

Universitatea “Petru Maior” din Tîrgu Mureş
Facultatea de Inginerie
Sisteme automate de conducere a proceselor industriale
Master Anul I

Sisteme SCADA
Modernizare linii ambalare Adex NPK
Proiect

Masterand:
Ing. Turcu Florin Mihai

Tema proiectului

Realizarea unei aplicații de monitorizare și control a linilor de ambalare îngreșământ chimic în saci de 50Kg. Sistemul de control este compus din PLC Siemens S7-300 CPU 314C și SIWAREX FTA. Sistemul de monitorizare este compus din HMI Siemens TP1200 și stație PC cu aplicația WinCC RunTime Advance 512 Tag-uri.

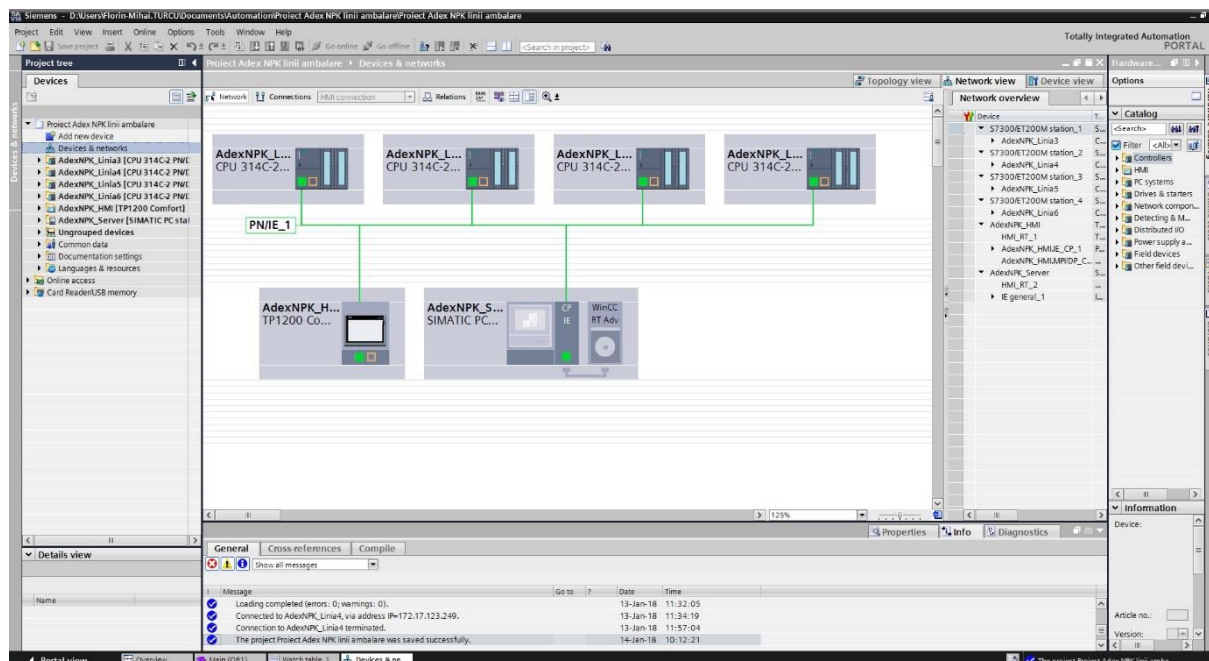


Figura 1 – Topologia sistemului

Interfața SCADA

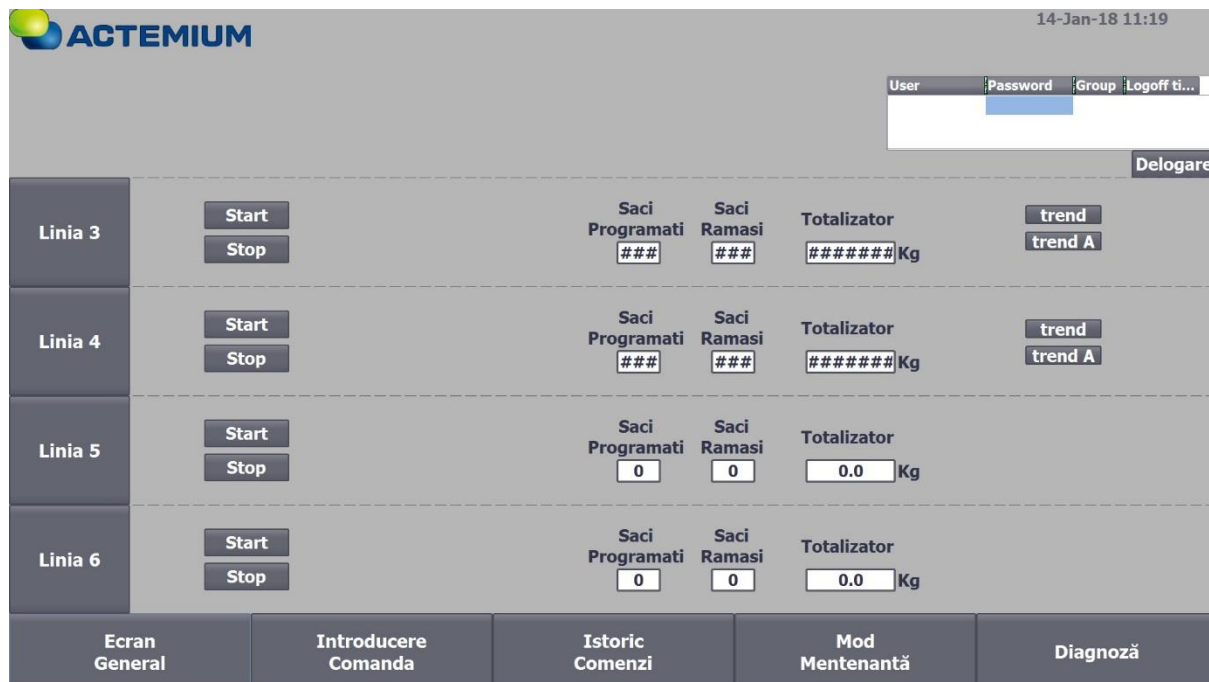


Figura 2 – Ecran general

În ecranul general avem câteva informații despre linii (Saci programați, Saci rămași, Totalizator), starea liniei (pornită/oprită). În partea de jos avem ecrane secundare aferente liniei selectate (**Introducere comandă**, **Istoric comenzi**, **Mod mentenanță**, **Diagnoză**). În partea stângă putem selecta linia pe care dorim să o vizualizăm. La selectarea liniei se va deschide un ecran cu linia selectată.

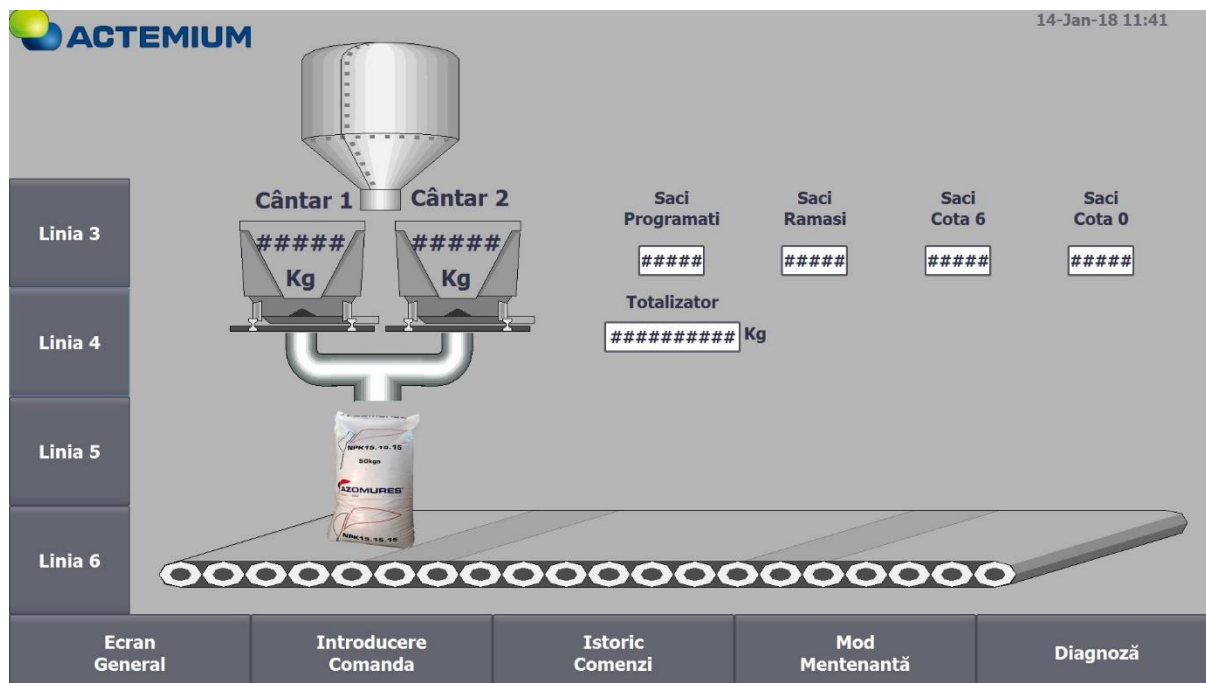


Figura 3 – Linia 4

În ecranul „**Linia 4**” avem informații cu greutatea fiecărui cântar, **saci programați** (indica numărul sacilor programați de operator), **saci rămași** (este diferența dintre numărul de saci programați și cei efectuați), **saci cota 6** (indica numărul sacilor la cota 6 metri), **saci cota 0** (indica numărul sacilor la cota 0 metri), **totalizator** (indică greutatea totală a sacilor efectuați în ziua curentă).

La apăsarea pe unul din cântare va apărea o fereastră de tip POP-UP operațiunile permise operatorului asupra cântarelor.



Figura 4 – Fereastră de tip Pop-UP

ACTEMIUM 14-Jan-18 11:58

Linia 3 Greutate Sac ##### Kg

Saci Programati #####

Linia 4 Numar Vagon #####

Sort 20.20.0

Tip Dozare Sac

Linia 5

Linia 6

Ecran General Introducere Comanda Istoric Comenzi Mod Mentenanță Diagnoză

Figura 5 – Introducere Comandă

În ecranul „**Introducere Comandă**” avem posibilitatea de a introduce greutatea sacului, numărul de saci, număr de vagon, sortul îngrășământului și tipul de dozare (sac sau vrac).

ACTEMIUM 14-Jan-18 12:04

	Data	Număr Comandă	Saci Programati	Nr. Sac Cota 6	Nr. Sac Cota 0	Greutate Sac(Kg)	Totalizator (Kg)	Număr Vagon	Sort
Linia 3	01-Jan-99 12:00:00	0	0	0	0	0.00	0.00		
	01-Jan-99 12:00:00	0	0	0	0	0.00	0.00		
	01-Jan-99 12:00:00	0	0	0	0	0.00	0.00		
	01-Jan-99 12:00:00	0	0	0	0	0.00	0.00		
Linia 4	01-Jan-99 12:00:00	0	0	0	0	0.00	0.00		
	01-Jan-99 12:00:00	0	0	0	0	0.00	0.00		

Linia 5 Data: 01-Jan-99 12:00:00

Linia 6 Căutare

Ecran General Introducere Comanda Istoric Comenzi Mod Mentenanță Diagnoză

Figura 6 – Istoric Comenzi

În ecranul „**Istoric Comenzi**” putem vizualiza după dată toate comenzile efectuate.

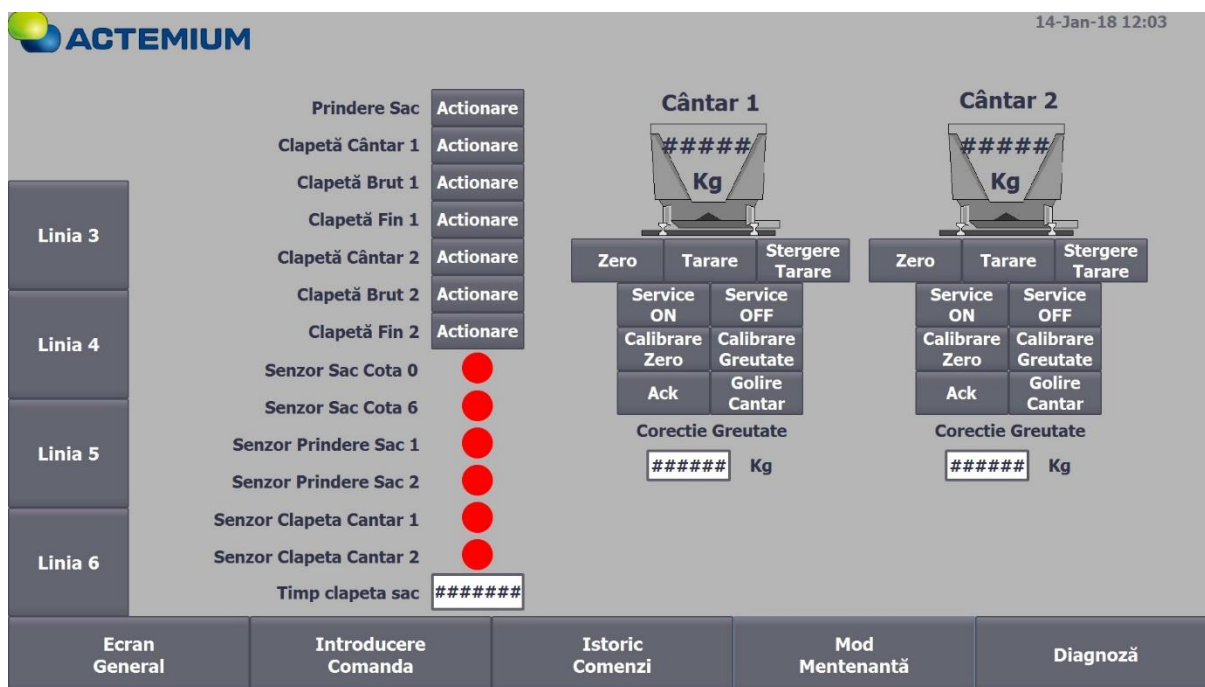


Figura 7 – Mod Mentenanță

În ecranul „**Mod Mentenanță**” putem acționa manual fiecare element din sistem. Clapetele (prindere sac, brut, fin, cantar1, cantar2) sunt actionate pneumatic cu comandă electrică cu ajutorul electroventilelor 5/3 cai. Confirmarea senzorilor de tip inductiv este vizibilă prin înverzirea cercurilor roșii. Se pot face corecții de greutate prin introducerea valori necesare. Sunt posibile diferite comenzi asupra cântarelor precum zero, tarare, mod service, calibrare. Calibrarea cântarului se va face cu greutate etalon de 50Kg, fiind posibil doar in modul service.

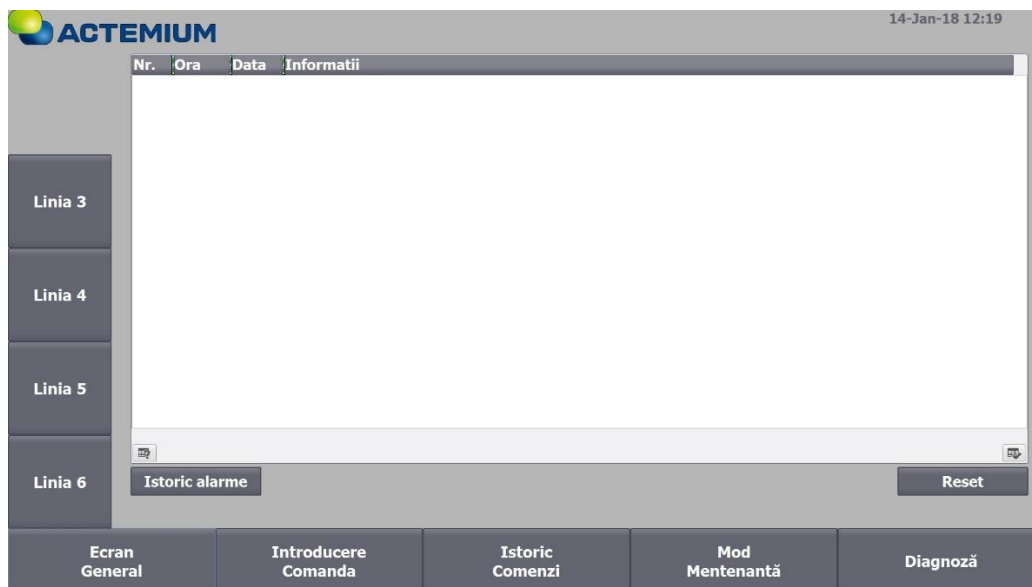


Figura 8 – Diagnoză

În ecranul „**Diagnoză**” vor apărea mesajele de eroare active. La apăsarea butonului „Istoric alarme” vor apărea toate mesajele de eroare pe o perioada de 30 zile.

Conexiunea MySQL cu sistemul SCADA

Pentru a realiza conexiunea între baza de date MySQL și sistemul Scada WinCC RT Advance, este nevoie de instalarea programului **wampserver**, acesta va fi introdus în startup pentru a rula automat la pornirea PC. Se va instala driver-ul **mysql-connector-odbc**. În ODBC Data Source Administrator se va adăuga o nouă conexiune cu driver-ul instalat.

Se accesează <http://localhost:8080/phpmyadmin/> unde va fi creată o nouă bază de date aferentă liniei. În fiecare bază de date va fi creat un tabel cu denumirea „istoric” având 10 coloane cu caracteristicile din Figura 9.

#	Nume	Tip	Interclasare	Proprietăți	Nul	Implicit	Comentarii	Extra
1	ID 	int(11)			Nu	None		AUTO_INCREMENT
2	Data	datetime			Nu	None		
3	Numar Comanda	int(11)			Nu	None		
4	Saci Programati	int(11)			Nu	None		
5	Numar Sac Cota 6	int(11)			Nu	None		
6	Numar Sac Cota 0	int(11)			Nu	None		
7	Greutate Sac	float			Nu	None		
8	Totalizator	float			Nu	None		
9	Numar Vagon	varchar(12)	latin1_swedish_ci		Nu	None		
10	Sort	char(12)	latin1_swedish_ci		Nu	None		

Figura 9 – Tabel bază de date

Citirea și scrierea în baza de date se va face prin intermediul unor scripturi din WinCC RT Adv.

VB Script pentru adăugarea unui nou rând în baza de date:

```
1. 'Declaration of local tags - Deklaration von lokalen Variablen
2. Dim conn, rst, SQL_Table, connString, sort
3.
4. On Error Resume Next
5.
6. Set conn = CreateObject("ADODB.Connection")
7. Set rst = CreateObject("ADODB.Recordset")
8.
9. 'Open data source - Datenquelle öffnen
10.
11. connString = "Driver={MySQL ODBC 5.3 Driver}; Server=127.0.0.1;" & _
12.             "Data Source=Test; User=Test; Database=linia3;"
13. conn.Open connString
14.
15. Select Case SmartTags("DB_HMI_SortODBC(3)")
16.     Case 0
17.         sort = "20.20.0"
18.     Case 1
19.         sort = "27.13,5.0"
20.     Case 2
21.         sort = "16.16.16"
22.     Case 3
```

```

23.         sort = "15.15.15SOP"
24.     Case 4
25.         sort = "15.9.20"
26.     Case 5
27.         sort = "16.16.8"
28.
29. End Select
30.
31. 'Definition of data record - Definition des Datensatzes
32. SQL_Table = "INSERT INTO istoric VALUES (NULL, Now(),'" & SmartTags("DB_HMI_Numa
rComanda(3)") & "' , '" & SmartTags("DB_HMI_SaciProgramatiODBC(3)") & _
33.         "' , '" & SmartTags("DB_HMI_NumarSaciCota6(3)") & "' , '" & SmartTags
("DB_HMI_NumarSaciCota0(3)") & "' , '" & SmartTags("DB_HMI_GreutateSacODBC(3)") & "'
, '" & SmartTags("DB_HMI_Totalizator(3)") & "' , '" & SmartTags("DB_HMI_NumarVagonOD
BC(3)") & "' , '" & sort & "' )"
34.
35. 'Insert the data record of the table - Datensatz in die Tabelle hinzufügen
36. Set rst = conn.Execute(SQL_Table)
37.
38.
39. 'Close data source - Datenquelle schließen
40. conn.close
41.
42. Set rst = Nothing
43. Set conn = Nothing

```

VB Script pentru citirea unui rând din baza de date:

```

1. Dim conn, rst, SQL_Table, i, j, connectionString, startDate, stopData, linia , test
2.
3. On Error Resume Next
4.
5. If SmartTags("DB_SCADA_LinieSelectata(3)") = 3 Then
6.     linia="Data Source=Test; User= Test; Database=linia3;"
7. End If
8. If SmartTags("DB_SCADA_LinieSelectata(4)") = 4 Then
9.     linia="Data Source= Test; User= Test; Database=linia4;"
10. End If
11.
12. Set conn = CreateObject("ADODB.Connection")
13. Set rst = CreateObject("ADODB.Recordset")
14.
15. connectionString = "Driver={MySQL ODBC 5.3 Driver}; Server=127.0.0.1;" & _
16.         linia
17. conn.Open connectionString
18.
19. startDate= Year(SmartTags("startDate_1")) & "-"
    " & Month(SmartTags("startDate_1")) & "-" & Day(SmartTags("startDate_1"))
20. stopData= Year(SmartTags("startDate_1")) & "-"
    " & Month(SmartTags("startDate_1")) & "-" & Day(SmartTags("startDate_1")+1)
21.
22. SQL_Table = "SELECT * FROM istoric WHERE Data BETWEEN " & "'" & startDate & "'" & "
AND" & "'" & stopData & "'" & " ORDER by ID ASC"
23.
24. Set rst = conn.Execute(SQL_Table)
25.
26.
27. If Not (rst.EOF And rst.BOF) Then
28.     'Vergleich ob "End of File" oder "Begin of File" ist, wenn nicht wird der Zeiger
    auf den Ersten Eintrag zurueckgesetzt
29.     'Compare if "End of File" or "Begin of File" exists, if not the pointer will be
    reset to the first entry
30.
31.     rst.MoveFirst 'reset to 1st entry - auf 1. Eintrag zuruecksetzen
32.

```

```

33. 'Definition of local tags - Definition von lokalen Variablen
34. j=0
35.
36. 'Amount of the entries in the table - Anzahl der Tabelleneinträge
37. Do
38.     j=j+1
39.     rst.MoveNext
40. Loop Until rst.EOF
41.
42. rst.MoveFirst 'reset to 1st entry - auf 1. Eintrag zuruecksetzen Do
43.
44. 'Selection with Arrow Buttons - Auswahl mit den Pfeil-Tasten
45. If SmartTags("nTab")>=j-6 Then
46.     SmartTags("nTab")=j-6
47. End If
48. If SmartTags("nTab")<j-5 Then
49.     For i=1 To SmartTags("nTab")
50.         rst.MoveNext
51.     Next
52. End If
53. If SmartTags("nTab")<0 Then
54.     SmartTags("nTab")=0
55. End If
56.
57. For i=1 To 6
58.     'Entries of the table - Einträge in die Tabelle
59.     If rst.EOF Then
60.         SmartTags("Value_" & i & "_0") = 0
61.         SmartTags("Value_" & i & "_1") = 0
62.         SmartTags("Value_" & i & "_2") = 0
63.         SmartTags("Value_" & i & "_3") = 0
64.         SmartTags("Value_" & i & "_4") = 0
65.         SmartTags("Value_" & i & "_5") = 0
66.         SmartTags("Value_" & i & "_6") = 0
67.         SmartTags("Value_" & i & "_7") = 0
68.         SmartTags("Value_" & i & "_8") = 0
69.     Else
70.         SmartTags("Value_" & i & "_0") = rst.Fields(1).Value
71.         SmartTags("Value_" & i & "_1") = rst.Fields(2).Value
72.         SmartTags("Value_" & i & "_2") = rst.Fields(3).Value
73.         SmartTags("Value_" & i & "_3") = rst.Fields(4).Value
74.         SmartTags("Value_" & i & "_4") = rst.Fields(5).Value
75.         SmartTags("Value_" & i & "_5") = rst.Fields(6).Value
76.         SmartTags("Value_" & i & "_6") = rst.Fields(7).Value
77.         SmartTags("Value_" & i & "_7") = rst.Fields(8).Value
78.         SmartTags("Value_" & i & "_8") = rst.Fields(9).Value
79.         rst.MoveNext
80.     End If
81. Next
82.
83. rst.close
84. Else
85.     ShowSystemAlarm "No entries are available."
86. End If
87.
88. 'Close data source - Datenquelle schließen
89. conn.close
90.
91. Set rst = Nothing
92. Set conn = Nothing

```

O dată la 24 ore se salveaza automat din baza de date in fisier .csv toate datele din ziua curentă.

VB Script salvare automată fisier .csv:

```
1. Dim conn, rst, SQL_Table, connString, strQry, filsSysObj, csvFile, Separator, Column
   , i, Fields, startData, dataFisier, stopData
2. Dim recordCount
3. Dim regEx
4. Dim dtWait
5. Set regEx = New RegExp
6. regEx.Pattern = "\r|\n|,|'"
7.
8. Set conn = CreateObject("ADODB.Connection")
9. Set rst = CreateObject("ADODB.Recordset")
10.
11. 'Open data source - Datenquelle öffnen
12.
13. connString = "Driver={MySQL ODBC 5.3 Driver}; Server=127.0.0.1;" & _
14.              "Data Source=Test; User=Test; Database=linia3;"
15. conn.Open connString
16.
17. startData= Year(Date) & "-" & Month(Date) & "-" & Day(Date)-1
18.
19. stopData=Year(Date) & "-" & Month(Date) & "-" & Day(Date)
20.
21. dataFisier= Day(Date)-1 & "-" & Month(Date) & "-" & Year(Date)
22.
23. strQry = "SELECT * FROM istoric WHERE Data BETWEEN " & "'" & startData & "'" & "AND"
   " & "'" & stopData & "'" & " ORDER by ID ASC"
24. Set rst = conn.execute(strQry)
25.
26. Set filsSysObj = CreateObject("Scripting.FileSystemObject")
27. Set csvFile = filsSysObj.OpenTextFile ("D:\ODBC\SalvareAutomata\Linia3\" & dataFisie
   r & ".csv",8, 2)
28. csvFile.WriteLine ("sep=;")
29. csvFile.WriteLine ("Id" & ";" & "Data" & ";" & "Numar Comanda" & ";" & "Saci Program
   ati" & ";" & "Numar Sac Cota 6" & ";" & "Numar Sac Cota 0" & ";" & "Greutate Sac (Kg
   )" & ";" & "Totalizator (Kg)" & ";" & "Numar Vagon" & ";" & "Sort")
30. recordCount = rst.Fields.Count
31.
32. Do Until rst.EOF
33.
34. Separator = ""
35. For i = 0 To rst.Fields.Count - 1
36. Column = rst.Fields(i).Value & ""
37. If regEx.Test( Column ) Then
38. Column = "" & Replace( Column, "", " " ) & ""
39. End If
40. csvFile.Write Separator & Column
41. Separator = ";"
42. Next
43. csvFile.Write vbNewLine
44. rst.MoveNext
45. Loop
46.
47. SmartTags("DB_HMI_ResetTotalizator(3)") = True
48. dtWait = DateAdd("s", 1, Now())
49. Do Until (Now() > dtWait)
50. Loop
51. SmartTags("DB_HMI_ResetTotalizator(3)") = False
```