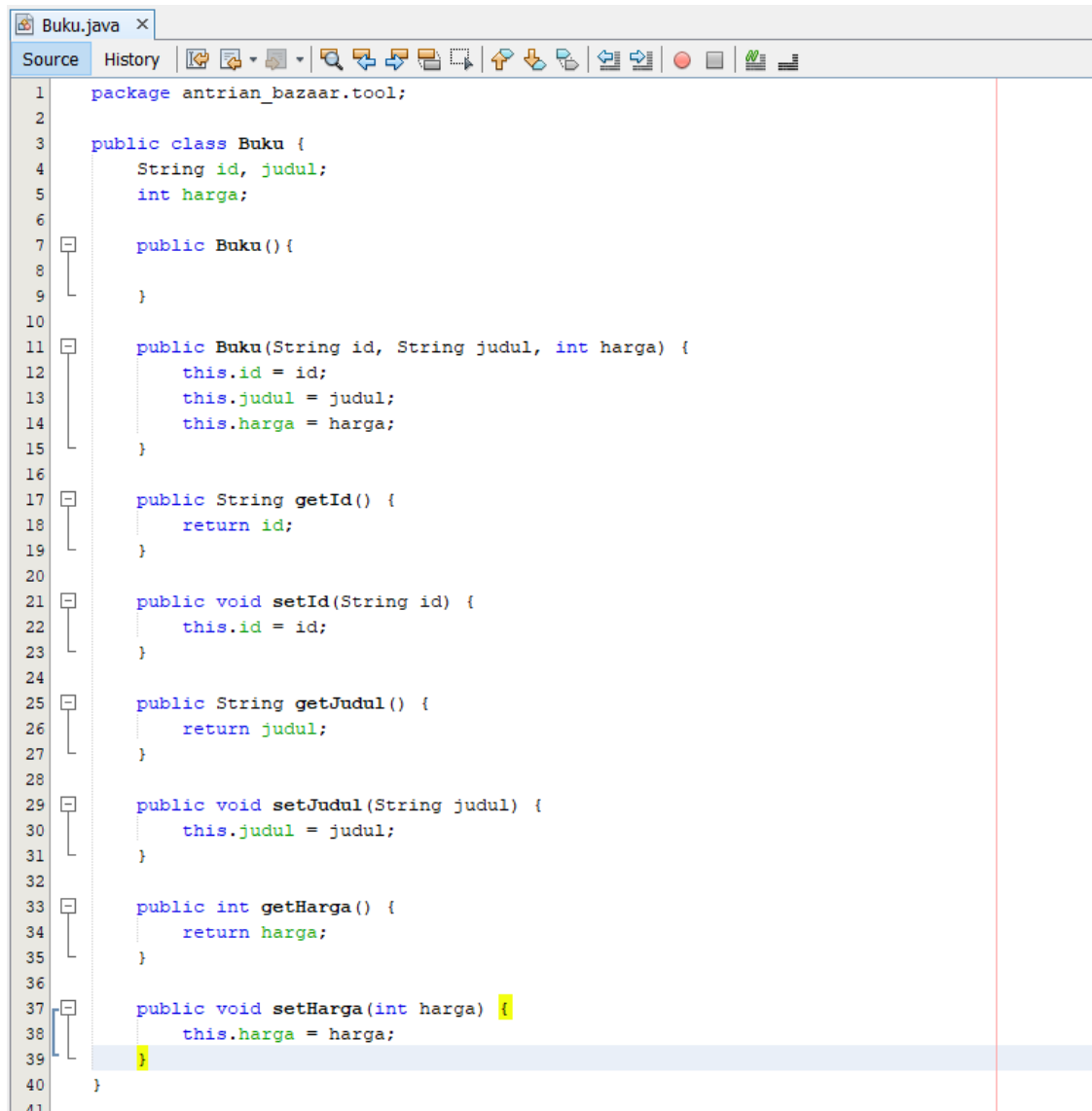
SIMPAN

RESET

C. Package antrian_bazaar.tool

1. Class Buku

1) Koding



```
1 package antrian_bazaar.tool;
2
3 public class Buku {
4     String id, judul;
5     int harga;
6
7     public Buku() {
8
9     }
10
11    public Buku(String id, String judul, int harga) {
12        this.id = id;
13        this.judul = judul;
14        this.harga = harga;
15    }
16
17    public String getId() {
18        return id;
19    }
20
21    public void setId(String id) {
22        this.id = id;
23    }
24
25    public String getJudul() {
26        return judul;
27    }
28
29    public void setJudul(String judul) {
30        this.judul = judul;
31    }
32
33    public int getHarga() {
34        return harga;
35    }
36
37    public void setHarga(int harga) {
38        this.harga = harga;
39    }
40 }
41
```

Berikut hanya digunakan untuk setter getter data buku.

2. Class RunAntrianRegistrasi

1) Koding

```
...avz RunAntrianRegistrasi.java x FrameUtama.java x RegistrasiList.java x DialogAntrianRegistrasi.j..
Source History
1 package antrian_bazaar.tool;
2
3 import java.util.logging.Level;
4 import java.util.logging.Logger;
5 import javax.swing.JTable;
6 import javax.swing.table.DefaultTableModel;
7 import javax.swing.JOptionPane;
8 import linkedlist.RegistrasiList;
9
10 public class RunAntrianRegistrasi implements Runnable {
11
12     private final Thread tt;
13     Stopwatchku stop;
14     JTable table2, table3, table1, table4;
15     public boolean kendaliRun = true;
16     int flag = 0;
17
18     //ini ubah jTable1
19     public RunAntrianRegistrasi(JTable table2, JTable table3, JTable table4) {
20         tt = new Thread(this);
21         tt.start();
22         this.table2 = table2;
23         this.table3 = table3;
24         this.table4 = table4;
25     }
26 }
```

```
Buku.java x RunAntrianRegistrasi.java x
Source History
28 @Override
29 public void run() {
30     while (kendaliRun) {
31         try {
32             WaktuPelayanan w[] = new WaktuPelayanan[table3.getRowCount()];
33             int j = 0;
34             while (j < table3.getRowCount()) {
35                 if (table3.getValueAt(j, 4).toString().equalsIgnoreCase("")) {
36                     String no_antrian = "";
37                     String waktu_antrian = "0";
38                     String status = "";
39                     String nama = "";
40                     String buku = "";
41                     String id = "";
42                     int harga = 0;
```

Berikut adalah bagian dari metode Searching array

```
43
44     int[] value = RegistrasiList.cariPrioritas();
45     DefaultTableModel m = (DefaultTableModel) table2.getModel();
46     int i = 0;
47
48     while (i < m.getRowCount()) {
49         if (value[i] == Integer.parseInt(
50             m.getValueAt(i, 0).toString())) {
51             no_antrian = m.getValueAt(i, 0).toString();
52             no_antrian = m.getValueAt(i, 0).toString();
53             waktu_antrian = m.getValueAt(i, 2).toString();
54             status = m.getValueAt(i, 1).toString();
55             nama = m.getValueAt(i, 3).toString();
56             buku = m.getValueAt(i, 4).toString();
57
58             //tambahan untuk searching array
59             for (int k = 0; k < table4.getRowCount(); k++) {
60                 if (buku.equalsIgnoreCase(table4.getValueAt(k, 1).toString())) {
61                     flag = 2;
62                     id = table4.getValueAt(k, 0).toString();
63                     harga = Integer.parseInt(table4.getValueAt(k, 2).toString());
64                     break;
65                 } else {
66                     flag = 0;
67                 }
68             }
69         }
70     }
```

```
Buku.java x RunAntrianRegistrasi.java x
Source History
69
70     DefaultTableModel m3 = (DefaultTableModel) table3.getModel();
71     m3.setValueAt("No Urut " + no_antrian, 3, 0);
72     m3.setValueAt("Status " + status, 3, 1);
73     m3.setValueAt(waktu_antrian, 3, 2);
74     m3.setValueAt(nama, 3, 5);
75     m3.setValueAt(buku, 3, 6);
76
77     m.removeRow(1);
78     RegistrasiList.deleteAntrianPrioritas(value[0]);
79     RegistrasiList.cetakAntrianRegistrasi();
80
81 }
82 i++;
83
84 //menampilkan Hasil search ARRAY
85 if (flag == 2) {
86     JOptionPane.showMessageDialog(null, "Data Buku ditemukan : \n- id Buku " + id + "\n- Judul "
87         + buku + "\n- harga Rp." + harga + "\n\nSilahkan Melakukan Pembayaran ke Teller !");
88
89 } else {
90     JOptionPane.showMessageDialog(null, "Data Buku dengan judul " + buku + " tidak ditemukan");
91 }
92 }
93
94 Stopwatch sto = new Stopwatchku(new Waktuku(0, 0, 0));
95 w[j] = new WaktuPelayanan(sto,
96     table3,
97     Integer.parseInt(waktu_antrian), true);
98
99 }
100
101 j++;
102 }
103 Thread.sleep(1000);
104
105 } catch (InterruptedException ex) {
106     ex.printStackTrace();
107     //kondisiRun=false;
108 }
109
110 }
111
112 }
```

Pada AntrianRegistrasi ini digunakan untuk menjalankan isi dari table ke-2 pindah ke table ke-3. Pada RunAntianRegistrasi terdapat Pencarian Prioritas yang berguna untuk memprioritaskan member daripada non-member. Pada RunAntrianRegistrasi juga mengambil data table ke-4 yang digunakan untuk searching array dengan data buku yang ada didalam array tersebut. Jika data tidak ditemukan maka akan muncul kotak dialog data tidak ditemukan dan jika data ditemukan maka akan muncul kotak dialog dengan id buku dan harga buku tersebut. Pada class ini juga mengambil Stopwatch yang berguna untuk berjalannya waktu pelayanannya.

3. Class Antrian

1) Koding

```
1 package antrian_bazaar.tool;
2
3 public class Antrian {
4
5     int no_urut;//no urut antrian teller
6     int status;//status antrian teller(biasa/vip)
7     int waktu;//lama pelayanan dalam menit
8     int point = 0;//untuk prioritas pengurutan
9     String nama;
10    String buku;
11
12    public Antrian(int no, int status, int waktu, String nama, String buku) {
13        this.no_urut = no;
14        this.status = status;
15        this.nama = nama;
16        this.buku = buku;
17
18        this.waktu = waktu;
19        if (status == 2) {
20            point = point + 4 + no;//biasa
21        } else {
22            point = point + 8 + no;//VIP
23        }
24        if (waktu >= 10) {
25            point = point + 13;
26        }
27    }
28
29    public Antrian() {
30    }
31
32
33    public int getPoint() {
34        return point;
35    }
36
37    public int getNo_urut() {
38        return no_urut;
39    }
40
41    public void setNo_urut(int no_urut) {
42        this.no_urut = no_urut;
43    }
44
45    public int getStatus() {
46        return status;
47    }
48 }
```

```

49 public void setStatus(int status) {
50
51     this.status = status;
52 }
53
54 public int getWaktu() {
55     return waktu;
56 }
57
58 public void setWaktu(int waktu) {
59     if (waktu > 60) {
60         waktu = 60;
61     }
62     if (waktu < 5) {
63         waktu = 5;
64     }
65     if (status == 1) { //VIP
66         waktu = 5;
67         System.out.println("dikerjakan waktu");
68     }
69     this.waktu = waktu;
70 }
71
72 public String getNama() {
73     return nama;
74 }
75
76 public void setNama(String nama) {
77     this.nama = nama;
78 }
79
80 public String getBuku() {
81     return buku;
82 }
83
84 public void setBuku(String buku) {
85     this.buku = buku;
86 }
87
88 }

```

2) Penjelasan

Pada Class Antrian digunakan untuk menentukan point yang digunakan untuk prioritas yang lebih diutamakan. Untuk class Antrian hanya digunakan untuk melakukan set dan get untuk setiap parameter pada Antrian. Untuk Set yang berbeda hanya waktu karena pada waktu terdapat kondisi jika dipilih lebih dari 60 maka waktu akan terset 60 dan jika waktu kurang dari 5 maka waktu akan diset 5. Pada status juga terdapat kondisi, jika nilai status sama dengan 1 maka di set waktunya 5.

4. Class GenerateAntrian

1) Koding

```
1 package antrian_bank.tool;
2
3 public class GenerateAntrian {
4     Antrian antrian;
5     public GenerateAntrian(int no_urut, String status, int waktunya, String nama, String buku) {
6         int sta=1;
7         if(status.equalsIgnoreCase("Non-Member")){
8             sta=1;
9         }else{
10             sta=2;
11         }
12         antrian=new Antrian(no_urut, sta, waktunya, nama, buku);
13     }
14
15     public Antrian getAntrian() {
16         return antrian;
17     }
18
19     public void setAntrian(Antrian antrian) {
20         this.antrian = antrian;
21     }
22
23 }
```

2) Penjelasan

Untuk class GenerationAntrian ini digunakan untuk mengerakkan antrian yang ada. Generation memanggil Antrian dengan 5 parameter yang didalamnya terdapat no urut, perbandingan status, waktu, nama dan buku. Setelah melakukan perbandingan status, dikirim dengan sta bernilai angka, jika non-member bernilai angka 1 dan jika member bernilai angka 2 lalu dikirimkan ke antrian.

5. Class RunAntrianInputBuku

1) Koding

```
1 package antrian_bazaar.tool;
2
3 import java.util.logging.Level;
4 import java.util.logging.Logger;
5 import javax.swing.JTable;
6 import javax.swing.table.DefaultTableModel;
7 import linkedlist.BukuList;
8 import linkedlist.RegistrasiList;
```

```

10 public class RunInputBuku implements Runnable {
11
12     private final Thread tt;
13     JTable table3, table1;
14     public boolean kendaliRun = true;
15
16     //jtable3,jtable1 yang diujung
17     public RunInputBuku(JTable table3, JTable table1) {
18         tt = new Thread(this);
19         tt.start();
20         this.table3 = table3;
21         this.table1 = table1;
22     }
23
24     @Override
25     public void run() {
26         while (kendaliRun) {
27
28             try {
29                 WaktuPelayanan w[] = new WaktuPelayanan[table1.getRowCount()];
30                 int j = 0;
31                 while (j < table1.getRowCount()) {
32                     if (table1.getValueAt(j, 4).toString().equalsIgnoreCase("")) {
33                         String no_antrian = "";
34                         String waktu_antrian = "0";
35                         String status = "";
36                         String id_judul = "";
37                         int harga;
38
39                         int[] value = BukuList.cariPrioritas();
40                         DefaultTableModel m = (DefaultTableModel) table3.getModel();
41                         for (int i = 0; i < m.getRowCount(); i++) {
42
43                             //String cs=m3.getValueAt(i, 0).toString();
44                             if (value[i] == Integer.parseInt(
45                                 m.getValueAt(i, 0).toString())) {
46                                 no_antrian = m.getValueAt(i, 1).toString();
47                                 waktu_antrian = m.getValueAt(i, 2).toString();
48                                 //
49                                 status=m.getValueAt(i, 1).toString();
50
51                                 DefaultTableModel m3 = (DefaultTableModel) table1.getModel();
52                                 // Thread t=new Thread();
53                                 //t.start();
54                                 String html = "<html><body><font size=3 color=ff0000><b>No antrian " + no_antrian
55                                     + " ditunggu </b></font></body></html>";
56                                 m3.setValueAt(html,
57                                     j, 1);
58
59                                 m3.setValueAt("No " + no_antrian + " status " + status, j, 1);
60                                 m3.setValueAt(waktu_antrian, j, 2);
61                                 m3.setValueAt("", j, 5);
62                                 m3.setValueAt("", j, 6);
63                                 m.removeRow(i);
64                                 BukuList.deleteAntrianPrioritas(value[0]);
65                                 BukuList.cetakAntrianRegistrasi();
66                             }
67                         }
68                     }
69                 }
70             }
71         }
72     }
73 }

```



```

67         Stopwatchku sto = new Stopwatchku(new Waktuku(0, 0, 0));
68         w[j] = new WaktuPelayanan(sto,
69             table1,
70             Integer.parseInt(waktu_antrian), true);
71     }
72     j++;
73 }
74 Thread.sleep(1000);
75 } catch (InterruptedException ex) {
76     ex.printStackTrace();
77     //kendaliRun=false;
78 }
79 }
80 }
81 }
82 }

```

2) Penjelasan

Pada AntrianInputBuku ini digunakan untuk menjalankan isi dari table ke-4. Pada RunAntianInputBuku hanya berguna untuk melakukan pengisian untuk isi table ke-4 .

6. Class Stopwatchku

1) Koding

```

1 package antrian_bank.tool;
2
3 import java.awt.event.ActionEvent;
4 import java.awt.event.ActionListener;
5 import javax.swing.Timer;
6
7 public class Stopwatchku {
8     Timer timer;
9     int jam=0;
10    int menit=0;
11    int detik=0;
12    Waktuku waktuku;
13    public Stopwatchku(Waktuku waktuku){
14        this.waktuku=waktuku;
15        this.jam=waktuku.getJam();
16        this.menit=waktuku.getMenit();
17        this.detik=waktuku.getDetik();
18        ActionListener aksi=new ActionListener() {
19            @Override
20            public void actionPerformed(ActionEvent e) {
21                //waktuku.getDetik()+=1;
22                detik+=1;
23                if(detik>59){
24                    detik=0;
25                    menit+=1;
26                    if(menit>59){
27                        jam+=1;
28                        menit=0;
29                    }
30                }
31            }
32            //System.out.println("timer : "+jam+" "+menit+" "+detik);
33        };
34    };
35    this.timer=new Timer(1000, aksi);
36    this.timer.start();
37 }
38

```

Pada class stopwatchku digunakan untuk timer antrian sehingga menggunakan fitur timer untuk jam menit dan detik , namun dalam implementasinya hanya menggunakan detik. Kemudian menggunakan parameter waktuku untuk getter jam menit atau detik nya, kemudian untuk mendeteksi apakah waktu itu adalah menit maka ada kondisi detik>59 maka menit + 1 kemudian timer akan dimulai dengan method start. Timer akan mulai dari 0 detik yaitu seperti pada method main (new Waktuku(0 → jam, 0 → menit, 0 → detik)).

```
39  public Waktuku getWaktuku() {
40      waktuku.setJam(this.jam);
41      waktuku.setMenit(this.menit);
42      waktuku.setDetik(this.detik);
43      return waktuku;
44  }
45
46  public int getDetik() {
47      return detik;
48  }
49
50  public void setDetik(int detik) {
51      this.detik = detik;
52  }
53
54  public int getJam() {
55      return jam;
56  }
57
58  public void setJam(int jam) {
59      this.jam = jam;
60  }
61
62  public int getMenit() {
63      return menit;
64  }
65
66  public void setMenit(int menit) {
67      this.menit = menit;
68  }
69
70  public Timer getTimer() {
71      return timer;
72  }
73
74  public void setTimer(Timer timer) {
75      this.timer = timer;
76  }
77
78  public static void main(String[] args) {
79      new Stopwatchku(new Waktuku(0, 0, 0));
80  }
81
82  }
```

7. Class Waktu

1) Koding

```
1  + ...4 lines
5  package antrian_bank.tool;
6  import javax.swing.JLabel;
7  + /**...4 lines */
11 public class Waktu implements Runnable{
12     StopWatchku stop;
13     Thread tt;
14     JLabel labelWaktu;
15     public static boolean kendali=true;
16     public Waktu(StopWatchku stopWatch, JLabel labelWaktu){
17         this.stop=stopWatch;
18         tt=new Thread(this);
19         tt.start();
20         this.labelWaktu=labelWaktu;
21     }
22
23     public void destroy(){
24         tt.stop();
25         stop.getTimer().stop();
26     }
27     public void paused(){
28         kendali=false;
29     }
30     public void resume(){
31
32         kendali=true;
33     }
34     @Override
35     public void run() {
36         while(kendali){
37             try {
38                 labelWaktu.setText(""+stop.getJam()+":"+stop.getMenit()+":"+stop.getDetik());
39                 Thread.sleep(1000);
40             } catch (InterruptedException ex) {
41
42             }
43         }
44     }
45 }
```

8. Class WaktuPelayanan

1) Koding

```
1 package antrian_bank.tool;
2
3 import javax.swing.JTable;
4 import javax.swing.table.DefaultTableModel;
5 public class WaktuPelayanan implements Runnable{
6     Stopwatch stop;
7     Thread tt;
8     JTable table;
9     DefaultTableModel m;
10    int noTeller;
11    int waktuHabis;
12    public boolean kendali=true;
13    public WaktuPelayanan(StopWatchku stopWatch, JTable table, int noTeller, int waktuHabis, boolean cek){
14        this.stop=stopWatch;
15        tt=new Thread(this);
16        tt.start();
17        this.table=table;
18        this.m=(DefaultTableModel) table.getModel();
19        this.noTeller=noTeller;
20        // System.out.println("no teller "+this.noTeller);
21        this.waktuHabis=waktuHabis;
22        kendali=cek;
23    }
24
25    public void destroy(){
26        tt.stop();
27        stop.getTimer().stop();
28    }
29    public void paused(){
30        kendali=false;
31    }
32    public void resume(){
33
34        kendali=true;
35    }
36
37    @Override
38    public void run() {
39        while(kendali){
40            try {
41                m.setValueAt(""+stop.getJam()+":"+stop.getMenit()+":"+stop.getDetik(), noTeller-1, 3);
42                m.setValueAt("Sibuk", noTeller-1, 4);
43                if(this.waktuHabis==stop.getDetik()){
44                    //System.out.println("WAKTU HABIS "+this.waktuHabis+" STOP "+ stop.getDetik());
45                    m.setValueAt("", noTeller-1, 1);
46                    m.setValueAt("", noTeller-1, 2);
47                    m.setValueAt("", noTeller-1, 3);
48                    m.setValueAt("", noTeller-1, 4);
49                    m.setValueAt("", noTeller-1, 5);
50                    m.setValueAt("", noTeller-1, 6);
51                    this.paused();
52                }
53
54                Thread.sleep(1000);
55            } catch (InterruptedException ex) {
56                this.kendali=false;
57            }
58        }
59    }
60 }
```

9. Class Waktuku

1) Koding

```
1  [+ ...4 lines
5  package antrian_bank.tool;
6  import java.io.Serializable;
7  [+ /**...4 lines */
11 public class Waktuku implements Cloneable, Serializable{
12     int jam=0;
13     int menit=0;
14     int detik=0;
15     public Waktuku(){
16     }
17     }
18     public Waktuku(int jam, int menit, int detik){
19         this.jam=jam;
20         this.menit=menit;
21         this.detik=detik;
22     }
23     public String getTime(){
24         String sjam=String.valueOf(jam);
25         String smenit=String.valueOf(menit);
26         String sdetik=String.valueOf(detik);
27
28         if(menit<10){
29             smenit="0"+smenit;
30         }
31         if(jam<10){
32             sjam="0"+sjam;
33         }
34         if(detik<10){
35             sdetik="0"+sdetik;
36         }
37         return sjam+":"+smenit+":"+sdetik;
38     }
39     public int getDetik() {
40         return detik;
41     }
42
43     public void setDetik(int detik) {
44         this.detik = detik;
45     }
46
47     public int getJam() {
48         return jam;
49     }
50
51     public void setJam(int jam) {
52         this.jam = jam;
53     }
54
55     public int getMenit() {
56         return menit;
57     }
58
59     public void setMenit(int menit) {
60         this.menit = menit;
61     }
62     @Override
63     public Object clone() throws CloneNotSupportedException{
64         return super.clone();
65     }
66 }
```

Pada saat pembuatan terdapat beberapa error


a. Untuk notifikasi data ditemukan atau tidak sebelumnya seperti ini

[illegible]

Maka ketika dijalankan

The screenshot displays a Java Swing application for a queue system. The main window is titled "Sistem Antrian Queue" and features three primary panels on the left side, each containing a table. The top panel, "Antrean Registrasi", has a table with columns: No Urut, Status, Waktu, Waktu Pelayanan, Status, Name, and Buku. The middle panel, "Input Data Buku", has a table with columns: Nama and IDK. The bottom panel, "Antrean", has a table with columns: ID Buku, Subst Buku, and Harga. A modal dialog titled "Antrean Registrasi" is open, showing a "Form Registrasi" with fields for Username, Password, Status (New Member), Buku, and Waktu, and a "Daftar" button.


Message

 Data Buku dengan judul JKD tidak ditemukan

OK

Notifikasi ini akan berulang sampai data mengarah pada posisi data JKD. Setelah itu baru tampilan nya menjadi

Message

 Data Buku ditemukan !

- Id Buku B-005
- Judul JKD
- harga Rp.35000

Silahkan Melakukan Pembayaran ke Teller 1

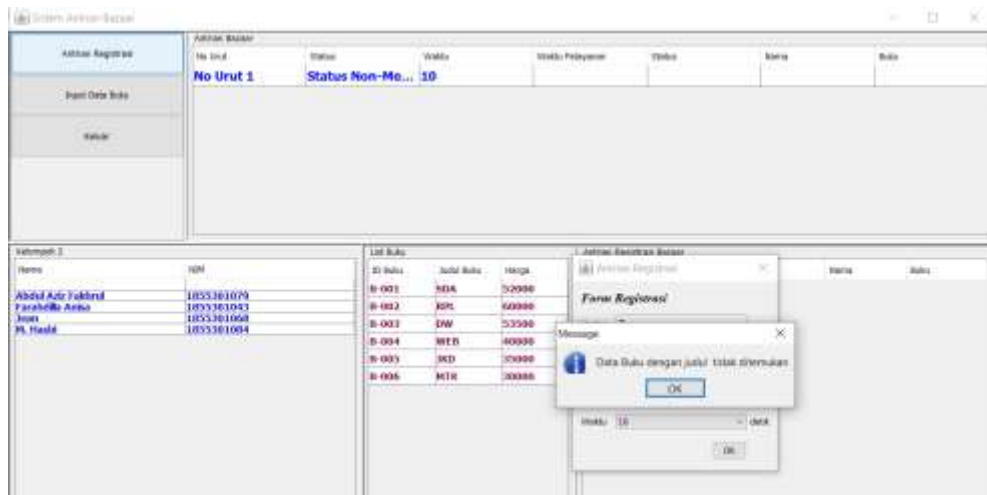
OK

Maka solusi nya adalah

```
//tambahan untuk searching array
for (int k = 0; k < table4.getRowCount(); k++) {
    if (buku.equalsIgnoreCase(table4.getValueAt(k, 1).toString())) {
        flag = 2;
        id = table4.getValueAt(k, 0).toString();
        harga = Integer.parseInt(table4.getValueAt(k, 2).toString());
        break;
    } else {
        flag = 0;
    }
}

//menampilkan hasil search array
if (flag == 2) {
    JOptionPane.showMessageDialog(null, "Data Buku ditemukan ! \n- Id Buku " + id + "\n- Judul " +
        buku + "\n- harga Rp." + harga + "\nSilahkan Melakukan Pembayaran ke Teller !");
} else {
    JOptionPane.showMessageDialog(null, "Data Buku dengan judul " + buku + " tidak ditemukan");
}
}
```

Selain itu juga terdapat permasalahan saat data buku dan nama tidak terbaca sedangkan data yang dimasukkan benar



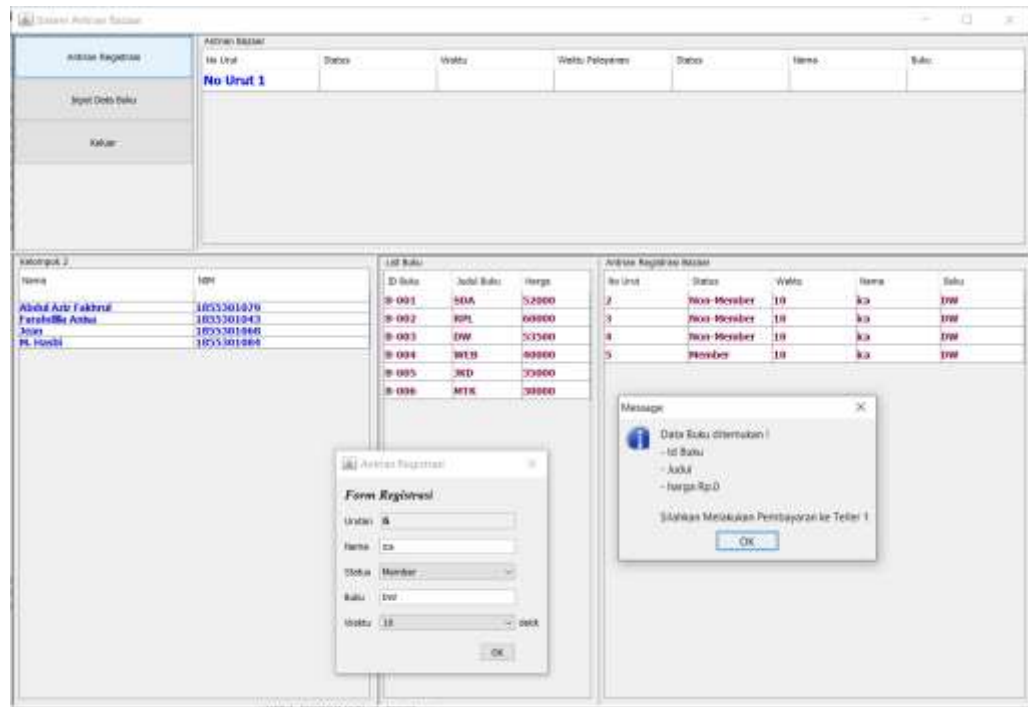
Ini adalah solusi nya

```

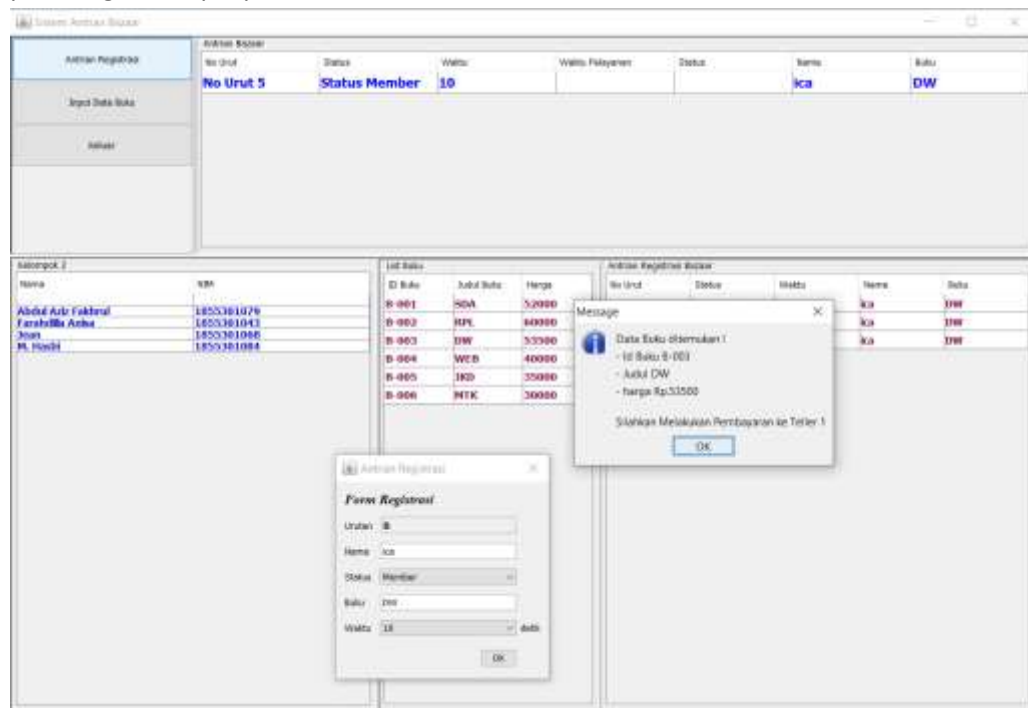
47 while (i < m.getRowCount()) {
48     if (value[i] == Integer.parseInt(
49         m.getValueAt(i, 0).toString())) {
50         no_antrian = m.getValueAt(i, 0).toString();
51         no_antrian = m.getValueAt(i, 0).toString();
52         waktu_antrian = m.getValueAt(i, 2).toString();
53         status = m.getValueAt(i, 1).toString();
54         nama = m.getValueAt(i, 3).toString();
55         buku = m.getValueAt(i, 4).toString();
56
57         //tambahan untuk searching array
58         for (int k = 0; k < table4.getRowCount(); k++) {
59             if (buku.equalsIgnoreCase(table4.getValueAt(k, 1).toString())) {
60                 flag = 2;
61                 id = table4.getValueAt(k, 0).toString();
62                 harga = Integer.parseInt(table4.getValueAt(k, 2).toString());
63                 break;
64             } else {
65                 flag = 0;
66             }
67         }
68
69         DefaultTableModel m3 = (DefaultTableModel) table3.getModel();
70         m3.setValueAt("No Urut " + no_antrian, 3, 0);
71         m3.setValueAt("Status " + status, 3, 1);
72         m3.setValueAt(waktu_antrian, 3, 2);
73         m3.setValueAt(nama, 3, 5);
74         m3.setValueAt(buku, 3, 6);
75
76         m.removeRow(i);
77         RegistrasiList.deleteAntrianPrioritas(value[0]);
78         RegistrasiList.cetakAntrianRegistrasi();
79

```

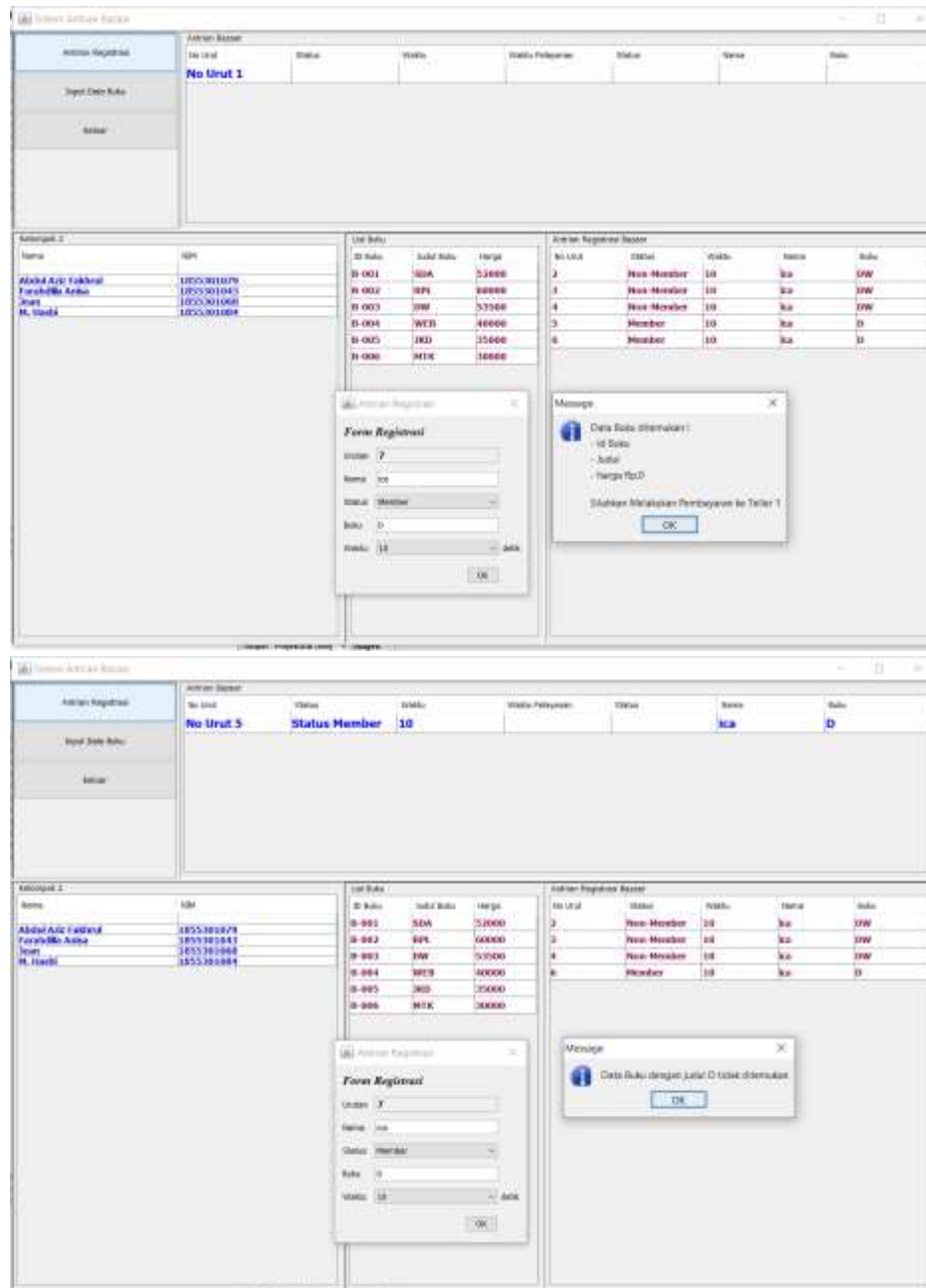
Selain itu terdapat beberapa permasalahan pada bagian antrian jika terdapat lebih dari 3 antrian yang dimasukkan dengan antrian yang terdapat member



Buku yang dimasukkan tidak dapat menampilkan info data buku tersebut hingga perulangan sampai pada baris data DW



Berikut adalah permasalahan yang terjadi jika antrian terdapat jenis non dan member, dan member memasukkan data yang salah



4. Pembagian Tugas Kelompok

a. Abdul Aziz Fakhrol Soleh

- Mengelola bagian RegistrasiList ,Buku List, Generate Antrian, Antrian dan membuat laporan dan Analisa bagian tersebut.

b. Farahdilla Anisa Putri

- Mengelola Bagian FrameUtama, DialogAntrianRegistrasi, DialogInputBuku, Buku dan RunAntrianRegistrasi serta membuat laporan dan Analisa mengenai bagian tersebut.

c. Jean

- Mengelola Bagian Stopwatch dan membantu dalam pembuatan laporan seperti screenshot coding

d. M. Hasbi Assidiq

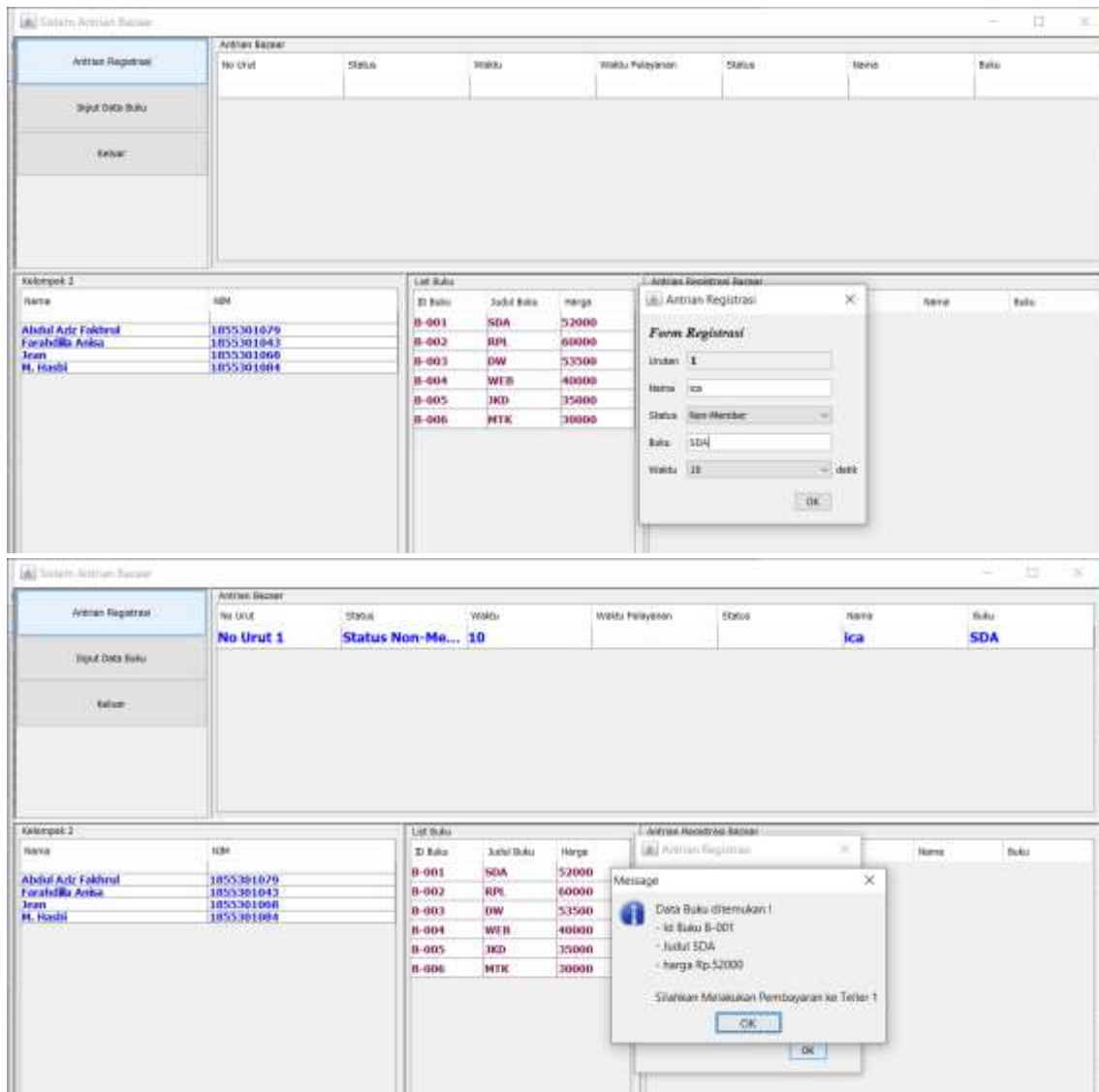
- Membantu dalam pembuatan laporan seperti screenshot bagian pengujian aplikasi.

5. Link Aplikasi

<https://github.com/FarahdillaAnisa/Tugas-Projek-SDA.git>

6. Pengujian

1. Pengujian Pertama untuk antrian dengan data buku yang benar.



Sistem Antrian Bazaar

Antrian Bazaar							
No Urut	Status	Waktu	Waktu Pelayanan	Status	Nama	Buku	
No Urut 1	Status Non-Me...	10	0:0:2	Sibuk	Ica	SDA	

Antrean Registrasi

Input Data Buku

Keluar

2. Jika Data buku yang dicari tidak ada.

Sistem Antrian Bazaar

Antrian Bazaar							
No Urut	Status	Waktu	Waktu Pelayanan	Status	Nama	Buku	
No Urut 3	Status Non-Me...	10	0:0:7	Sibuk	Ica	SDA	

Antrean Registrasi

Input Data Buku

Keluar

Kelompok 2

Nama	NIM
Abdul Aziz Fakhru	1855301079
Farahdila Anisa	1855301043
Jean	1855301060
M. Hasbi	1855301084

List Buku

ID Buku	Judul Buku	Harga
B-001	SDA	52000
B-002	RPL	60000
B-003	DW	53500
B-004	WEB	40000
B-005	JKD	35000
B-006	MTK	30000

Antrean Registrasi Bazaar

Form Registrasi

Urutan: 6

Nama: Ica

Status: Non-Member

Buku: SD

Waktu: 10

OK

Sistem Antrian Bazaar

Antrian Bazaar							
No Urut	Status	Waktu	Waktu Pelayanan	Status	Nama	Buku	
No Urut 5	Status Non-Me...	10			Ica	SD	

Antrean Registrasi

Input Data Buku

Keluar

Kelompok 2

Nama	NIM
Abdul Aziz Fakhru	1855301079
Farahdila Anisa	1855301043
Jean	1855301060
M. Hasbi	1855301084

List Buku

ID Buku	Judul Buku	Harga
B-001	SDA	52000
B-002	RPL	60000
B-003	DW	53500
B-004	WEB	40000
B-005	JKD	35000
B-006	MTK	30000

Antrean Registrasi Bazaar

No Urut	Status	Waktu	Nama	Buku

Message

Data Buku dengan judul SD tidak ditemukan

OK

3. Jika antrian terdiri dari non member dan member, maka akan didahulukan yang member.

The screenshot shows the 'Sistem Antrian Buku' application. The main window has a sidebar with 'Antrian Registrasi', 'Input Data Buku', and 'Keluar'. The main area displays a queue table with columns: No Urut, Status, Waktu, Waktu Pelayanan, Status, Nama, and Buku. The queue contains one entry: No Urut 11, Status Non-Me..., Waktu 10, Waktu Pelayanan 0:0:5, Status Sibuk, Nama Ica, and Buku RPL. Below the queue, there are two tables: 'Kelompok 2' (listing names and NPM) and 'List Buku' (listing book IDs, titles, and prices). A 'Form Registrasi' dialog box is open, showing fields for Urutan (14), Nama (Ica), Status (Member), Buku (RPL), and Waktu (10 detik). The 'OK' button is visible.

No Urut	Status	Waktu	Waktu Pelayanan	Status	Nama	Buku
No Urut 11	Status Non-Me...	10	0:0:5	Sibuk	Ica	RPL

ID Buku	Judul Buku	Harga
B-001	SDA	52000
B-002	RPL	60000
B-003	DW	53500
B-004	WEB	40000
B-005	JKD	35000
B-006	MTK	30000

Nama	NPM
Abdul Aziz Fakhriul	1855301079
Farahdila Anisa	1855301043
Jean	1855301068
M. Hasbi	1855301084

The screenshot shows the 'Sistem Antrian Buku' application. The main window has a sidebar with 'Antrian Registrasi', 'Input Data Buku', and 'Keluar'. The main area displays a queue table with columns: No Urut, Status, Waktu, Waktu Pelayanan, Status, Nama, and Buku. The queue contains one entry: No Urut 13, Status Member, Waktu 10, Waktu Pelayanan, Status, Nama Ica, and Buku RPL. Below the queue, there are two tables: 'Kelompok 2' (listing names and NPM) and 'List Buku' (listing book IDs, titles, and prices). A 'Message' dialog box is open, displaying the text: 'Data Buku ditemukan!', '- Id Buku B-002', '- Judul RPL', '- Harga Rp.60000', and 'Silahkan Melakukan Pembayaran ke Teller !'. The 'OK' button is visible.

No Urut	Status	Waktu	Waktu Pelayanan	Status	Nama	Buku
No Urut 13	Status Member	10			Ica	RPL

ID Buku	Judul Buku	Harga
B-001	SDA	52000
B-002	RPL	60000
B-003	DW	53500
B-004	WEB	40000
B-005	JKD	35000
B-006	MTK	30000

Nama	NPM
Abdul Aziz Fakhriul	1855301079
Farahdila Anisa	1855301043
Jean	1855301068
M. Hasbi	1855301084

4. Input Buku

Sistem Antrian Bazaar

Antrian Registrasi

Input Data Buku

Keluar

Antrian Bazaar

No Urut	Status	Waktu	Waktu Pelayanan	Status	Nama	Buku
No Urut 12						

kelompok 2

Nama	IDM
Abdul Aziz Fakhriul	1855301079
Farahdilla Anisa	1855301043
Jean	1855301068
M. Hasbi	1855301084

List Buku

ID Buku	Judul Buku	Harga
B-001	SDA	52000
B-002	RPL	60000
B-003	DW	53500
B-004	WEB	40000
B-005	JKD	35000
B-006	MTK	38000

Antrian Registrasi Bazaar

No Urut	Status	Waktu	Nama	Buku
---------	--------	-------	------	------

INPUT DATA BUKU

ID

B-007

Judul buku

OK-OCE

Harga

50000

SIMPAN

RESET

Sistem Antrian Bazaar

Antrian Registrasi

Input Data Buku

Keluar

Antrian Bazaar

No Urut	Status	Waktu	Waktu Pelayanan	Status	Nama	Buku
No Urut 12						

kelompok 2

Nama	IDM
Abdul Aziz Fakhriul	1855301079
Farahdilla Anisa	1855301043
Jean	1855301068
M. Hasbi	1855301084

List Buku

ID Buku	Judul Buku	Harga
B-001	SDA	52000
B-002	RPL	60000
B-003	DW	53500
B-004	WEB	40000
B-005	JKD	35000
B-006	MTK	38000
B-007	OK-OCE	50000

Antrian Registrasi Bazaar

No Urut	Status	Waktu	Nama	Buku
---------	--------	-------	------	------

5. Pengecekan buku dengan data buku yang baru saja di input.

Sistem Antrian Bazaar

Antrean Registrasi

Input Data Buku

Keluar

Antrean Bazar

No Urut	Status	Waktu	Waktu Pelayanan	Status	Nama	Buku
No Urut 12						

Kelompok 2

Nama	NIM
Abdul Aziz Fakhru	1855301079
Farahdila Anisa	1855301043
Jean	1855301068
M. Hasbi	1855301084

List Buku

ID Buku	Judul Buku	Harga
B-001	SDA	52000
B-002	RPL	60000
B-003	DW	53500
B-004	WEB	40000
B-005	JKD	35000
B-006	MTK	30000
B-007	OK-OCE	50000

Antrean Registrasi Bazaar

Form Registrasi

Urutan: 14

Nama: Ica

Status: Non-Member

Buku: OK-OCE

Waktu: 10 det

OK

Sistem Antrian Bazaar

Antrean Bazar

No Urut	Status	Waktu	Waktu Pelayanan	Status	Nama	Buku
No Urut 14	Status Non-Me...	10			Ica	OK-OCE

Kelompok 2

Nama	NIM
Abdul Aziz Fakhru	1855301079
Farahdila Anisa	1855301043
Jean	1855301068
M. Hasbi	1855301084

List Buku

ID Buku	Judul Buku	Harga
B-001	SDA	52000
B-002	RPL	60000
B-003	DW	53500
B-004	WEB	40000
B-005	JKD	35000
B-006	MTK	30000
B-007	OK-OCE	50000

Antrean Registrasi Bazaar

Message

Data Buku ditemukan!

- Id Buku B-007
- Judul OK-OCE
- harga Rp.50000

Silahkan Melakukan Pembayaran ke Teller 1

OK

Sistem Antrian Bazaar

Antrean Bazar

No Urut	Status	Waktu	Waktu Pelayanan	Status	Nama	Buku
No Urut 14	Status Non-Me...	10	0:0:3	Sibuk	Ica	OK-OCE