

# SIMULASI SISTEM UNTUK UJI KESERAGAMAN BOBOT TABLET

Pulut Suryati<sup>1)</sup>, Syamsu Windarti<sup>2)</sup>

<sup>1)</sup> STMIK AKAKOM Yogyakarta, Jl. Raya Janti 143, Karangjambe Yogyakarta

E-mail : [jut.surya@akakom.ac.id](mailto:jut.surya@akakom.ac.id)

<sup>2)</sup> STMIK AKAKOM Yogyakarta, Jl. Raya Janti 143, Karangjambe Yogyakarta

E-mail : [wiwien@akakom.ac.id](mailto:wiwien@akakom.ac.id)

## STRAK

*Pengujian keseragaman bobot tablet merupakan satu tahapan yang harus selalu dikejakan dalam proses produksi tablet monitoring kualitas obat. Dalam proses produksi, pengujian dilakukan secara periodik. Tahapan ini memerlukan kecermatan dan ketelitian dalam menimbang dan menghitung. Untuk dalam penelitian ini akan dicoba mengembangkan sistem untuk mensimulasikan penghitungan dalam uji keseragaman bobot tablet, sehingga bisa memberikan keputusan bahwa tablet memenuhi syarat atau tidak.*

*Sistem ini dibangun melalui tahapan pemuatan algoritma, perancangan interface, penulisan kode program dan pengujian sistem. Sistem dibangun menggunakan C++ Builder. Sistem ini digunakan untuk menimbang berat tablet pengujian. Sistem memberikan hasil berupa keputusan bahwa tablet tersebut memenuhi syarat atau tidak.*

*Kunci : pendukung keputusan, uji keseragaman*

## PENDAHULUAN

Obat diperlukan oleh masyarakat untuk melihara kesehatan, mengobati gangguan kesehatan dan memulihkan kesehatan. Sehingga mencapai tujuan tersebut perlu dilakukan langkah agar obat yang dihasilkan aman, bermanfaat dan bermutu. Kualitas dan mutu obat tergantung pada bahan bangunan, prosedur dan pelaksanaan proses produksi, pengemas termasuk bahannya serta orang-orang yang terlibat dalam proses pembuatan obat. Maka perlu pengawasan yang menyeluruh dan jujuan untuk menyediakan obat yang aman memenuhi persyaratan yang berlaku. Suatu proses pengawasan yaitu dengan uji keseragaman bobot. Uji keseragaman bobot dilakukan dengan cara yang tercantum dalam Buku Indonesia Edisi III.

Pengujian keseragama bobot tablet merupakan satu tahapan yang harus dikerjakan dalam

produksi tablet monitoring kualitas obat. Pengujian ini sangat penting karena berhubungan dengan dosis untuk pengobatan pasien, yang tiap pasien mempunyai spesifikasi dosis obat sesuai dengan penyakit yang di derita. Hal ini sangat membantu proses yang memerlukan kecermatan dan ketelitian dalam melakukan proses tahapan pengujian keseragam bobot tablet karena hal ini berhubungan dengan keselamatan jiwa pasien.

Komputasi sebetulnya bisa dijelaskan sebagai menemukan pemecahan masalah dari input yang diberikan dengan menggunakan algoritma. Selama ribuan tahun, perhitungan dilakukan dengan pena dan kertas, atau kapur dan batu tulis, atau secara mental, kadang-kadang dengan bantuan tabel.

Sehingga proses pengambilan keputusan apakah suatu proses produksi suatu tablet memenuhi syarat atau tidak akan lebih mudah dilakukan.

## 2. TEORI, MODEL, DAN DESAIN

Tablet adalah sediaan padat kompat, dibuat secara kempacetak, alam bentuk tabun pipih atau sirkuler, kedua permukaan rata atau cembung, mengandung satu jenis obat atau lebih dengan atau tanpa tambahan. Tablet yang akan diuji adalah tablet yang tidak bersalut. Proses pengujian kualitas tablet berdasarkan bobot diuji dengan metode uji keseragam bobot.

### Uji Keseragaman bobot

Tablet tidak bersalut harus memenuhi syarat keseragaman bobot yang ditetapkan sebagai berikut : timbang 2 tablet, hitung bobot rata-rata tiap tablet. Jika bobot ditimbang satu persatu, tidak boleh lebih dari 2 tablet yang masing-masing bobotnya menyimpang dari bobot rata-ratanya lebih besar dari harga yang ditetapkan kolom A, dan tidak satu tablet pun yang bobotnya menyimpang dari

bobot rata-ratanya lebih dari harga yang tetapkan kolom B. Jika tidak mencukupi 20 tablet dapat menggunakan 10 tablet : tidak satu tablet pun yang bobotnya menyimpang lebih besar dari bobot rata-rata yang ditetapkan kolom A dan tidak satu tablet pun yang menyimpang lebih besar dari bobot rata-rata yang ditetapkan kolom B.

Tabel 2.1. Tabel Uji Keseragaman Bobot

Bobot rata-rata	Penyimpangan bobot rata-rata dalam %	
	A	B
25 mg atau kurang	15%	30%
26 mg sampai dengan 150 mg	10%	20%
151 mg sampai dengan 300 mg	7.5%	15%
Lebih dari 300 mg	5%	10%

Toleransi bobot yang diujikan adalah 10% atau bobot yang diijinkan adalah

$$T_1 = B - (B * a\%)$$

$$T_2 = B + (B * a\%)$$

Keterangan

T1 = Toleransi batas bawah

T2 = Toleransi batas atas

B = bobot rata-rata tablet yang akan diuji

a = prosentase penyimpangan bobot rata-rata

Bobot toleransi yang diijinkan adalah antara T1 sampai dengan T2. Untuk Proses pengujian A, maksimal penyimpangan bobot tablet adalah 2 tablet sedang untuk Proses Pengujian B, maksimal penyimpangan bobot tablet adalah 2.

Misal :

Jika bobot tablet rata-rata pengujian adalah 100 mg maka

- Pengujian A, maka toleransi yang diijinkan adalah

$$T_1 = 100 \text{ mg} - (100 \text{ mg} * 10\%) = 90 \text{ mg}$$

$$T_2 = 100 \text{ mg} + (100 \text{ mg} * 10\%) = 110 \text{ mg}$$

Jadi tidak boleh lebih dari 2 dari 20 tablet yang diuji beratnya antara 90 mg – 110 mg

Maka tablet dinyatakan layak.

- Pengujian B, maka toleransi yang diijinkan adalah

$$T_1 = 100 \text{ mg} - (100 \text{ mg} * 20\%) = 80 \text{ mg}$$

$$T_2 = 100 \text{ mg} + (100 \text{ mg} * 20\%) = 120 \text{ mg}$$

Jadi tidak boleh lebih dari 1 dari 10 tablet yang

diuji beratnya antara 80 mg – 120 mg

Maka tablet dinyatakan layak.

Dari proses pengujian Uji Keseragaman bobot berdasarkan rumusan dari Farmakope Indonesia dibuat simulasi secara komputasi. Simulasi merupakan upaya untuk memahami pola-laku suatu sistem melalui modelnya. Dengan demikian prasyarat melakukan simulasi adalah dipunyainya model dari sistem. Model suatu sistem merupakan rumusan suatu hasil analisis dari sistem tersebut, yang pengungkapannya dapat bermacam-macam mulai dari model naratif sampai pada model matematis dan model komputer..

Metode yang digunakan untuk pembuatan sistem proses uji keseragaman bobot yaitu menemukan pemecahan masalah dari input yang diberikan dengan menggunakan algoritma. Unit input/output adalah bagian dari sistem mikroprosesor yang digunakan oleh mikroprosesor untuk berhubungan dengan dunia luar. Unit input adalah unit luar yang digunakan untuk memasukkan data dari luar ke dalam mikroprosesor, contohnya keyboard. Kemudian dilakukan proses pengujian yaitu untuk kriteria pengujian A dan pengujian B sehingga akan menghasilkan output. Output yang dihasilkan diharapkan dapat sebagai pendukung keputusan dalam menentukan kualitas tablet.

Sistem Pendukung Keputusan / *Decision support system* (DSS) merupakan sistem informasi yang membantu mengidentifikasi kesempatan pembuatan keputusan atau menyediakan informasi untuk membantu pembuatan keputusan. Dalam masalah ini DSS akan membantu memberikan keputusan apakah obat yang dihasilkan dalam suatu proses produksi tersebut layak atau tidak layak.

**Algoritma** (*algorithm*) merupakan kumpulan perintah untuk menyelesaikan suatu masalah. Proses algoritma ini dapat dijalankan sejalan bertahap dari awal hingga akhir. Masalah tersebut dapat berupa apa saja, dengan catatan untuk setiap masalah, ada kriteria kondisi awal yang harus dipenuhi sebelum menjalankan algoritma.

Jenis Algoritma yang digunakan adalah *Divide and Conquer* yaitu dengan paradigma membagi suatu permasalahan besar menjadi permasalahan-permasalahan yang lebih kecil. Pembagian masalah ini dilakukan terus menerus sampai ditemukan bagian masalah kecil yang mudah untuk dipecahkan. Singkatnya menyelesaikan keseluruhan masalah dengan membagi masalah besar dan kemudian mempecahkan permasalahan-permasalahan kecil yang terbentuk. Algoritma uji keseragaman bobot sebagai berikut :

### A. Algoritma untuk proses Input

1. Masukan nama tablet (TObat.nama\_tbt)
2. Masukan berat rata-rata tablet (TObat.berat\_tbt)
3. Pilih Jumlah tablet yang akan diuji  
TObat.T = 20 atau TObat.T = 10

### B. Algoritma untuk proses Pengujian A

1. Pengujian A, Bandingkan berat rata-rata tablet (TObat.berat\_tbt) dengan prosentasi toleransi tiap tablet  
Jika TObat.berat\_tbt <= 25  
Maka toleransi = 15  
Else Jika TObat.berat\_tbt <= 150  
Maka toleransi = 10  
Else Jika TObat.berat\_tbt <= 300  
Maka toleransi = 7,5  
Else  
Maka toleransi = 5
2. Hitung prosentasi berat toleransi  
 $R = \text{TObat.Berat\_tbt} * \text{toleransi} / 100$   
 $T1 = \text{TObat.berat\_tbt} - R$   
 $T2 = \text{TObat.berat\_tbt} + R$   
 $A = 0;$
3. Bandingkan selisih tiap-tiap tablet yang ditimbang dengan batas toleransi T1 dan T2 dengan berat hasil penimbangan tablet dari tablet 1 sampai jumlah tablet yang diuji (Tobat.T).
4. Jika TObat.tim[i] < T1 atau TObat.tim[i] > T2  
Maka A = A+1

### C. Algoritma untuk proses Pengujian B

1. Pengujian B, Bandingkan berat rata-rata tablet (TObat.berat\_tbt) dengan prosentasi toleransi tiap tablet  
Jika TObat.berat\_tbt <= 25  
Maka toleransi = 30  
Else Jika TObat.berat\_tbt <= 150  
Maka toleransi = 20  
Else Jika TObat.berat\_tbt <= 300  
Maka toleransi = 15  
Else  
Maka toleransi = 5
2. Hitung prosentasi berat toleransi  
 $R = \text{TObat.Berat\_tbt} * \text{toleransi} / 100$   
 $T1 = \text{TObat.berat\_tbt} - R$   
 $T2 = \text{TObat.berat\_tbt} + R$   
 $A = 0;$
3. Bandingkan selisih tiap-tiap tablet yang ditimbang dengan batas toleransi T1 dan T2

dengan berat hasil penimbangan tablet dari tablet 1 sampai jumlah tablet yang diuji (Tobat.T).

4. Jika TObat.tim[i] < T1 atau TObat.tim[i] > T2  
Maka A = A+1

### D. Algoritma untuk proses Output / Hasil

1. Jika TObat.T = 20  
Maka  
Jika A <= 2 atau B <= 1  
Maka Hasil = "Layak"  
Else Hasil = "Tidak Layak"
2. Jika TObat.T = 10  
Maka  
Jika A <= 1 atau B <= 1  
Maka Hasil = "Tidak Layak"  
Else Hasil = "Layak"

## 3. HASIL DAN ANALISIS

Dari algoritma yang ada kemudian disusun code program, kode program ditulis dengan aplikasi C++ Builder, Hasil berupa output kesimpulan dari hasil pengujian yang dilakukan apakah suatu produksi tablet obat layak atau tidak, sehingga obat yang ada di pasaran tidak merugikan pasien / konsumen karena penyimpang berat obat yang.

Tabel 3.1. Contoh Pengujian Uji Keseragaman Bobot

Nama Obat : Panadol					
Berat rata-rata = 100 mg					
Jumlah tablet yang di uji 20 butir					
Penyimpangan rata Pengujian A = 10 % ; T1 = 90 mg ; T2 = 110					
No	Berat penimbang (mg)	Pengujian A <T1	Pengujian A >T2	Pengujian B <T1	Pengujian B >T2
1	101	-	-	-	-
2	102	-	-	-	-
3	98	-	-	-	-
4	102	-	-	-	-
5	90	-	-	-	-
6	100	-	-	-	-
7	100	-	-	-	-
8	102	-	-	-	-
9	89	□	-	-	-
10	98	-	-	-	-
Banyak Penyimpangan		1		0	
Hasil				Layak	

Dari algoritma yang telah dibuat disusun kode program untuk Proses Uji keseragaman bobot sebagai berikut :

```
//  
#include <vcl.h>  
#pragma hdrstop  
#include "Uobat.h"  
//
```

```

#pragma package(smart_init)
#pragma resource "*.dfm"
TFormObat *FormObat;
struct Obat
{
String nama_tbt;
int berat_tbt;
int T;
float berat_tim[20];
bool layak;
} TObat;
int n, A, B;
float nm, toleransi, R, T1, T2;
float berat_toleransi[20];
String hasil;
//_____
__fastcall TFormObat::TFormObat(TComponent* Owner)
: TForm(Owner)
{
}
//_____
void __fastcall TFormObat::FormCreate(TObject *Sender)
{
EditNama->Enabled = true;
RadioGroupJum->Enabled = true;
EditBerat->Enabled = true;
LabelKe->Visible = false;
EditTimbang->Enabled = false;

StringGrid1->Cells[1][0] = "Penimbangan";
StringGrid1->Cells[2][0] = "Berat";
Button5->Enabled = false;
Button3->Enabled = false;
ButtonHasil->Enabled = false;
ButtonOk->Enabled = false;
}
//_____
void __fastcall TFormObat::ButtonProsesClick(TObject
*Sender)
{
if (EditNama->GetTextLen() == 0)
{
EditNama->SetFocus();
return;
}
if (EditBerat->GetTextLen() == 0)
{
EditBerat->SetFocus();
return;
}
if (RadioGroupJum->ItemIndex == -1)
{
RadioGroupJum->SetFocus();
return;
}

EditNama->Enabled = false;
RadioGroupJum->Enabled = false;
EditBerat->Enabled = false;
n = 1;
LabelKe->Caption = IntToStr(n);
LabelKe->Visible = true;

EditTimbang->Enabled = true;
TObat.nama_tbt = EditNama->Text;
TObat.berat_tbt = StrToInt(EditBerat->Text);
if (RadioGroupJum->ItemIndex == 0 )
TObat.T = 10 ;
else if (RadioGroupJum->ItemIndex == 1 )
TObat.T = 20 ;

EditTimbang->SetFocus();
ButtonProses->Enabled = false;
ButtonOk->Enabled = true;
ListBoxHasil->Clear();
}
//_____
void __fastcall TFormObat::Button1Click(TObject *Sender)
{
Close();
}
//_____
void __fastcall TFormObat::ButtonHasilClick(TObject
*Sender)
{
if (TObat.T == 20)
{
if (TObat.berat_tbt <= 25)
toleransi = 15;
else if (TObat.berat_tbt <= 150)
toleransi = 10;
else if (TObat.berat_tbt <= 300)
toleransi = 7,5;
else
toleransi = 5;

R = TObat.berat_tbt * toleransi / 100;
A = 0;
T1 = TObat.berat_tbt - R;
T2 = TObat.berat_tbt + R;

for(int i=1; i<=TObat.T; i++)
{
TObat.berat_tim[i] = StrToFloat(StringGrid1->Cells[2][i]);

if ((TObat.berat_tim[i] < T1) || (TObat.berat_tim[i] > T2))
A = A + 1;
}

if (TObat.berat_tbt <= 25)
toleransi = 30;
else if (TObat.berat_tbt <= 150)
toleransi = 20;
else if (TObat.berat_tbt <= 300)
toleransi = 15;
else
toleransi = 10;

R = TObat.berat_tbt * toleransi / 100;
B = 0;
T1 = TObat.berat_tbt - R;
T2 = TObat.berat_tbt + R;

for(int i=1; i<=TObat.T; i++)
{
TObat.berat_tim[i] = StrToFloat(StringGrid1->Cells[2][i]);

if ((TObat.berat_tim[i] < T1) || (TObat.berat_tim[i] > T2))
B = B + 1;
}
}

```

```

    }
}
else
{
if (TObat.berat_tbt <= 25)
toleransi = 15;
else if (TObat.berat_tbt <= 150)
toleransi = 10;
else if (TObat.berat_tbt <= 300)
toleransi = 7,5;
else
toleransi = 5;
R = TObat.berat_tbt * toleransi / 100;
A = 0;
T1 = TObat.berat_tbt - R;
T2 = TObat.berat_tbt + R;
for(int i=1; i<=TObat.T; i++)
{
TObat.berat_tim[i]=StrToFloat(StringGrid1->Cells[2][i]);
if ((TObat.berat_tim[i] < T1) || (TObat.berat_tim[i] > T2))
A = A + 1;
}
if (TObat.berat_tbt <= 25)
toleransi = 30;
else if (TObat.berat_tbt <= 150)
toleransi = 20;
else if (TObat.berat_tbt <= 300)
toleransi = 15;
else
toleransi = 10;
R = TObat.berat_tbt * toleransi / 100;
B = 0;
T1 = TObat.berat_tbt - R;
T2 = TObat.berat_tbt + R;
for(int i=1; i<=TObat.T; i++)
{
TObat.berat_tim[i]=StrToFloat(StringGrid1->Cells[2][i]);
if ((TObat.berat_tim[i] < T1) || (TObat.berat_tim[i] > T2))
B = B + 1;

if (TObat.T == 10)
if ((A <= 1) && (B <= 1))
TObat.layak = true;
hasil = "Layak";
else
TObat.layak = false;
hasil = "Tidak Layak";
else
if ((A <= 2) && (B <= 1))
{
TObat.layak = true;
hasil = "Layak";
}
else
{
TObat.layak = false;
hasil = "Tidak Layak";
}

ListBoxHasil->Items->Add("Nama Tablet : ");
ListBoxHasil->Items->Add(TObat.nama_tbt);
ListBoxHasil->Items->Add("Berat Rata-rata : ");
ListBoxHasil->Items->Add(TObat.berat_tbt);
ListBoxHasil->Items->Add("Banyak Tablet yang di uji : ");

```

```

    ListBoxHasil->Items->Add(TObat.T);
    ListBoxHasil->Items->Add("Berat Tablet hasil Pengujian : ");
    for(int i=1; i<=TObat.T; i++)
    ListBoxHasil->Items->Add(FloatToStr(TObat.berat_tim[i]));
    ListBoxHasil->Items->Add("Hasil Uji : ");
    ListBoxHasil->Items->Add("Banyaknya penyimpangan");
    ListBoxHasil->Items->Add("Pengujian A : " + IntToStr(A) + " Tablet");
    ListBoxHasil->Items->Add("Pengujian B : " + IntToStr(B) + " Tablet");
    ListBoxHasil->Items->Add("Kesimpulan : ");
    ListBoxHasil->Items->Add(hasil);

ButtonHasil->Enabled = false;
Button5->Enabled = true;
Button3->Enabled = true;
for(int i=1; i<=TObat.T; i++)
{
StringGrid1->Cells[1][i] = "";
StringGrid1->Cells[2][i] = "";
}

ButtonProses->Enabled = true;
EditNama->Enabled = true;
EditNama->Text = "";
RadioGroupJum->Enabled = true;
EditBerat->Enabled = true;
RadioGroupJum->ItemIndex = -1;
EditBerat->Text = "";
StringGrid1->Enabled = false;
}

//_____________________________________
void __fastcall TFormObat::Button3Click(TObject *Sender)
{
if (SaveDialog1->Execute())
ListBoxHasil->Items->SaveToFile(SaveDialog1->FileName);
Button5->Enabled = true;
}
//_____________________________________
void __fastcall TFormObat::Button4Click(TObject *Sender)
{
if (OpenDialog1->Execute())
ListBoxHasil->Items->LoadFromFile(OpenDialog1->FileName);
Button5->Enabled = true;
}
//_____________________________________
void __fastcall TFormObat::Button2Click(TObject *Sender)
{
ButtonProses->Enabled = true;
EditNama->Enabled = true;
EditNama->Text = "";
RadioGroupJum->Enabled = true;
EditBerat->Enabled = true;
RadioGroupJum->ItemIndex = -1;
EditBerat->Text = "";

LabelKe->Visible = false;
for(int i=1; i<=TObat.T; i++)
{
StringGrid1->Cells[1][i] = "";
StringGrid1->Cells[2][i] = "";
}
ListBoxHasil->Clear();
StringGrid1->Enabled = false;
ButtonHasil->Enabled = false;
}

```

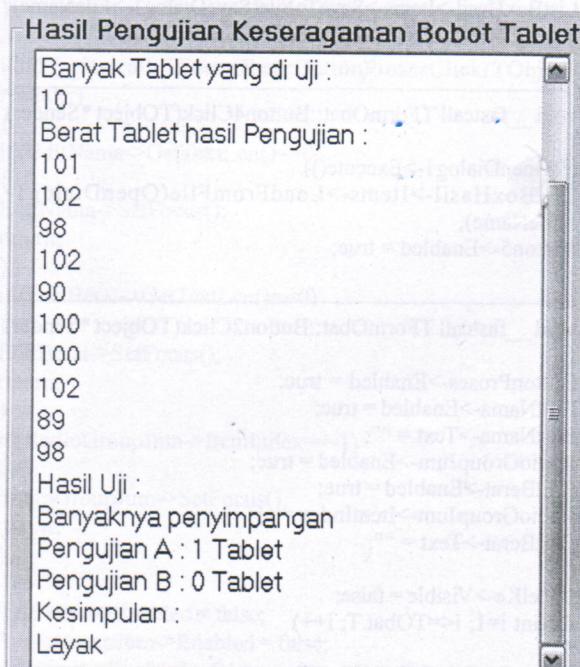
```

}

//—————
void __fastcall TFormObat::Button5Click(TObject *Sender)
{
    ListBoxHasil->Items->Add("Nama Tablet : ");
    EditNama->Text = ListBoxHasil->Items->Strings[1];
    TObat.nama_tbt = ListBoxHasil->Items->Strings[1];
    EditBerat->Text = ListBoxHasil->Items->Strings[3];
    TObat.berat_tbt = StrToFloat(ListBoxHasil->Items->Strings[3]);
    if (ListBoxHasil->Items->Strings[5] == "10")
    {
        TObat.T = 10;
        RadioGroupJum->ItemIndex = 0;
    }
    else
    {
        TObat.T = 20;
        RadioGroupJum->ItemIndex = 1;
    }
    for(int i=1; i<=TObat.T; i++)
    {
        StringGrid1->Cells[1][i] = IntToStr(i);
        StringGrid1->Cells[2][i] = ListBoxHasil->Items->Strings[i+6];
    }
    ListBoxHasil->Items->Clear();
    StringGrid1->Enabled = true;
    Button5->Enabled = false;
    ButtonHasil->Enabled = true;
}
//—————

```

Dari program tersebut akan memberikan hasil sebagai berikut :



Gambar 3.1 Hasil Pengujian Uji Keseragaman Obat

Dari hasil perhitungan berdasarkan tabel 2.1 dibandingkan dengan hasil perhitungan sistem diperoleh hasil yang sama. Sehingga tingkat Accuracy / Ketepatan sistem mencirikan tingkat kebenaran dari fungsi, menjadi lebih cepat pengambilan keputusan.

#### 4. PENUTUP

Pengujian uji keseragaman bobot dapat lebih mudah dilakukan oleh sistem sehingga akan membantu pihak terkait seperti apotik atau badan pengawasan obat dan pangan dalam mengawasi salah satu variable kelayakan suatu obat. Dari hasil pengujian sistem dan perhitungan uji keseragaman bobot memperoleh hasil yang dilaksanakan dapat membantu dalam proses pengambilan keputusan apakah suatu proses produksi suatu tablet memenuhi syarat atau tidak.

#### DAFTAR PUSTAKA

- [1] Departemen Kesehatan Republik Indonesia, "Farmakope Indonesia Edisi Ketiga", Indonesia, 1997.
- [2] <http://id.wikipedia.org>
- [3] Efrain Turban, Jay E. Aronson, Ting-Peng Liang, "Decision Support System and Intelligent System, Jilid 1", Andi, Yogyakarta, 2005.
- [4] Jeffrey L.Whitten, Lannu D. Bentley, Kevin C.Diftman, "Metode Desain & Analisis System", Andi, Yogyakarta, 2004.