**LAPORAN PROYEK AKHIR**

**SISTEM CERDAS DAN PENDUKUNG KEPUTUSAN**

**SEMESTER GENAP T.A 2023/2024**

# **HALAMAN JU**

# 

# **DUL**

**A picture containing emblem, logo, clipart, symbol

Description automatically generated**

**DISUSUN OLEH :**

**NIM : 123220052**

**123220020**

**NAMA : NAUFAL RAFID MUHAMMAD FADDILA**

**DEA REIGINA**

**KELAS : IF-E**

**NAMA DOSEN : Dr. HERLINA JAYADIANTI., S.T., M.T.**

**PROGRAM STUDI INFORMATIKA**

**JURUSAN INFORMATIKA**

**FAKULTAS TEKNIK INDUSTRI**

**UNIVERSITAS PEMBANGUNAN NASIONAL ”VETERAN” YOGYAKARTA**

**2023**

# **DAFTAR ISI**

HALAMAN JUDULi

DAFTAR ISI1

JUDUL PROYEK AKHIR2

BAB I PENDAHULUAN3

1.1 Latar Belakang Masalah3

1.2 Tujuan Proyek Akhir3

1.3 Manfaat Proyek Akhir3

BAB II PEMBAHASAN4

2.1 Dasar Teori4

2.2 Deskripsi Umum Proyek Akhir4

2.3 Inti Pembahasan5

BAB IIII JADWAL PENGERJAAN DAN PEMBAGIAN TUGAS19

3.1 Jadwal Pengerjaan19

3.2 Pembagian Tugas19

BAB IV KESIMPULAN DAN SARAN20

4.1 Kesimpulan20

4.2 Saran20

DAFTAR PUSTAKA21

# **Seleksi Calon Penerima Bansos**

# **BAB I**

# **PENDAHULUAN**

## **Latar Belakang Masalah**

Pemerintah sering kali menghadapi tantangan dalam mendistribusikan bantuan sosial (bansos) secara adil dan tepat sasaran. Proses seleksi penerima bansos melibatkan berbagai faktor seperti tingkat pendapatan, jumlah tanggungan keluarga, kondisi kesehatan, dan status pekerjaan. Namun, faktor-faktor ini sering kali bersifat subjektif dan ambigu, sehingga menyulitkan penyeleksi untuk membuat keputusan yang konsisten dan objektif. Seleksi manual juga memakan waktu dan rentan terhadap kesalahan serta bias.

Untuk mengatasi masalah ini, diperlukan Sistem Pendukung Keputusan (SPK) berbasis Fuzzy Logic. Fuzzy Logic memungkinkan pengambilan keputusan yang lebih fleksibel dengan mempertimbangkan berbagai tingkat keanggotaan dari kriteria yang ambigu. SPK berbasis Fuzzy Logic membantu proses seleksi penerima bansos menjadi lebih efisien, objektif, dan konsisten.

Program Seleksi Calon Penerima Bansos berbasis Fuzzy Logic ini dirancang untuk menyederhanakan proses seleksi dengan mempertimbangkan kriteria yang ambigu dan mengurangi subjektivitas. Dengan sistem ini, diharapkan distribusi bansos dapat dilakukan secara lebih tepat sasaran, adil, dan transparan, sehingga bantuan benar-benar sampai kepada mereka yang membutuhkan.

## **Tujuan Proyek Akhir**

Tujuan dari dibuatnya Program Seleksi Calon Penerima Bansos adalah untuk mengembangkan sebuah sistem pendukung keputusan yang dapat melakukan perbandingan antara pembentuk himpunan *fuzzy* tiap kriteria dengan kasus yang terjadi secara nyata di lapangan. Adanya sistem dengan metode logika *fuzzy* ini, diharapkan dapat membantu dalam membandingkan kriteria yang sudah ditetapkan.

**1.3 Manfaat Proyek Akhir**

1. Mengurangi waktu dan usaha yang dibutuhkan dalam menyeleksi penerima bantuan sosial dengan mengotomatisasi proses penilaian dan rekomendasi.
2. Mengurangi subjektivitas dan bias dalam penilaian dengan menggunakan pendekatan Fuzzy Logic, sehingga keputusan yang dihasilkan lebih konsisten dan adil.
3. Meningkatkan transparansi dalam proses seleksi, memungkinkan penyeleksi dan penerima bansos memahami dasar keputusan yang diambil.
4. Mengurangi beban kerja penyeleksi dengan menyediakan alat bantu yang mudah digunakan dan mengotomatiskan sebagian besar proses penilaian.

# **BAB II**

# **PEMBAHASAN**

## **2.1 Dasar Teori**

Fuzzy mengarah pada sesuatu yang ambigu dan tidak pasti. DI dunia nyata, kita pasti pernah berhadapan dengan situasi di mana kita tidak dapat menentukan sesuatu tersebut benar atau salah. Dengan logika fuzzy, kita dapat mempertimbangkan ketidakakuratan dan ketidakpastian dalam segala situasi. Fuzzy dapat berupa variabel dari bilangan *real* antara 0 dan 1, yaitu variabel yang berada di antara benar atau salah.

Fuzzy digunakan untuk menyelesaikan hitungan dengan informasi yang tidak menentu dan merupakan metode dalam matematika untuk merepresentasikan keambiguan dalam pengambilan keputusan. Konsep dari benar atau salah terlalu ketat dan terdapat bayangan di antara keduanya. Logika Fuzzy memungkinkan kita untuk mendapatkan hasil setengah benar, yaitu di mana pernyataan dapat menjadi sebagian benar atau salah, daripada benar atau salah sepenuhnya.

## **2.2 Deskripsi Umum Proyek Akhir**

Program ini bertujuan untuk mengembangkan sebuah sistem atau model computer yang dapat menentukan hasil seleksi dari calon penerima bantuan social berdasarkan kriterianya. Metode yang digunakan dalam proyek ini adalah logika fuzzy, merupakan model pendukung Keputusan dengan membagi kriteria yang memiliki keterangan ambigu menjadi beberapa himpunan fuzzy, sehingga mempermudah dalam membuat Keputusan yang tepat.

Untuk mempermudah menentukan siapa yang berhak menerima bantuan social, ditetapkan tiga aspek yang perlu diperhatikan dari para calon penerima, yaitu pendapatan perbulan, jumlah tanggungan dan kondisi rumah. Sistem seleksi penerima bantuan social ini menyeleksi para calon penerima dengan 3 kriteria diatas sebagai input dengan keterangan:

1. Pendapatan terdiri dari 3 himpunan fuzzy, yaitu rendah, sedang dan tinggi.

2. jumlah tanggungan terdiri dari 3 himpunan fuzzy, yaitu sedikit, sedang dan banyak.

3. kondisi rumah terdiri dari 3 himpunan fuzzy, yaitu buruk, cukup dan baik.

Hasil dari sistem hanya ada satu akan menunjukkan apakah calon penerima tersebut diterima, dipertimbangkan atau ditolak sebagai penerima.

## **2.2 Inti Pembahasan**

1. **Fuzzy dan Himpunannya**
2. Pendapatan

|  |  |
| --- | --- |
| Himpunan | Range (Rupiah) |
| Rendah | 0 – 2.700.000 |
| Sedang | 2.000.000 – 4.000.000 |
| Tinggi | 3.500.000 – 5.000.000 |

1. Himpunan fungsi pendapatan: rendah

1. Himpunan fungsi pendapatan: sedang
2. Himpunan fungsi pendapatan: tinggi
3. Jumlah tanggungan

|  |  |
| --- | --- |
| Himpunan | Range (0 – 10) |
| Sedikit | 0 – 2 |
| Sedang | 1 – 5 |
| Banyak | 4 – 10 |

1. Himpunan jumlah tanggungan: sedikit
2. Himpunan jumlah tanggungan: sedang
3. Himpunan jumlah tanggungan: banyak
4. Kondisi rumah

|  |  |
| --- | --- |
| Himpunan | Range (0 – 10) |
| Buruk | 0 – 4.5 |
| Cukup | 3 – 7 |
| Baik | 5.5 - 10 |

1. Himpunan fungsi kondisi rumah: buruk
2. Himpunan fungsi kondisi rumah: cukup
3. Himpunan fungsi kondisi rumah: baik
4. Hasil

|  |  |
| --- | --- |
| Himpunan | Range (0 – 10) |
| Ditolak | 0 – 4.5 |
| Dipertimbangkan | 3 – 7.5 |
| Diterima | 6.5 - 10 |

1. Himpunan fungsi keanggotaan hasil: ditolak
2. Himpunan fungsi keanggotaan hasil: dipertimbangkan
3. Himpunan fungsi keanggotaan hasil: diterima
4. Aturan (rules)

Aturan yang akan digunakan dalam sistem seleksi calon penerima bantuan social, sebagai berikut:

1. Jika Pendapatan Rendah dan Jumlah Tanggungan Banyak dan Kondisi Rumah Buruk maka Hasil Diterima
2. Jika Pendapatan Rendah dan Jumlah Tanggungan Banyak dan Kondisi Rumah cukup maka Hasil Diterima
3. Jika Pendapatan Rendah dan Jumlah Tanggungan Banyak dan Kondisi Rumah Baik maka Hasil Diterima
4. Jika Pendapatan Rendah dan Jumlah Tanggungan sedang dan Kondisi Rumah Buruk maka Hasil Diterima
5. Jika Pendapatan Rendah dan Jumlah Tanggungan sedang dan Kondisi Rumah cukup maka Hasil Dipertimbangkan
6. Jika Pendapatan Rendah dan Jumlah Tanggungan sedang dan Kondisi Rumah Baik maka Hasil Dipertimbangkan
7. Jika Pendapatan Rendah dan Jumlah Tanggungan sedikit dan Kondisi Rumah Buruk maka Hasil Dipertimbangkan
8. Jika Pendapatan rendah dan Jumlah Tanggungan sedikit dan Kondisi Rumah cukup maka Hasil ditolak
9. Jika Pendapatan rendah dan Jumlah Tanggungan sedikit dan Kondisi Rumah Baik maka Hasil Ditolak
10. Jika Pendapatan sedang dan Jumlah Tanggungan Banyak dan Kondisi Rumah Buruk maka Hasil Diterima
11. Jika Pendapatan sedang dan Jumlah Tanggungan Banyak dan Kondisi Rumah cukup maka Hasil Diterima
12. Jika Pendapatan sedang dan Jumlah Tanggungan Banyak dan Kondisi Rumah baik maka Hasil Dipertimbangkan
13. Jika Pendapatan sedang dan Jumlah Tanggungan sedang dan Kondisi Rumah Buruk maka Hasil Dipertimbangkan
14. Jika Pendapatan sedang dan Jumlah Tanggungan sedang dan Kondisi Rumah cukup maka Hasil Ditolak
15. Jika Pendapatan sedang dan Jumlah Tanggungan sedang dan Kondisi Rumah Baik maka Hasil Ditolak
16. Jika Pendapatan sedang dan Jumlah Tanggungan sedikit dan Kondisi Rumah Buruk maka Hasil Dipertimbangkan
17. Jika Pendapatan sedang dan Jumlah Tanggungan sedikit dan Kondisi Rumah cukup maka Hasil ditolak
18. Jika Pendapatan sedang dan Jumlah Tanggungan sedikit dan Kondisi Rumah baik maka Hasil ditolak
19. Jika Pendapatan tinggi dan Jumlah Tanggungan banyak dan Kondisi Rumah buruk maka Hasil dipertimbangkan
20. Jika Pendapatan tinggi dan Jumlah Tanggungan banyak dan Kondisi Rumah cukup maka Hasil ditolak
21. Jika Pendapatan tinggi dan Jumlah Tanggungan banyak dan Kondisi Rumah baik maka Hasil ditolak
22. Jika Pendapatan tinggi dan Jumlah Tanggungan sedang dan Kondisi Rumah buruk maka Hasil dipertimbangkan
23. Jika Pendapatan tinggi dan Jumlah Tanggungan sedang dan Kondisi Rumah cukup maka Hasil ditolak
24. Jika Pendapatan tinggi dan Jumlah Tanggungan sedang dan Kondisi Rumah baik maka Hasil ditolak
25. Jika Pendapatan tinggi dan Jumlah Tanggungan sedikit dan Kondisi Rumah buruk maka Hasil ditolak
26. Jika Pendapatan tinggi dan Jumlah Tanggungan sedikit dan Kondisi Rumah cukup maka Hasil ditolak
27. Jika Pendapatan tinggi dan Jumlah Tanggungan sedikit dan Kondisi Rumah baik maka Hasil ditolak

Untuk memperjelas detail perhitungan dari logika fuzzy, diberikan contoh dengan cukup tiga aturan saja. Misalkan ada seorang kepala keluarga bernama Pak Samsudin dengan Pendapatan 2.500.000 yang terdaftar sebagai warga desa sukamaju. Dia memiliki tanggungan 4 orang yaitu satu orang anak dan satu orang istri. Keadaan rumahnya cukup bagus dan layak huni dengan penilaian 6.8. Apakah dia akan diterima sebagai penerima bansos dengan aturan sebagai berikut.

* IF pendapatan sedikit AND jumlah tanggungan sedang AND kondisi rumah cukup THEN Hasil diterima
* IF Pendapatan sedang AND Jumlah tanggungan sedang AND Kondisi rumah buruk THEN H dipertimbangkan
* IF Pendapatan tinggi AND Jumlah tanggungan banyak AND Kondisi rumah cukup THEN Hasil ditolak

Jawab:

* Pendapatan sedikit: (2.700.000-2.500.000/2.700.000-500.000) = 0.09
* Pendapatan sedang: 0
* Pendapatan banyak: 0
* Jumlah tanggungan sedikit: 0
* Jumlah tanggungan sedang: (5-4/5-3) = 0.5
* Jumlah tanggungan banyak: 0
* Kondisi rumah buruk : 0
* Kondisi rumah cukup: (7-6.8/7-5) = 0.1
* Kondisi rumah baik: 0
* Rule 1

IF pendapatan sedikit AND jumlah tanggungan sedang AND kondisi rumah cukup THEN Hasil diterima

a1 = min(0.09;0.5;0.1) = 0.09

Untuk a1 = 0.09 pada himpunan H dipertimbangkan

= (8.5 - x)/(8.5 – 6.5) = 0.09

x1 = 8.32

* Rule 2

IF Pendapatan sedang AND Jumlah tanggungan sedang AND Kondisi rumah buruk THEN H dipertimbangkan

a2 = min(0;0.5;0) = 0

Untuk a2 = 0.5 pada himpunan H tidak\_diterima

= (4.5 - x)/(4.5 – 1.25) = 0.5

x2 = 2.875

* Rule 3

IF Pendapatan tinggi AND Jumlah tanggungan banyak AND Kondisi rumah cukup THEN Hasil ditolak

a3 = min(0;0;0.1) = 0.1

Untuk a3 = 0.1 pada himpunan H diterima

= (8.5 - x)/(8.5 – 6.5) = 0.1

x3 = 8.3.

* Hasil

Z=

=

= 4.375

Jadi hasilnya, yaitu = 4.813 (Dipertimbangkan)

1. **Listing Program**

Listing yang digunakan dalam program ini hanya satu jenis saja. Program menggunakan GUI agar lebih mudah digunakan penyeleksi. Berikut listing program yang digunakan.

|  |
| --- |
| [System]  Name='Penerimaan\_Beasiswa'  Type='mamdani'  Version=2.0  NumInputs=3  NumOutputs=1  NumRules=27  AndMethod='min'  OrMethod='max'  ImpMethod='min'  AggMethod='max'  DefuzzMethod='centroid'  [Input1]  Name='ipk'  Range=[3 4]  NumMFs=3  MF1='cukup':'trapmf',[0 2.5 3.1 3.4] |

**Program 2.1 Fuzzy**

|  |
| --- |
| MF2='baik':'trimf',[3.3 3.5 3.7]  MF3='sangat\_baik':'trapmf',[3.6 3.9 4 4]  [Input2]  Name='wawancara'  Range=[0 10]  NumMFs=3  MF1='kurang':'trapmf',[0 0 2 4.5]  MF2='cukup':'trimf',[3 5 7]  MF3='bagus':'trapmf',[5.5 8 10 10]  [Input3]  Name='gaji\_ortu'  Range=[500000 5000000]  NumMFs=3  MF1='rendah':'trapmf',[0 312500 2000000 2700000]  MF2='sedang':'trimf',[2000000 3000000 4000000]  MF3='tinggi':'trapmf',[3500000 4750000 5188000 6688000]  [Output1]  Name='hasil'  Range=[0 10]  NumMFs=3  MF1='Tidak\_Diterima':'trapmf',[0 0 1.25 4.5]  MF2='Dipertimbangkan':'trimf',[3 5.25 7.5]  MF3='Diterima':'trapmf',[6.5 8.5 10 10]  [Rules]  1 1 1, 1 (1) : 1  1 1 2, 1 (1) : 1  1 1 3, 1 (1) : 1  1 2 1, 1 (1) : 1  1 2 2, 1 (1) : 1  1 2 3, 1 (1) : 1  1 3 1, 2 (1) : 1  1 3 2, 1 (1) : 1  1 3 3, 1 (1) : 1  2 1 1, 2 (1) : 1  2 1 2, 1 (1) : 1  2 1 3, 1 (1) : 1  2 2 1, 2 (1) : 1  2 2 2, 2 (1) : 1  2 2 3, 1 (1) : 1  2 3 1, 3 (1) : 1  2 3 2, 2 (1) : 1  2 3 3, 2 (1) : 1  3 1 1, 3 (1) : 1  3 1 2, 2 (1) : 1  3 1 3, 1 (1) : 1  3 2 1, 3 (1) : 1  3 2 2, 3 (1) : 1  3 2 3, 2 (1) : 1  3 3 1, 3 (1) : 1  3 3 2, 3 (1) : 1  3 3 3, 3 (1) : 1 |

**Lanjutan Program 2.1 Fuzzy**

|  |
| --- |
| function varargout = GUI(varargin)  % GUI MATLAB code for GUI.fig  % GUI, by itself, creates a new GUI or raises the existing  % singleton\*.  % H = GUI returns the handle to a new GUI or the handle to  % the existing singleton\*.  % GUI('CALLBACK',hObject,eventData,handles,...) calls the local  % function named CALLBACK in GUI.M with the given input arguments.  %  % GUI('Property','Value',...) creates a new GUI or raises the  % existing singleton\*. Starting from the left, property value pairs are  % applied to the GUI before GUI\_OpeningFcn gets called. An  % unrecognized property name or invalid value makes property application  % stop. All inputs are passed to GUI\_OpeningFcn via varargin.  % \*See GUI Options on GUIDE's Tools menu. Choose "GUI allows only one  % instance to run (singleton)".  %  % See also: GUIDE, GUIDATA, GUIHANDLES  % Edit the above text to modify the response to help GUI  % Last Modified by GUIDE v2.5 03-Jun-2023 23:40:35  % Begin initialization code - DO NOT EDIT  gui\_Singleton = 1;  gui\_State = struct('gui\_Name', mfilename, ...  'gui\_Singleton', gui\_Singleton, ...  'gui\_OpeningFcn', @GUI\_OpeningFcn, ...  'gui\_OutputFcn', @GUI\_OutputFcn, ...  'gui\_LayoutFcn', [] , ...  'gui\_Callback', []);  if nargin && ischar(varargin{1})  gui\_State.gui\_Callback = str2func(varargin{1});  end  if nargout  [varargout{1:nargout}] = gui\_mainfcn(gui\_State, varargin{:});  else  gui\_mainfcn(gui\_State, varargin{:});  end  % End initialization code - DO NOT EDIT  % --- Executes just before GUI is made visible.  function GUI\_OpeningFcn(hObject, eventdata, handles, varargin)  % This function has no output args, see OutputFcn.  % hObject handle to figure  % eventdata reserved - to be defined in a future version of MATLAB  % handles structure with handles and user data (see GUIDATA)  % varargin command line arguments to GUI (see VARARGIN)  % Choose default command line output for GUI  handles.output = hObject;  % Update handles structure  guidata(hObject, handles);  % UIWAIT makes GUI wait for user response (see UIRESUME) |
| **Program 2.2 keseluruhan GUI Matlab** |

|  |
| --- |
| % uiwait(handles.figure1);  % --- Outputs from this function are returned to the command line.  function varargout = GUI\_OutputFcn(hObject, eventdata, handles)  % varargout cell array for returning output args (see VARARGOUT);  % hObject handle to figure  % eventdata reserved - to be defined in a future version of MATLAB  % handles structure with handles and user data (see GUIDATA)  % Get default command line output from handles structure  varargout{1} = handles.output;  % --- Executes on slider movement.  function ipkSlider\_Callback(hObject, eventdata, handles)  % hObject handle to ipkSlider (see GCBO)  % eventdata reserved - to be defined in a future version of MATLAB  % handles structure with handles and user data (see GUIDATA)  sliderValue = get(hObject, 'Value');  % Mengupdate text samping slider  set(handles.ipkText, 'String', num2str(sliderValue));  % Hints: get(hObject,'Value') returns position of slider  % get(hObject,'Min') and get(hObject,'Max') to determine range of slider  % --- Executes during object creation, after setting all properties.  % Get default command line output from handles structure  varargout{1} = handles.output;  % --- Executes on slider movement.  function ipkSlider\_Callback(hObject, eventdata, handles)  % hObject handle to ipkSlider (see GCBO)  % eventdata reserved - to be defined in a future version of MATLAB  % handles structure with handles and user data (see GUIDATA)  sliderValue = get(hObject, 'Value');  % Mengupdate text samping slider  set(handles.ipkText, 'String', num2str(sliderValue));  % Hints: get(hObject,'Value') returns position of slider  % get(hObject,'Min') and get(hObject,'Max') to determine range of slider  % --- Executes during object creation, after setting all properties.  function ipkSlider\_CreateFcn(hObject, eventdata, handles)  % hObject handle to ipkSlider (see GCBO)  % eventdata reserved - to be defined in a future version of MATLAB  % handles empty - handles not created until after all CreateFcns called  % Hint: slider controls usually have a light gray background.  if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))  set(hObject,'BackgroundColor',[.9 .9 .9]);  end |
| **Lanjutan Program 2.2 Keseluruhan GUI Matlab** |

|  |
| --- |
| % --- Executes on slider movement.  function wawancaraSlider\_Callback(hObject, eventdata, handles)  % hObject handle to wawancaraSlider (see GCBO)  % eventdata reserved - to be defined in a future version of MATLAB  % handles structure with handles and user data (see GUIDATA)  sliderValue = get(hObject, 'Value');  % Mengupdate text samping slider  set(handles.wawancaraText, 'String', num2str(sliderValue));  % Hints: get(hObject,'Value') returns position of slider  % get(hObject,'Min') and get(hObject,'Max') to determine range of slider  % --- Executes during object creation, after setting all properties.  function wawancaraSlider\_CreateFcn(hObject, eventdata, handles)  % hObject handle to wawancaraSlider (see GCBO)  % eventdata reserved - to be defined in a future version of MATLAB  % handles empty - handles not created until after all CreateFcns called  % Hint: slider controls usually have a light gray background.  if isequal(get(hObject,'BackgroundColor'),  get(0,'defaultUicontrolBackgroundColor'))  set(hObject,'BackgroundColor',[.9 .9 .9]);  end  % --- Executes on slider movement.  function gajiSlider\_Callback(hObject, eventdata, handles)  % hObject handle to wawancaraText (see GCBO)  % eventdata reserved - to be defined in a future version of MATLAB  % handles structure with handles and user data (see GUIDATA)  sliderValue = get(hObject, 'Value');  % Mengupdate text samping slider  set(handles.gajiText, 'String', num2str(sliderValue));  % Hints: get(hObject,'Value') returns position of slider  % get(hObject,'Min') and get(hObject,'Max') to determine range of slider  % --- Executes during object creation, after setting all properties.  function gajiSlider\_CreateFcn(hObject, eventdata, handles)  % hObject handle to wawancaraText (see GCBO)  % eventdata reserved - to be defined in a future version of MATLAB  % handles empty - handles not created until after all CreateFcns called  % Hint: slider controls usually have a light gray background.  if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))  set(hObject,'BackgroundColor',[.9 .9 .9]);  end  % --- Executes on button press in resultButton.  function resultButton\_Callback(hObject, eventdata, handles)  % hObject handle to resultButton (see GCBO)  % eventdata reserved - to be defined in a future version of MATLAB |
| **Lanjutan Program 2.2 Keseluruhan GUI Matlab** |

|  |
| --- |
| % handles structure with handles and user data (see GUIDATA)  a = readfis('Penerimaan\_Beasiswa');  ipk = get(handles.ipkText, 'string');  Ipk = str2double(ipk);  wawancara = get(handles.wawancaraText, 'string');  Wawancara = str2double(wawancara);  gaji = get(handles.gajiText, 'string');  Gaji = str2double(gaji);  out = evalfis([Ipk Wawancara Gaji],a);  if(out >= 0 && out <= 4.5)  hasil = "Tidak Diterima";  elseif(out >= 4.5 && out <= 6.5)  hasil = "Dipertimbangkan";  else  hasil = "Diterima";  end  set(handles.result, 'String', hasil);  function result\_Callback(hObject, eventdata, handles)  % hObject handle to result (see GCBO)  % eventdata reserved - to be defined in a future version of MATLAB  % handles structure with handles and user data (see GUIDATA)  % Hints: get(hObject,'String') returns contents of result as text  % str2double(get(hObject,'String')) returns contents of result as a double  % --- Executes during object creation, after setting all properties.  function result\_CreateFcn(hObject, eventdata, handles)  % hObject handle to result (see GCBO)  % eventdata reserved - to be defined in a future version of MATLAB  % handles empty - handles not created until after all CreateFcns called  % Hint: edit controls usually have a white background on Windows.  % See ISPC and COMPUTER.  if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))  set(hObject,'BackgroundColor','white');  end |
| **Lanjutan Program 2.2 Keseluruhan GUI Matlab** |

Meski begitu, untuk memperjelas bagian-bagian penting dalam listing program, berikut kami berikan beberapa listing secara terpisah.

|  |
| --- |
| % --- Executes on slider movement.  function ipkSlider\_Callback(hObject, eventdata, handles)  sliderValue = get(hObject, 'Value');  % Mengupdate text samping slider  set(handles.ipkText, 'String', num2str(sliderValue)); |
| **Listing Program 2.3 Slider ‘IPK’** |

|  |
| --- |
| % --- Executes on slider movement.  function wawancaraSlider\_Callback(hObject, eventdata, handles)  sliderValue = get(hObject, 'Value');  % Mengupdate text samping slider  set(handles.wawancaraText, 'String', num2str(sliderValue)); |
| **Listing Program 2.4 Slider ‘Wawancara’** |

|  |
| --- |
| % --- Executes on slider movement.  function gajiSlider\_Callback(hObject, eventdata, handles)  sliderValue = get(hObject, 'Value');  % Mengupdate text samping slider  set(handles.gajiText, 'String', num2str(sliderValue)); |
| **Listing Program 2.5 Listing Slider ‘Gaji Orangtua** |

|  |
| --- |
| % --- Executes on button press in resultButton.  function resultButton\_Callback(hObject, eventdata, handles)  a = readfis('Penerimaan\_Beasiswa');  ipk = get(handles.ipkText, 'string');  Ipk = str2double(ipk);  wawancara = get(handles.wawancaraText, 'string');  Wawancara = str2double(wawancara);  gaji = get(handles.gajiText, 'string');  Gaji = str2double(gaji);  out = evalfis([Ipk Wawancara Gaji],a);  if(out >= 0 && out <= 4.5)  hasil = "Tidak Diterima";  elseif(out >= 4.5 && out <= 6.5) |
| **Listing Program 2.6 Tombol ‘Hitung’** |

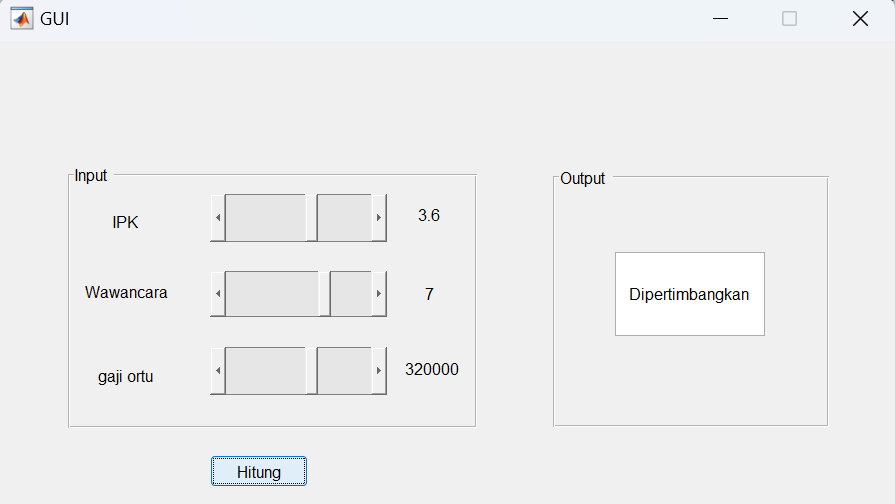
|  |
| --- |
| hasil = "Dipertimbangkan";  else  hasil = "Diterima";  end  set(handles.result, 'String', hasil); |

**Lanjutan Listing Program 2.6 Tombol ‘Hitung’**

1. **Tampilan Program**

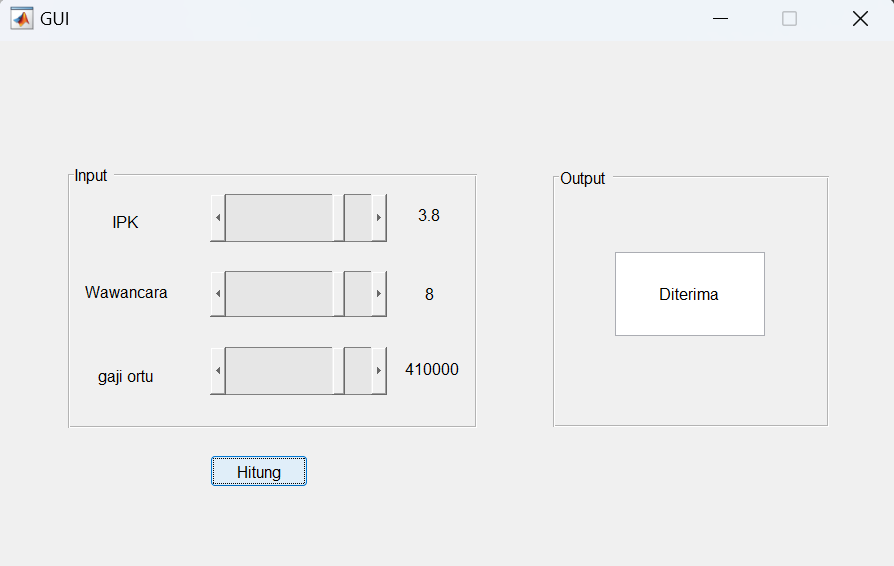
Halaman ini adalah halaman utama dari program yang dibuat. Terdapat slider yang berfungsi memberikan input angka, meliputi IPK, skor wawancara, dan gaji orangtua. Alasan penggunaan slider sebagai komponen pemberi input adalah agar input yang dimasukkan tidak lebih atau kurang dari batas yang sudah ditentukan. Tombol ‘Hitung’ akan melakukan proses penyeleksian sesuai metode logika *fuzzy*. Hasil seleksi akan muncul di palet ‘Output’ dan akan memberikan keterangan apakah mahasiswa tersebut Diterima, Dipertimbangkan, atau Tidak Diterima di program beasiswa. Berikut contoh tampilan program beserta contoh kasusnya.

1. Seorang mahasiswa bernama Ahmad dengan IPK 3,6 mendaftar program beasiswa dan dinyatakan lolos seleksi berkas. Hasil skor wawancaranya adalah 7 dari 10. Orangtuanya bekerja sebagai wirausahawan dan berpenghasilan Rp3.200.000,00 / bulan. Apakah dia akan diterima di program beasiswa?



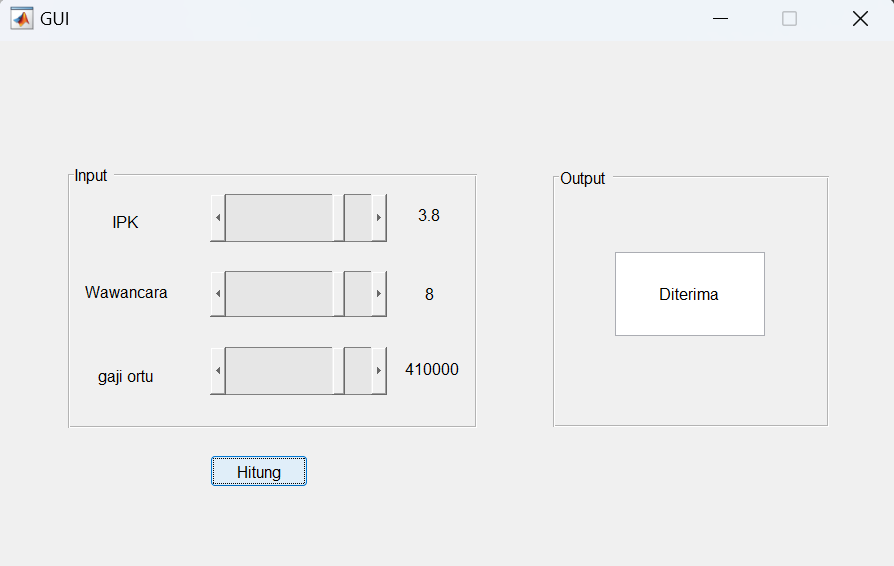
**Gambar 2.5 Tampilan Program untuk Kasus 1**

1. Seorang mahasiswi bernama Mawar yang juga mendaftar program beasiswa ikut lolos seleksi berkas. Ia memperoleh skor 8 untuk wawancara. Setelah dilihat kembali pada berkasnya, ia pun memperoleh IPK 3,8 dan memiliki orangtua dengan gaji Rp4.100.000,00 /bulan. Bagaimana hasil penyeleksian untuk Mawar?



**Gambar 2.6 Tampilan Program untuk Kasus 2**

1. Seorang mahasiswi bernama Dina mendaftar program beasiswa ikut lolos seleksi berkas. Ia memperoleh skor 5 untuk wawancara. Setelah dilihat kembali pada berkasnya, ia pun memperoleh IPK 3,4 dan memiliki orangtua dengan gaji Rp4.100.000,00 /bulan. Bagaimana hasil penyeleksian untuk Dina?



**Gambar 2.7 Tampilan Program untuk Kasus 3**

# **BAB III**

# **JADWAL PENGERJAAN DAN PEMBAGIAN TUGAS**

## **3.1 Jadwal Pengerjaan**

**Tabel 3. 1 Jadwal pengerjaan**

|  |  |  |  |
| --- | --- | --- | --- |
| **NO** | **Kegiatan** | **Juni 2023** | |
| **Minggu** | |
| **1** | **2** |
| 1. | Penentuan Ide |  |  |
| 2. | Pembuatan Program |  |  |
| 3. | Pembuatan Laporan |  |  |

## **3.2 Pembagian Tugas**

**Tabel 3. 2 Pembagian Tugas**

|  |  |  |
| --- | --- | --- |
| **NO** | **Kegiatan** | **Penanggung Jawab** |
| 1. | Penentuan Ide | Seva, Nisa |
| 2. | Pembuatan Program | Seva, Nisa |
| 3. | Pembuatan Laporan | Seva, Nisa |

# **BAB IV**

# **KESIMPULAN DAN SARAN**

## **4.1 Kesimpulan**

Sistem ini dibuat untuk membantu pengambilan keputusan dalam seleksi penerimaan beasiswa agar target mahasiswa yang menerima beasiswa tepat sasaran. Perhitungan yang kami lakukan dengan menggunakan *software* MATLAB dengan metode logika fuzzy memberikan hasil yang cepat dan akurat daripada dengan perhitungan manual yang membutuhkan waktu yang lama.

## **4.2 Saran**

Hasil perhitungan saat ini sudah cukup baik dan dapat membantu dalam mengambil keputusan seleksi penerimaan beasiswa. Akan tetapi, untuk pengembangan di masa depan, dapat dilakukan penyesuaian dengan persyaratan dari beasiswa terkait sehingga hasil yang didapatkan akurat dan benar-benar membantu dalam pengambilan keputusan.

**DAFTAR PUSTAKA**

GeeksforGeeks. (2023). Fuzzy Logic Introduction. Diakses pada 13 Juni 2023, dari https://www.geeksforgeeks.org/fuzzy-logic-introduction.