

## PENUTUP

### 8.1. Kesimpulan

Berdasarkan penelitian yang telah dilakukan, diperoleh bahwarute yang optimal dapat mengurangi waktu tempuh, jarak tempuh, dan biaya dalam proses pengiriman. Adapun peneliti mendapatkan beberapa kesimpulan seiring dengan berjalannya penelitian, maupun ketika penelitian sudah selesai dilakukan, antara lain:

1. Biaya yang dihasilkan oleh perhitungan menggunakan metode *saving matrix-nearest insert*, *saving matrix-nearest neighbor*, *sweep method-nearest insert*, dan *sweep method-nearest neighbor* berturut-turut adalah sebesar Rp55,301, Rp55,295, Rp85,050 dan Rp85,050, dimana biaya awal tanpa menggunakan metode adalah sebesar Rp137,237. Sehingga penghematan yang didapatkan berturut-turut adalah sebesar 60%, 60%, 38% dan 38%. Hal ini memenuhi salah satu *critical success factor* penelitian ini yaitu memangkas biaya awal
2. Perhitungan menggunakan metode *saving matrix-nearest neighbour* untuk tanggal 7, 14, 16 dan 28 Maret menghasilkan total jarak tempuh sebesar 175 km, perhitungan menggunakan *saving matrix-nearest insert* menghasilkan total jarak sebesar 180,4, perhitungan menggunakan *metode sweep method-nearest insert* menghasilkan jarak total sebesar 236,2, dan metode *sweep method-nearest neighbour* menghasilkan jarak total sebesar 236,2 dimana pada perhitungan awal didapatkan total jarak tempuh sebesar 280,4 km. Sehingga penghematan yang didapatkan berturut-turut adalah sebesar 38%, 35%, 16% dan 16%. Hal ini memenuhi salah satu *critical success factor* penelitian ini yaitu memangkas jarak tempuh awal. Dari penghematan yang didapatkan, ditarik kesimpulan bahwa metode *saving matrix-nearest neighbour* merupakan metode terbaik untuk mengurangi jarak tempuh pengiriman
3. Simulasi dengan program optimasi rute pada tanggal 29 dan 30 Juli menghasilkan penghematan waktu sebesar 12% dan biaya sebesar 2%, dari hasil ini diperoleh kesimpulan bahwa program berjalan dengan baik dan dapat diterapkan menjadi bagian dari sistem distribusi di toko besi XYZ

## 8.2. Saran

Demi mengembangkan sistem yang telah dirancang oleh peneliti kedepannya, peneliti memiliki beberapa saran agar penelitian berikutnya dapat menjadi lebih baik. Beberapa saran tersebut adalah sebagai berikut.

1. Kendaraan pengiriman di toko besi XYZ tidak terpasang alat odometer sehingga kalkulasi biaya bensin tidak akurat, sehingga dapat lebih baik apabila pada penelitian selanjutnya menggunakan odometer
2. Program optimasi rute dapat diintegrasikan dengan *google maps* sehingga *driver* dapat memvisualisasikan rute dengan lebih baik, dan melihat rute yang sedang macet
3. Perlu dilakukan adanya penelitian lebih lanjut terkait metode yang dapat memberikan minimasi waktu dan biaya lebih baik dari metode yang digunakan peneliti pada penelitian ini

## DAFTAR PUSTAKA

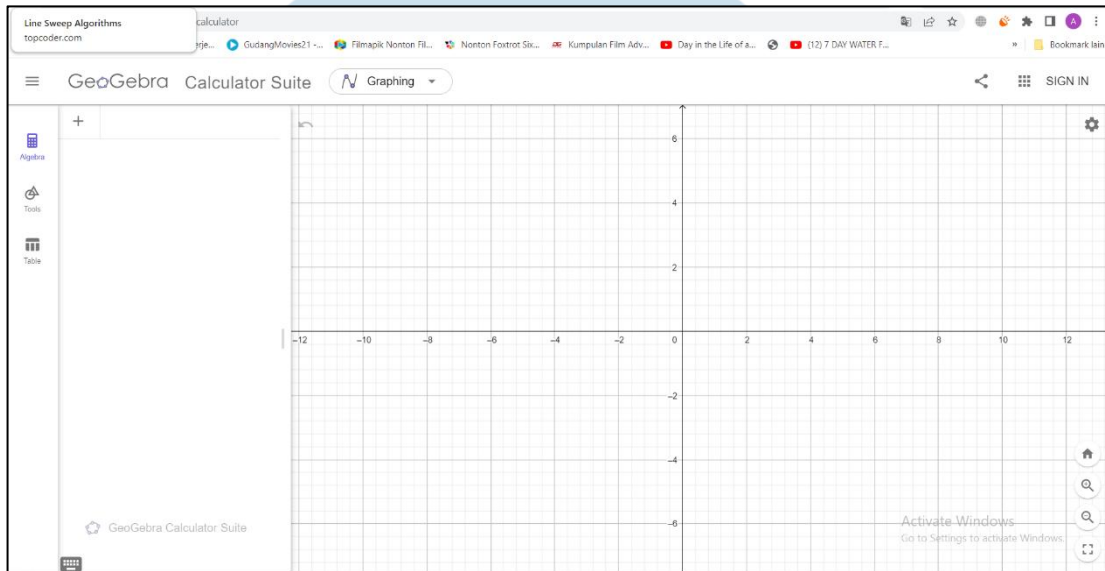
- Batubara, H. (2013). Penentuan harga pokok produksi berdasarkan metode full costing pada pembuatan etalase kaca dan alumunium di UD. Istana Alumunium Manado. *Jurnal EMBA: Jurnal Riset Ekonomi, Manajemen, Bisnis Dan Akuntansi*, 1(3).
- Leksono, A., & Sarwadi, S. (2009). *Algoritma Ant Colony Optimization (ACO) Untuk Menyelesaikan Traveling Salesman Problem (TSP)*. Mathematics and Natural science.
- Lubis, A. N. (2004). Peranan saluran distribusi dalam pemasaran produk dan jasa. *FE Jurusan Manajemen Universitas Sumatera Utara, e-USU Repository Universitas Sumatera Utara*.
- Nasution, H. M. N. (1996). *Manajemen Transportasi Kota*. Ghalia Indonesia.
- Pattiasina, T. J., Setyoadi, E. T., & Wijayanto, D. (2018). Saving matrix method for efficient distribution route based on google maps API. *Journal of Telecommunication, Electronic and Computer Engineering*, 10(2–3), 183–188.
- Pujawan, I. N., & Mahendrawati, E. R. (2010). *Supply Chain Management (Terbitan Kedua)*. Guna Widya.
- Salipadang, J. C. (2011). *analisis sistem pembangkitan sampah kota makassar dengan metode penyelesaian vehicle routing problem (vrp)*.
- Toth, P., & Vigo, D. (2002). *The vehicle routing problem*. SIAM.

## LAMPIRAN

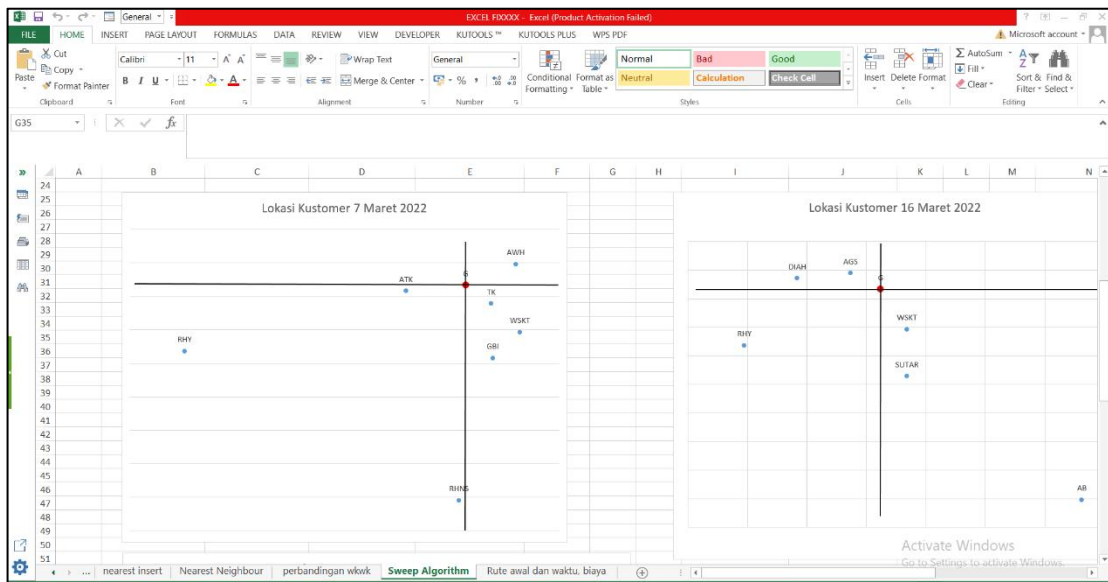
### Lampiran 1. Langkah-Langkah Mendapatkan Koordinat Kartesius

Berikut adalah langkah-langkah penentuan koordinat polar dari tiap titik kustomer menggunakan bantuan website online <https://www.geogebra.org/calculator>.

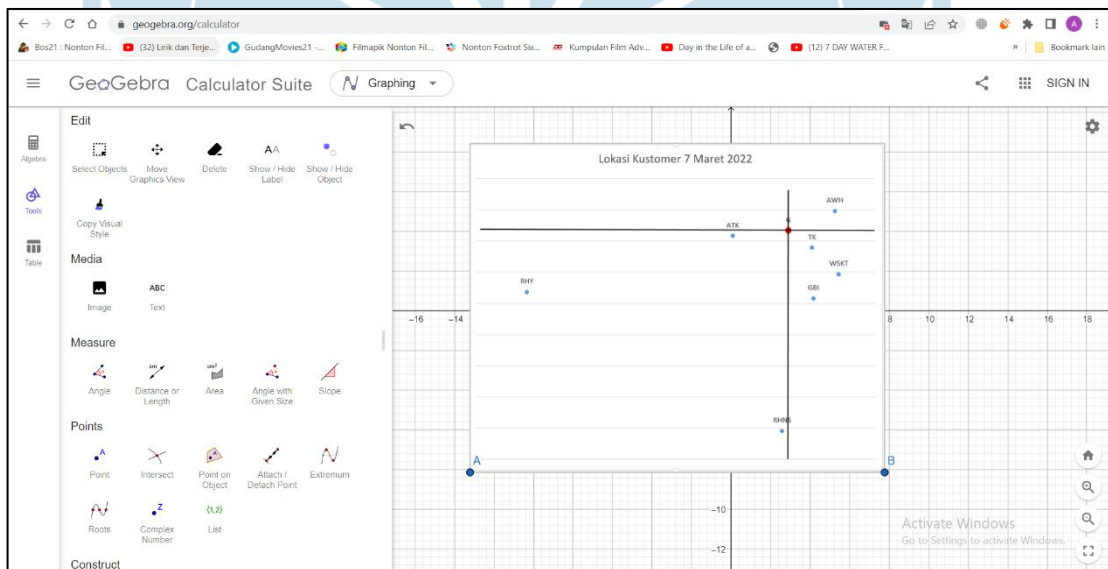
Langkah 1 : Membuka domain website <https://www.geogebra.org/calculator>



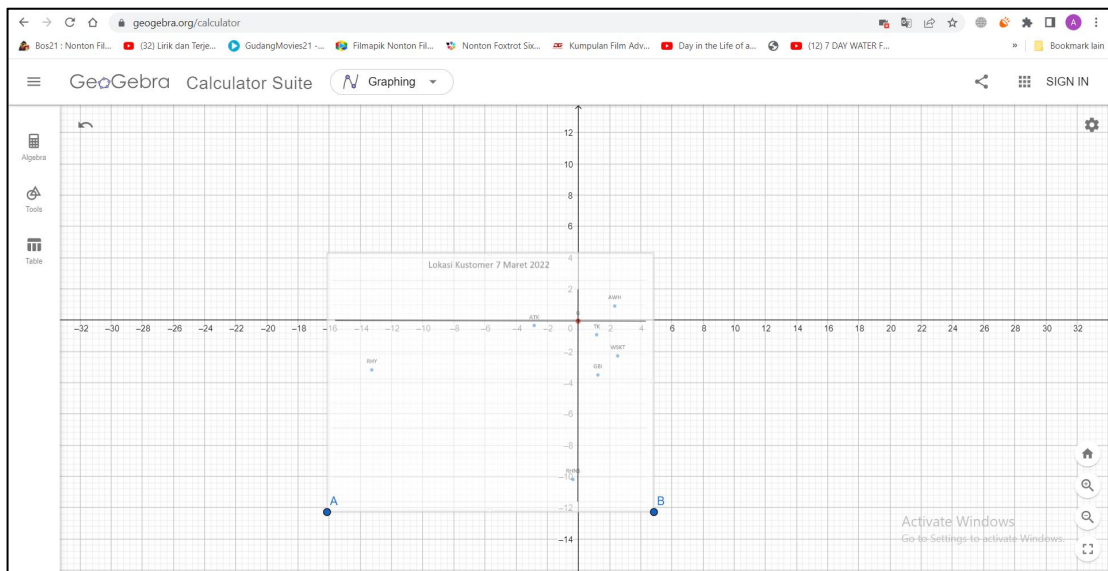
Langkah 2 : Membuat titik lokasi kustomer sesuai dengan koordinat UTM, dengan menggunakan *scatter plot chart*



Langkah 3 : Mengimpor gambar *scatter plot chart* yang telah dibuat dengan *Microsoft Excel* ke aplikasi *Geogebra*



Langkah 4 : Memposisikan gambar agar titik depot sebagai 0,0 di koordinat kartesius, kemudian mengatur *transparency* gambar agar terlihat tembus pandang, sehingga dapat dijiplak titik-titik koordinatnya



Langkah 5 : Pilih opsi *point* pada aplikasi *Geogebra* untuk menandai titik-titik kustomer sesuai gambar.



Langkah 6 : Mengubah koordinat tiap titik menjadi koordinat polar dengan cara klik kanan tiap titik – klik setting – *Algebra* – *Polar Coordinates*

Lampiran 2. *Coding Menu Utama*

```

Private Sub Edit_Data_Click()
UserForm2.Show 1
End Sub

Private Sub Exit_Button_Click()
Dim answer As Integer
answer = MsgBox("Anda yakin ingin keluar?", vbYesNo + vbQuestion, "Exit")
If answer = vbYes Then
ThisWorkbook.Save
Application.Quit
Else
End If
End Sub

Private Sub Tidak_Terlihat()
Application.Visible = False
End Sub

Private Sub UserForm_Click()

End Sub

```

### Lampiran 3. Coding Form Input Bagian 1

```

Private Sub ComboBox1_Change()

End Sub

Private Sub CommandButton1_Click()
Dim Baris terakhir As Integer
If TextBox1 = "" Or TextBox2 = "" Or TextBox3 = "" Or TextBox4 = "" Or ComboBox1 = "" Or TextBox5 = "" Then
MsgBox "Pastikan Semua Data Terisi"
Exit Sub
End If
Set Aktif_Sheet = Worksheets("Rekap Data")
Baris terakhir = Aktif_Sheet.Cells(Rows.Count, 2).End(xlUp).Row + 1
Aktif_Sheet.Cells(Baris_terakhir, 1) = TextBox1
Aktif_Sheet.Cells(Baris_terakhir, 2) = TextBox5
Aktif_Sheet.Cells(Baris_terakhir, 3) = TextBox2
Aktif_Sheet.Cells(Baris_terakhir, 4) = ComboBox1
Aktif_Sheet.Cells(Baris_terakhir, 6) = TextBox3
Aktif_Sheet.Cells(Baris_terakhir, 7) = TextBox4

With Sheet4

rowbb = Sheet4.Cells(Rows.Count, "D").End(xlUp).Row
'WorksheetFunction.VLookup(rngJ2.Offset(, -1), Sheets("Sheet1").Range("A:H"), 8, False)
bb = IIf(WorksheetFunction.IsNA(Application.VLookup(ComboBox1, Sheets("berat barang").Range("C:D"), 2, False)), 0, Application.VLookup(ComboBox1, Sheets
demand = bb * Val(TextBox2)

End With
Aktif_Sheet.Cells(Baris_terakhir, 5) = demand

Bersih

End Sub

Private Sub Label6_Click()

End Sub

Private Sub TextBox1_Change()

```

Activate Windows  
Go to Settings to activate Windows.

### Lampiran 4. Coding Form Input Bagian 2



```

Private Sub UserForm_Activate()
With ComboBox1
.AddItem "Semen"
.AddItem "Triplek 3/4"
.AddItem "Hollow 2X4"
.AddItem "Mill"
.AddItem "Asbes Gel. Besar 3 Meter"
.AddItem "Asbes Gel. Kecil 240 Cm"
.AddItem "Asbes Gel. Kecil 300 Cm"
.AddItem "Bata Ringan 7.5 Cm"
.AddItem "Bata Ringan 10 Cm"
.AddItem "Batako"
.AddItem "Bondek 4 Meter"
.AddItem "Bondek 6 Meter"
.AddItem "Galvalum Dek 5 Meter"
.AddItem "Galvalum Dek 6 Meter"
.AddItem "Kayu Reng"
.AddItem "Keramik 20X20"
.AddItem "Keramik 20X25"
.AddItem "Keramik 30X30"
.AddItem "Keramik 40X40"
.AddItem "Mortar"
.AddItem "Papan Gypsum Star"
.AddItem "Pipa 4""
.AddItem "Reng Baja Ringan"
.AddItem "Canal C 100 X 50 X 20 X 3,2"
.AddItem "Canal C 125 X 50 X 20 X 2,3 "
.AddItem "Canal C 150 X 50 X 20 X 2,3"

End With
End Sub

Private Sub Bersih()
TextBox1 = ""
TextBox2 = ""
TextBox3 = ""
TextBox4 = ""
ComboBox1 = ""
TextBox5 = ""
TextBox1.SetFocus
End Sub

```

Lampiran 5. Coding Matriks Jarak



```

Sub createMatrixReport(seet As Worksheet)
seet.Activate

With seet
    'menentukan max row -nya dari baris ke 4
    rmax = Sheet2.Cells(Rows.Count, "A").End(xlUp).Row
    jumlahrow = rmax - 3
    'rmax2 = (2 * rmax) + 5
    .Cells.Clear
    startrowtulis = 15

    'title header tabel
    .Cells(startrowtulis, "D") = "Nama"
    .Cells(startrowtulis, "E") = "X"
    .Cells(startrowtulis, "F") = "Y"
    .Cells(startrowtulis, "G") = "Demand"

    'Isi Tabel Nama , kooridnat X Y
    baca = 4
    tulis = startrowtulis
    For I = 1 To jumlahrow
        tulis = tulis + 1
        .Cells(tulis, "D") = Sheet2.Cells(baca, "A").Value
        .Cells(tulis, "E") = Sheet2.Cells(baca, "F").Value
        .Cells(tulis, "F") = Sheet2.Cells(baca, "G").Value
        .Cells(tulis, "G") = Sheet2.Cells(baca, "E").Value

        baca = baca + 1
    Next

    'Isi Tabel matrix , kolom nama

    'title header tabel
    baca = 4
    tulis = startrowtulis
    .Cells(startrowtulis, "H") = "Nama"
    For I = 1 To jumlahrow
        tulis = tulis + 1
        .Cells(tulis, "H") = Sheet2.Cells(baca, "A").Value
        baca = baca + 1
    Next
    baca = 4

```

Lampiran 6. Coding Pengelompokkan Bagian 1

```

Sub tuliskelompok()

Dim Lused_tujuan As Object
Dim Lused_kelompok As Object
Set Lused_tujuan = CreateObject("System.Collections.ArrayList")
Set Lused_kelompok = CreateObject("System.Collections.ArrayList")

Dim beratkendaraan As Object
Set beratkendaraan = CreateObject("System.Collections.ArrayList")
berat = "250;3000;3000"
berat1 = Split(berat, ";")
For I = LBound(berat1) To UBound(berat1)
    b = berat1(I)
    beratkendaraan.Add b
Next

With Sheet5

'Create title pembuatan kelompok
StartcolKelompok = colSavingMatrixStart + jumlahrow + 1
.Cells(startrow, StartcolKelompok + 1) = "Kapasitas"
.Cells(startrow, StartcolKelompok + 2) = "Kelompok"
.Cells(startrow, StartcolKelompok + 3) = "Nilai Saving"
.Cells(startrow, StartcolKelompok + 4) = "Tujuan"
.Cells(startrow, StartcolKelompok + 5) = "Demand"
.Cells(startrow, StartcolKelompok + 6) = "Kendaraan"

For I = 0 To beratkendaraan.Count - 1
    sisaisikendaraan = Val(beratkendaraan.Item(I))
    For J = 0 To LargestKelompok.Count - 1
        kel = LargestKelompok.Item(J)

        If kelblock = kel Then GoTo nextj

        If Not Lused_tujuan.Contains(LargestTujuan(J)) Then

            If Val(LargestDemand.Item(J)) <= Val(sisaisikendaraan) Then
                sisaisikendaraan = sisaisikendaraan - Val(LargestDemand.Item(J))

                Lused_tujuan.Add LargestTujuan.Item(J)
                Lused_kelompok.Add I + 1 ' LargestTujuan.Item(j)
            End If
        End If
    Next J
Next I
End With

```

## Lampiran 7. Coding Pengelompokan Bagian 2

```

If Not Lused_tujuan.contains(LargestTujuan(J)) Then

    If Val(LargestDemand.Item(J)) <= Val(sisaisikendaraan) Then
        sisaisikendaraan = sisaisikendaraan - Val(LargestDemand.Item(J))

        Lused_tujuan.Add LargestTujuan.Item(J)
        Lused_kelompok.Add I + 1 ' LargestTujuan.Item(j)

        startrow = startrow + 1
        .Cells(startrow, StartcolKelompok + 1) = beratkendaraan.Item(I)
        .Cells(startrow, StartcolKelompok + 2) = I + 1
        .Cells(startrow, StartcolKelompok + 3) = Largestnvalue.Item(J)
        .Cells(startrow, StartcolKelompok + 4) = LargestTujuan.Item(J) ' "Tujuan"
        .Cells(startrow, StartcolKelompok + 5) = LargestDemand.Item(J) ' "Demand"
        .Cells(startrow, StartcolKelompok + 6) = IIf(beratkendaraan.Item(I) = "250", "Motor", "Mobil" & I)

    Else

        kelblock = kel

    End If

End If

nextj:
    Next J
    kelblock = ""
    sisaisikendaraan = 0

Next I

End With

Dim x As OLEObject
Set x = Sheets("Saving Matrix").OLEObjects("cmdurut")
x.Left = Sheet5.Cells(15 - 4, StartcolKelompok + 1).Left

End Sub

```

Activate Windows  
Go to Settings to activate Windows.

## Lampiran 8. Coding Saving Matrix Bagian 1

```

Sub createSavingMatrixReport(seet As Worksheet)
seet.Activate

With seet
    'menentukan max row -nya dari baris ke 4
    rmax = Sheet2.Cells(Rows.Count, "A").End(xlUp).Row
    jumlahrow = rmax - 3
    'rmax2 = (2 * rmax) + 5
    .Cells.Clear

    startrowtulis = 15

    'Isi Tabel matrix , kolom nama
    'title header nama dan kolom demand
    'Nama
    ' ...
    ' ...

    baca = 4
    tulis = startrowtulis
    .Cells(startrowtulis, "D") = "Nama"
    For I = 1 To jumlahrow
        tulis = tulis + 1
        .Cells(tulis, "D") = Sheet2.Cells(baca, "A").Value
        .Cells(tulis, 4 + 2 + jumlahrow) = Sheet2.Cells(baca, "E").Value
        baca = baca + 1
    Next

    ' nama    gudang nm1    nm2 ..    ...    ....    Demand
    '
    baca = 4
    tulis = startrowtulis
    'title kolom matrix
    For I = 5 To (5 + jumlahrow + 1)      ' 2 untuk Gudang dan Demand
        If I > 5 Then
            If I = 5 + jumlahrow + 1 Then
                .Cells(tulis, I) = "Demand"
                coldemand = I
            End If
        End If
    Next
End With

```

Lampiran 9. Coding Saving Matrix Bagian 2

End Sub

End Sub

With seet

Lampiran 10. *Coding Saving Matrix* Bagian 3

```
Next I
'-----
colused = 5 ' tambah proses mulaila dari setelah gudang
```

Activate Windows  
Go to Settings to activate Windows.

## Lampiran 11. Coding Pengelompokkan Bagian 1

```

Sub tuliskelompok()

Dim Lused_tujuan As Object
Dim Lused_kelompok As Object
Set Lused_tujuan = CreateObject("System.Collections.ArrayList")
Set Lused_kelompok = CreateObject("System.Collections.ArrayList")

Dim beratkendaraan As Object
Set beratkendaraan = CreateObject("System.Collections.ArrayList")
berat = "250;3000;3000"
berat1 = Split(berat, ";")
For I = LBound(berat1) To UBound(berat1)
    b = berat1(I)
    beratkendaraan.Add b
Next

With Sheet5

'Create title pembuatan kelompok
    StartcolKelompok = colSavingMatrixStart + jumlahrow + 1
    .Cells(startrow, StartcolKelompok + 1) = "Kapasitas"
    .Cells(startrow, StartcolKelompok + 2) = "Kelompok"
    .Cells(startrow, StartcolKelompok + 3) = "Nilai Saving"
    .Cells(startrow, StartcolKelompok + 4) = "Tujuan"
    .Cells(startrow, StartcolKelompok + 5) = "Demand"
    .Cells(startrow, StartcolKelompok + 6) = "Kendaraan"

For I = 0 To beratkendaraan.Count - 1
    sisaisikendaraan = Val(beratkendaraan.Item(I))
    For J = 0 To LargestKelompok.Count - 1
        kel = LargestKelompok.Item(J)

        If kelblock = kel Then GoTo nextj

        If Not Lused_tujuan.Contains(LargestTujuan(J)) Then

            If Val(LargestDemand.Item(J)) <= Val(sisaisikendaraan) Then
                sisaisikendaraan = sisaisikendaraan - Val(LargestDemand.Item(J))

                Lused_tujuan.Add LargestTujuan.Item(J)
                Lused_kelompok.Add I + 1 ' LargestTujuan.Item(j)
            End If
        End If
    Next J
    kelblock = kel
Next I

End With

Dim x As OLEObject
Set x = Sheets("saving Matrix").OLEObjects("cmdurut")
x.Left = Sheet5.Cells(15 - 4, StartcolKelompok + 1).Left

End Sub

```

## Lampiran 12. Coding Pengelompokkan Bagian 2

```

If Not Lused_tujuan.Contains(LargestTujuan(J)) Then

    If Val(LargestDemand.Item(J)) <= Val(sisaisikendaraan) Then
        sisaisikendaraan = sisaisikendaraan - Val(LargestDemand.Item(J))

        Lused_tujuan.Add LargestTujuan.Item(J)
        Lused_kelompok.Add I + 1 ' LargestTujuan.Item(j)

        startrow = startrow + 1
        .Cells(startrow, StartcolKelompok + 1) = beratkendaraan.Item(I)
        .Cells(startrow, StartcolKelompok + 2) = I + 1
        .Cells(startrow, StartcolKelompok + 3) = Largestnvalue.Item(J)
        .Cells(startrow, StartcolKelompok + 4) = LargestTujuan.Item(J) ' "Tujuan"
        .Cells(startrow, StartcolKelompok + 5) = LargestDemand.Item(J) ' "Demand"
        .Cells(startrow, StartcolKelompok + 6) = IIf(beratkendaraan.Item(I) = "250", "Motor", "Mobil" & I)

    Else

        kelblock = kel

    End If

End If

nextj:
    Next J
    kelblock = ""
    sisaisikendaraan = 0

Next I

End With

Dim x As OLEObject
Set x = Sheets("saving Matrix").OLEObjects("cmdurut")
x.Left = Sheet5.Cells(15 - 4, StartcolKelompok + 1).Left

End Sub

```



### Lampiran 13. Coding Nearest Neighbour Bagian 1

```
Function ruteterdekat(Source As String, Dest As String) As String
Dim Dst() As Variant, jarak() As Variant
Dim dst1 As String

Set lusedurut_tujuan = CreateObject("System.Collections.ArrayList")
Set lusedurut_jarak = CreateObject("System.Collections.ArrayList")

With Sheet5

Dim rgjarak As Range
Set rgjarak = .Range(.Cells(startrow, CG), .Cells(startrow + jumlahrow, CG + 1 + jumlahrow)) ' range Jarak dari gudang ke destinatsi

If InStr(1, Dest, ",", > 0) Then
    dsts = Split(Dest, ",")
    For Z = LBound(dsts) To UBound(dsts)
        ReDim Preserve Dst(Z + 1)
        Dst(Z + 1) = dsts(Z)
    Next Z

    For y = LBound(dsts) To UBound(dsts)
        ReDim Preserve jarak(y + 1)
        dst1 = Dst(y + 1)
        If Source = "Gudang" Then
            a = GetJarakXDest_YDest(rgjarak, Source, dst1)
        Else
            a = GetJarakXDest_YDest(rgjarak, dst1, Source)
        End If
        jarak(y + 1) = a
    Next y

Call sort2(jarak, Dst)
ruteterdekat = Dst(1)

lusedurut_tujuan.Clear
For I = LBound(Dst) To UBound(Dst)
    lusedurut_tujuan.Add Dst(I)
Next
lusedurut_jarak.Clear
For I = LBound(jarak) To UBound(jarak)
    lusedurut_jarak.Add jarak(I)

```

Activate V  
Go to Setting

### Lampiran 14. Coding Workbook

```
Private Sub Workbook_open()
Menu_utama.Show
Sheet1.Activate
Userinterfaceonly = True
End Sub
```

### Lampiran 15. Coding Pendefinisian Variabel

```
Public colSavingMatrixStart As Long
Public CG As Long ' column gudang
Public RG As Long 'Row Gudang
Public LargestTujuan As Object
Public Largestnvalue As Object
Public LargestKelompok As Object
Public LargestDemand As Object
Public jumlahrow As Long
Public startrow As Long

Public lusedurut_tujuan As Object
Public lusedurut_jarak As Object
Public lused rute As Object
```