**LAPORAN PROYEK AKHIR**

**SISTEM CERDAS DAN PENDUKUNG KEPUTUSAN**

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

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

Description automatically generated**

**DISUSUN OLEH :**

**NIM : 123220020**

**123220052**

**NAMA : DEA REIGINA**

**NAUFAL RAFID MUHAMMAD**

**FADDILA**

**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**

**2024**

# **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 |

**Tabel 1. 1 Pendapatan**

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 |

**Tabel 1. 2 Jumlah tanggungan**

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 |

**Tabel 1. 3 Kondisi rumah**

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 |

**Tabel 1. 4 Hasil**

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='seleksi\_bansos'  Type='mamdani'  Version=2.0  NumInputs=3  NumOutputs=1  NumRules=27  AndMethod='min'  OrMethod='max'  ImpMethod='min'  AggMethod='max'  DefuzzMethod='centroid'  [Input1]  Name='pendapatan'  Range=[0 5000000]  NumMFs=3  MF1='rendah':'trapmf',[-555600 -208300 500000 2700000]  MF2='sedang':'trimf',[2000000 3000000 4000000]  MF3='tinggi':'trapmf',[3500000 4750000 5000000 5000000]  [Input2]  Name='jumlah\_tanggungan'  Range=[0 10]  NumMFs=3  MF1='sedikit':'trapmf',[0 0 0.5 2]  MF2='sedang':'trimf',[1 3 5]  MF3='banyak':'trapmf',[4 6 10 10]  [Input3]  Name='kondisi\_rumah'  Range=[0 10]  NumMFs=3  MF1='buruk':'trapmf',[-1.111 -0.4167 2 4.5]  MF2='cukup':'trimf',[3 5 7]  MF3='baik':'trapmf',[5.5 8 10.42 13.75]  [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 3 1, 3 (1) : 1  1 3 2, 3 (1) : 1  1 3 3, 3 (1) : 1  1 2 1, 3 (1) : 1  1 2 2, 3 (1) : 1  1 2 3, 2 (1) : 1  1 1 1, 2 (1) : 1  1 1 2, 1 (1) : 1  1 1 3, 1 (1) : 1  2 3 1, 3 (1) : 1  2 3 2, 3 (1) : 1  2 3 3, 2 (1) : 1  2 2 1, 2 (1) : 1  2 2 2, 1 (1) : 1  2 2 3, 1 (1) : 1  2 1 1, 2 (1) : 1  2 1 2, 1 (1) : 1  2 1 3, 1 (1) : 1  3 3 1, 2 (1) : 1  3 3 2, 1 (1) : 1  3 3 3, 1 (1) : 1  3 2 1, 2 (1) : 1  3 2 2, 1 (1) : 1  3 2 3, 1 (1) : 1  3 1 1, 1 (1) : 1  3 1 2, 1 (1) : 1  3 1 3, 1 (1) : 1 |

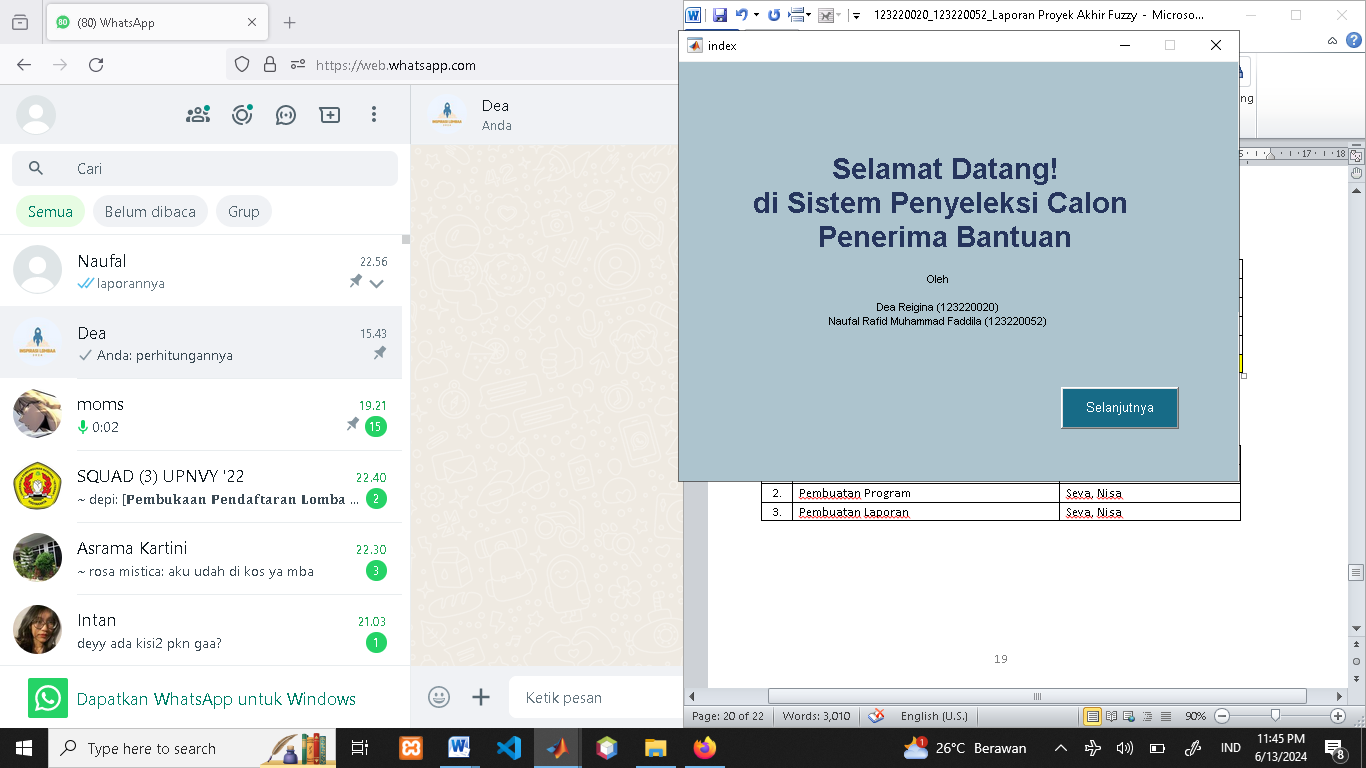
**Program 1 Fuzzy ‘seleksi\_bansos.fis’**

|  |
| --- |
| function varargout = index(varargin)  % INDEX MATLAB code for index.fig  % INDEX, by itself, creates a new INDEX or raises the existing  % singleton\*.  %  % H = INDEX returns the handle to a new INDEX or the handle to  % the existing singleton\*.  %  % INDEX('CALLBACK',hObject,eventData,handles,...) calls the local  % function named CALLBACK in INDEX.M with the given input arguments.  %  % INDEX('Property','Value',...) creates a new INDEX or raises the  % existing singleton\*. Starting from the left, property value pairs are  % applied to the GUI before index\_OpeningFcn gets called. An  % unrecognized property name or invalid value makes property application  % stop. All inputs are passed to index\_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 index  % Last Modified by GUIDE v2.5 13-Jun-2024 19:12:16  % Begin initialization code - DO NOT EDIT  gui\_Singleton = 1;  gui\_State = struct('gui\_Name', mfilename, ...  'gui\_Singleton', gui\_Singleton, ...  'gui\_OpeningFcn', @index\_OpeningFcn, ...  'gui\_OutputFcn', @index\_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 index is made visible.  function index\_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 index (see VARARGIN)  % Choose default command line output for index  handles.output = hObject;  % Update handles structure  guidata(hObject, handles);  % UIWAIT makes index wait for user response (see UIRESUME)  % uiwait(handles.figure1);  % --- Outputs from this function are returned to the command line.  function varargout = index\_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 button press in selanjutnya.  function selanjutnya\_Callback(hObject, eventdata, handles)  % hObject handle to selanjutnya (see GCBO)  % eventdata reserved - to be defined in a future version of MATLAB  % handles structure with handles and user data (see GUIDATA)  inputan;  close('index'); |
| **Program 2 GUI Mathlab Program ‘index.m’** |

|  |
| --- |
| function varargout = inputan(varargin)  % inputan MATLAB code for inputan.fig  % inputan, by itself, creates a new inputan or raises the existing  % singleton\*.  %  % H = inputan returns the handle to a new inputan or the handle to  % the existing singleton\*.  %  % inputan('CALLBACK',hObject,eventData,handles,...) calls the local  % function named CALLBACK in inputan.M with the given input arguments.  %  % inputan('Property','Value',...) creates a new inputan or raises the  % existing singleton\*. Starting from the left, property value pairs are  % applied to the inputan before inputan\_OpeningFcn gets called. An  % unrecognized property name or invalid value makes property application  % stop. All inputs are passed to inputan\_OpeningFcn via varargin.  %  % \*See inputan Options on GUIDE's Tools menu. Choose "inputan allows only one  % instance to run (singleton)".  %  % See also: GUIDE, GUIDATA, GUIHANDLES  % Edit the above text to modify the response to help inputan  % Last Modified by GUIDE v2.5 13-Jun-2024 21:47:27  % Begin initialization code - DO NOT EDIT  gui\_Singleton = 1;  gui\_State = struct('gui\_Name', mfilename, ...  'gui\_Singleton', gui\_Singleton, ...  'gui\_OpeningFcn', @inputan\_OpeningFcn, ...  'gui\_OutputFcn', @inputan\_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 inputan is made visible.  function inputan\_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 inputan (see VARARGIN)  % Choose default command line output for inputan  handles.output = hObject;  % Update handles structure  guidata(hObject, handles);  % UIWAIT makes inputan wait for user response (see UIRESUME)  % uiwait(handles.figure1);  % --- Outputs from this function are returned to the command line.  function varargout = inputan\_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 pendapatanSlider\_Callback(hObject, eventdata, handles)  % hObject handle to pendapatanSlider (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.pendapatanText, '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 pendapatanSlider\_CreateFcn(hObject, eventdata, handles)  % hObject handle to pendapatanSlider (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 tanggunganSlider\_Callback(hObject, eventdata, handles)  % hObject handle to tanggunganSlider (see GCBO)  % eventdata reserved - to be defined in a future version of MATLAB  % handles structure with handles and user data (see GUIDATA)  sliderValue = round(get(hObject, 'Value'));  % Mengupdate text samping slider  set(handles.tanggunganTextt, 'String', num2str(sliderValue));  set(hObject, 'Value', sliderValue); % Ensure the slider value is an integer  % 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 tanggunganSlider\_CreateFcn(hObject, eventdata, handles)  % hObject handle to tanggunganSlider (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 rumahSlider\_Callback(hObject, eventdata, handles)  % hObject handle to tanggunganText (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.rumahText, '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 rumahText\_CreateFcn(hObject, eventdata, handles)  % hObject handle to tanggunganText (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  function pendapatanText\_Callback(hObject, eventdata, handles)  % hObject handle to pendapatanText (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 pendapatanText as text  % str2double(get(hObject,'String')) returns contents of pendapatanText as a double  textValue = str2double(get(hObject, 'String'));  if textValue < 0  textValue = 0;  elseif textValue > 5000000  textValue = 5000000;  end  set(handles.pendapatanText, 'String', num2str(textValue));  set(handles.pendapatanSlider, 'Value', textValue);  % --- Executes during object creation, after setting all properties.  function pendapatanText\_CreateFcn(hObject, eventdata, handles)  % hObject handle to pendapatanText (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  function tanggunganTextt\_Callback(hObject, eventdata, handles)  % hObject handle to tanggunganTextt (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 tanggunganTextt as text  % str2double(get(hObject,'String')) returns contents of tanggunganTextt as a double  textValue = str2double(get(hObject, 'String'));  if textValue < 0  textValue = 0;  elseif textValue > 10  textValue = 10;  end  set(handles.tanggunganTextt, 'String', num2str(textValue));  set(handles.tanggunganSlider, 'Value', textValue);  % --- Executes during object creation, after setting all properties.  function tanggunganTextt\_CreateFcn(hObject, eventdata, handles)  % hObject handle to tanggunganTextt (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  function rumahText\_Callback(hObject, eventdata, handles)  % hObject handle to rumahText (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 rumahText as text  % str2double(get(hObject,'String')) returns contents of rumahText as a double  textValue = str2double(get(hObject, 'String'));  if textValue < 0  textValue = 0;  elseif textValue > 10  textValue = 10;  end  set(handles.rumahText, 'String', num2str(textValue));  set(handles.rumahSlider, 'Value', textValue);  % --- 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  % handles structure with handles and user data (see GUIDATA)  global hasill;  % Get and validate jumlah tanggungan  jumlah\_tanggungan = get(handles.tanggunganTextt, 'string');  jumlah\_tanggungan = str2double(jumlah\_tanggungan);  if isnan(jumlah\_tanggungan) || mod(jumlah\_tanggungan, 1) ~= 0 || jumlah\_tanggungan < 0 || jumlah\_tanggungan > 10  errordlg('Jumlah tanggungan harus angka bulat tanpa koma dan berada dalam rentang 0 hingga 10, silahkan inputkan ulang di ', 'Input Error');  return;  end  a = readfis('seleksi\_bansos');  pendapatan = get(handles.pendapatanText, 'string');  pendapatan = str2double(pendapatan);  jumlah\_tanggungan = get(handles.tanggunganTextt, 'string');  jumlah\_tanggungan = str2double(jumlah\_tanggungan);  kondisi\_rumah = get(handles.rumahText, 'string');  kondisi\_rumah = str2double(kondisi\_rumah);  out = evalfis([pendapatan jumlah\_tanggungan kondisi\_rumah],a);  if(out >= 0 && out <= 4.5)  hasill = "Tidak diterima";  elseif(out >= 4.5 && out <= 6.5)  hasill = "Dipertimbangkan";  else  hasill = "Diterima";  end  hasil;  close('inputan');  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  % --- Executes on mouse press over figure background.  function figure1\_ButtonDownFcn(hObject, eventdata, handles)  % hObject handle to figure1 (see GCBO)  % eventdata reserved - to be defined in a future version of MATLAB  % handles structure with handles and user data (see GUIDATA)  % --- Executes during object creation, after setting all properties.  function rumahSlider\_CreateFcn(hObject, eventdata, handles)  % hObject handle to rumahSlider (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 btnreset.  function btnreset\_Callback(hObject, eventdata, handles)  % hObject handle to btnreset (see GCBO)  % eventdata reserved - to be defined in a future version of MATLAB  % handles structure with handles and user data (see GUIDATA)  set(handles.pendapatanText, 'String', '');  set(handles.tanggunganTextt, 'String', '');  set(handles.rumahText, 'String', '');  guidata(hObject, handles);    % --- Executes on button press in btnback.  function btnback\_Callback(hObject, eventdata, handles)  % hObject handle to btnback (see GCBO)  % eventdata reserved - to be defined in a future version of MATLAB  % handles structure with handles and user data (see GUIDATA)  index;  close('inputan');  % --- Executes during object creation, after setting all properties.  function edit5\_CreateFcn(hObject, eventdata, handles)  % hObject handle to rumahText (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 |
| **Program 3 Program GUI mathlab ‘inputan.m’** |

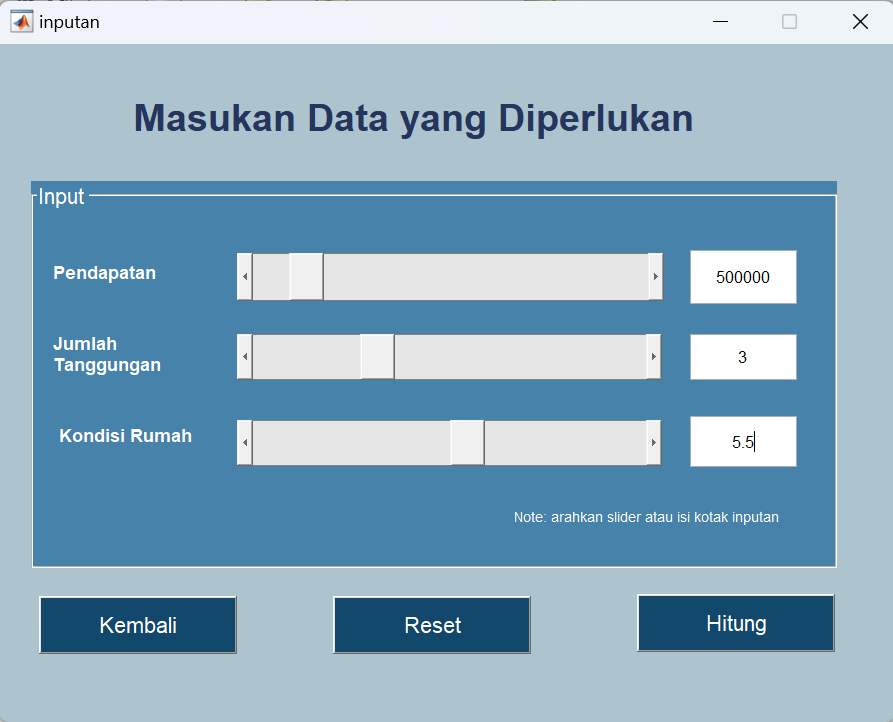
|  |
| --- |
| function varargout = hasil(varargin)  % HASIL MATLAB code for hasil.fig  % HASIL, by itself, creates a new HASIL or raises the existing  % singleton\*.  %  % H = HASIL returns the handle to a new HASIL or the handle to  % the existing singleton\*.  %  % HASIL('CALLBACK',hObject,eventData,handles,...) calls the local  % function named CALLBACK in HASIL.M with the given input arguments.  %  % HASIL('Property','Value',...) creates a new HASIL or raises the  % existing singleton\*. Starting from the left, property value pairs are  % applied to the GUI before hasil\_OpeningFcn gets called. An  % unrecognized property name or invalid value makes property application  % stop. All inputs are passed to hasil\_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 hasil  % Last Modified by GUIDE v2.5 13-Jun-2024 19:32:06  % Begin initialization code - DO NOT EDIT  gui\_Singleton = 1;  gui\_State = struct('gui\_Name', mfilename, ...  'gui\_Singleton', gui\_Singleton, ...  'gui\_OpeningFcn', @hasil\_OpeningFcn, ...  'gui\_OutputFcn', @hasil\_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 hasil is made visible.  function hasil\_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 hasil (see VARARGIN)  % Choose default command line output for hasil  handles.output = hObject;  global hasill;  handles.hasill = hasill; % Simpan hasil dalam handles  set(handles.result, 'String', hasill); % Perbarui field result dengan hasil  disp(['Nilai hasill saat ini: ', hasill]);  guidata(hObject, handles); % Update handles  % UIWAIT makes hasil wait for user response (see UIRESUME)  % uiwait(handles.figure1);  % --- Outputs from this function are returned to the command line.  function varargout = hasil\_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 button press in btnkembali.  function btnkembali\_Callback(hObject, eventdata, handles)  % hObject handle to btnkembali (see GCBO)  % eventdata reserved - to be defined in a future version of MATLAB  % handles structure with handles and user data (see GUIDATA)  inputan;  close('hasil');  % --- Executes on button press in btnselesai.  function btnselesai\_Callback(hObject, eventdata, handles)  % hObject handle to btnselesai (see GCBO)  % eventdata reserved - to be defined in a future version of MATLAB  % handles structure with handles and user data (see GUIDATA)  close('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  global hasill;  disp(['Nilai hasill saat ini di result\_Callback: ', hasill]);  set(handles.result, 'String', hasill); % Pastikan nilai yang diharapkan diperbarui di GUI  % --- 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 |
| **Program 4 GUI Mathlab Program ‘hasil.m’** |

1. **Tampilan Program**
2. Halaman beranda yang menyambut pengguna dan mengarahkan kehalaman selanjutnya.



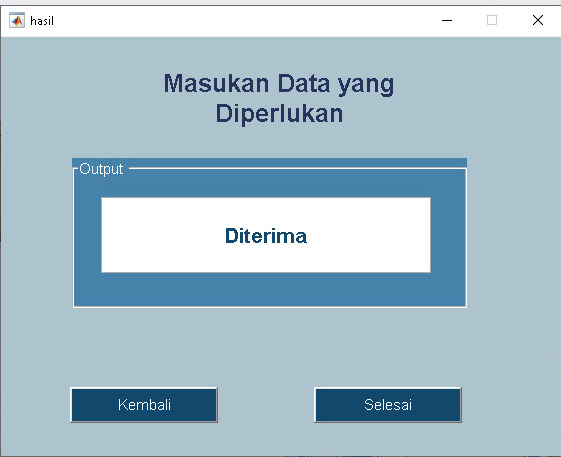
**Gambar 1 Tampilan program untuk halaman beranda**

1. Seorang warga bernama Samsudin memiliki gaji atau penghasilan sebesar 500.000 /bulan dan memiliki tanggungan 3 orang, berdasarkan survei Lokasi keadaaan rumahnya cukup walaupun tidak terlalu baik. Berdasarkan pengamatan tim survei memberikan nilai 5.5 untuk kondisi rumahnya. Bagaimana hasil seleksi, apakah Samsudin berhak untuk menerima bantuan sosial dari pemerintah?



**Gambar 2 Tampilan program untuk memasukkan data**

1. Berikut halaman yang menampilkan hasil dari perhitungan fuzzy, hasilnya akan mengeluarkan output diterima, dipertimbangkan atau ditolak.



**Gambar 3 Tampilan Program untuk bagian input data**

# **BAB III**

# **JADWAL PENGERJAAN DAN PEMBAGIAN TUGAS**

## **3.1 Jadwal Pengerjaan**

**Tabel 2. 1 Jadwal pengerjaan**

|  |  |  |  |
| --- | --- | --- | --- |
| **NO** | **Kegiatan** | **Juni 2024** | |
| **Minggu** | |
| **1** | **2** |
| 1. | Penentuan Ide |  |  |
| 2. | Pembuatan Program |  |  |
| 3. | Pembuatan Laporan |  |  |

## **3.2 Pembagian Tugas**

**Tabel 2. 2 Pembagian Tugas**

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

# **BAB IV**

# **KESIMPULAN DAN SARAN**

## **4.1 Kesimpulan**

Sistem ini dirancang untuk membantu pengambilan keputusan dalam seleksi penerima bantuan sosial agar tepat sasaran. Perhitungan yang kami lakukan menggunakan software MATLAB dengan metode logika fuzzy memberikan hasil yang lebih cepat dan akurat dibandingkan dengan perhitungan manual yang memakan waktu lama dan diragukan objektivitasnya.

## **4.2 Saran**

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

**DAFTAR PUSTAKA**

GeeksforGeeks. (2023). Fuzzy Logic Introduction. Diakses pada 11 Juni 2024, dari https://www.geeksforgeeks.org/fuzzy-logic-introduction.