

Web-Server mit Temperaturanzeige

in Gambas




```

letztes Systemdatum des Servers unbekannt
<br><br>
<h3>
<a name="13" href="abmelden">Abmelden</a>
</h3>
</div>
</body>
</html>

```

pw.html

```

<!DOCTYPE html>
<html>
<head>
<title></title>
</head>
<body text="#000000" bgcolor="#FF7F00" link="#FF0000" alink="#FF0000" vlink="#FF0000">
<div align="center">
<form action="" method="" target="">
<br>
<h2>Passworteingabe</h2>
<br>
<input type="Password" name="p1" value="" size="20" maxlength="12">
<br><br><br>
<input type="Submit" value="PW OK" Style="WIDTH:140px" WIDTH="140px" Style= "HEIGHT:30px" HEIGHT ="30px" >
<br>
<br>
<input type="Submit" value="Hauptseite laden" Style="WIDTH:140px" WIDTH="140px" Style= "HEIGHT:30px" HEIGHT ="30px" >
</form>
</div>
</body>
</html>

```

' Gambas class file

```

'Web Server geht gut
'GPIO 26 ist verwendet am Raspberrry als Schalter
'GPIO 17 für DHT22 Temp Sensor

'ServerSocket1.status
'7 = verbunden, 6 = nicht verbunden, 0 = inaktiv

Public thisSocket As Socket
Public statuszustand1 As Integer = 0
Public pwmerker As Integer = 0 'Passwortmerker
Public uhrlan As String
Public uhrlaus As String
Public uhr2an As String
Public uhr2aus As String
Public statusuhr As Integer = 0 '=0 Uhr aus =1 Uhr an
Public tempfehler As Integer 'Anzahl der Temp Messfehler zum Abschalten der Messungen
Public letztetemp As String
Public betrieb1 As Integer = 0 'für Schaltuhr
Public betrieb2 As Integer = 0 'für Schaltuhr
Public uhrschaltean As Integer '=1 wenn uhr an geschaltet hat

Public Sub ServerSocket1_Connection(strHostIP As String)
    Dim inhalt As String
    Dim inhalt2 As String
    Dim s As String
    Dim dateiname As String

    If ServerSocket1.Status > Net.Inactive Then

        thisSocket = ServerSocket1.Accept()
        thisSocket.Blocking = False
        If thisSocket.Status = Net.Connected Then
            TextArea1.Text &= "Connected: " & thisSocket.RemoteHost & ":" & thisSocket.RemotePort & gb.Lf
        End If

        If pwmerker = 1 Then dateiname = "/index.html"
    End Sub

```

```

If pwmerker = 0 Then dateiname = "/pw.html"

s = File.Load(Application.Path & dateiname) 'HTML Seite einlesen

Wait 0.5

'Auswertung des Zustands und in Webseite einfügen
If statuszustand1 = 1 Then
    s = Replace(s, "aktueller Status Zustand1 ist unbekannt", "aktueller Status Zustand1 ist an")
Endif
If statuszustand1 = 0 Then
    s = Replace(s, "aktueller Status Zustand1 ist unbekannt", "aktueller Status Zustand1 ist aus")
Endif

'Uhr Status
If statusuhr = 1 Then 'Uhr an
    s = Replace(s, "Schaltuhr Status ist unbekannt", "Schaltuhr Status ist an")
Endif
If statusuhr = 0 Then 'Uhr aus
    s = Replace(s, "Schaltuhr Status ist unbekannt", "Schaltuhr Status ist aus")
Endif

'Temperatur
If letztetemp <> "" Then 'Temp ausgeben
    s = Replace(s, "letzte Temperatur ist unbekannt", "letzte Temperatur:&nbsp;&nbsp; " & letztetemp)
Endif

'Schaltzeiten eintragen
s = Replace(s, "v11", uhr1an)
s = Replace(s, "v12", uhr1aus)
s = Replace(s, "v21", uhr2an)
s = Replace(s, "v22", uhr2aus)
'Systemdatum eintragen
s = Replace(s, "letztes Systemdatum des Servers unbekannt", "letztes Systemdatum des Servers:&nbsp;&nbsp; " &
FMain.Label16.Text)

inhalt2 = s 'Inhalt der HTML Seite

'Vorspann mit Längenangabe der HTML Seite

```

```

    inhalt = "HTTP/1.1 200 OK \r\nContent-Length: " & Str(Len(inhalt2)) & "\r\n\r\n" & inhalt2 'HTML Vorspann \r =
CHR(10) \n = CHR(13)

    Write #thisSocket, inhalt, Len(inhalt) 'senden an Browser

Endif

thisSocket.Close
Catch 'Fehlerbehandlung
    If Error Then
        thisSocket.Close
        Return
    Endif
End

Public Sub Socket_Read()
    '- empfangen vom den Clients
    Dim strMsg As String
    Dim inhalt As String

    strMsg = thisSocket.ReadLine() 'besser
    TextArea1.Text &= "Texteingang:" & strMsg & gb.Lf
    Wait
    inhalt = strMsg

    'empfangene Daten auslesen

    If InStr(inhalt, "p1=reinhard") <> 0 Then 'PW stimmt
        inhalt = ""
        strMsg = ""
        pwmerker = 1 'Passwortmerker = 1
        Return
    Endif

    'Zeiten speichern muß vor den anderen Abfragen stehen ???
    If InStr(strMsg, "speichern") <> 0 Then 'Zeit speichern

    uhr1an = Mid(inhalt, InStr(inhalt, "z11") + 4, 2) & ":" & Mid(inhalt, InStr(inhalt, "z11") + 9, 2)
    uhr1aus = Mid(inhalt, InStr(inhalt, "z12") + 4, 2) & ":" & Mid(inhalt, InStr(inhalt, "z12") + 9, 2)
    uhr2an = Mid(inhalt, InStr(inhalt, "z21") + 4, 2) & ":" & Mid(inhalt, InStr(inhalt, "z21") + 9, 2)

```

```

uhr2aus = Mid(inhalt, InStr(inhalt, "z22") + 4, 2) & ":" & Mid(inhalt, InStr(inhalt, "z22") + 9, 2)
Wait
'in Datei speichern
File.Save(Application.Path & "/uhr1an.dat", uhr1an)
File.Save(Application.Path & "/uhr1aus.dat", uhr1aus)
File.Save(Application.Path & "/uhr2an.dat", uhr2an)
File.Save(Application.Path & "/uhr2aus.dat", uhr2aus)
Wait
FMain.Label7.Text = uhr1an
FMain.Label8.Text = uhr1aus
FMain.Label9.Text = uhr2an
FMain.Label10.Text = uhr2aus
strMsg = ""
Return
Endif

If InStr(inhalt, "abmelden") <> 0 Then 'abmelden
    inhalt = ""
    strMsg = ""
    pwmerker = 0 'Passwortmerker = 0
    Return
Endif

If InStr(inhalt, "zulan") <> 0 Then 'schalten an
    'Befehle für zulan- 'zuan- Strich wegen der gleichen Länge der Internetseite
    inhalt = ""
    strMsg = ""
    statuszustand1 = 1
    Label2.Text = "an"
    Label2.Background = Color.RGB(255, 0, 0)
    Shell "gpio -g Write 26 1"
    'in Datei schreiben
    File.Save(Application.Path & "/zustand1.dat", "1")
    Return
Endif

If InStr(inhalt, "zulaus") <> 0 Then 'schalten aus
    'Befehle für zulaus
    inhalt = ""
    strMsg = ""

```

```

statuszustand1 = 0
Label2.Text = "aus"
Label2.Background = Color.RGB(0, 255, 0)

'Ausschalten aber nur wenn Schaltuhr nicht gerade eingeschaltet ist
If statusuhr = 0 Or uhrschalteran = 0 Then 'nur ausschalten wenn manuell nicht an ist
    Shell "gpio -g Write 26 0"
Endif

If statusuhr = 1 And uhrschalteran = 1 Then Shell "gpio -g Write 26 1" 'wenn Uhr angeschaltet ist nicht abschalten

'in Datei speichern
File.Save(Application.Path & "/zustand1.dat", "0")
Return
Endif

If InStr(inhalt, "ulan") <> 0 Then 'Schaltuhr an
    'Befehle für zulaus
    inhalt = ""
    statusuhr = 1
    Label4.Text = "an"
    Label4.Background = Color.RGB(255, 0, 0)
    Timer1.Enabled = True
    'in Datei schreiben
    File.Save(Application.Path & "/uhr.dat", "1")
    Return
Endif

If InStr(inhalt, "ulaus") <> 0 Then 'Schaltuhr aus über Timer wird dann Schaltuhr nicht mehr ausgeführt
    'Befehle für zulaus
    inhalt = ""
    strMsg = ""
    statusuhr = 0
    Label4.Text = "aus"
    Label4.Background = Color.RGB(0, 255, 0)
    If statuszustand1 <> 1 Then Shell "gpio -g Write 26 0" 'nicht ausschalten wenn manuell an ist
    Timer1.Enabled = False
    'in Datei schreiben
    File.Save(Application.Path & "/uhr.dat", "0")
    Return

```



```

Endif

If Len(TextArea1.Text) > 1000 Then TextArea1.Text = "" 'Überlauf

End

Public Sub ServerSocket1_Error()
    Message.Error("Unable to bind socket")
End

Public Sub Form_Open()

Shell "gpio Export 26 out" 'port 22 aktivieren
Shell "gpio -g Write 26 0" 'port 22 ausschalten

' Verbinden

    ServerSocket1.Port = 8080
    ServerSocket1.Type = Net.Internet

    ServerSocket1.Listen()

    If ServerSocket1.Status = Net.Active Then
        TextArea1.Text &= "Listening on Port: " & ServerSocket1.Port & gb.CrLf
    End If

'Schaltuhrzeiten aus Dateien lesen
uhr1an = File.Load(Application.Path & "/uhr1an.dat")
uhr1aus = File.Load(Application.Path & "/uhr1aus.dat")
uhr2an = File.Load(Application.Path & "/uhr2an.dat")
uhr2aus = File.Load(Application.Path & "/uhr2aus.dat")

FMain.Label7.Text = uhr1an
FMain.Label8.Text = uhr1aus
FMain.Label9.Text = uhr2an
FMain.Label10.Text = uhr2aus

```

```

'gespeicherte Zustände holen
statuszustand1 = File.Load(Application.Path & "/zustand1.dat")
statusuhr = File.Load(Application.Path & "/uhr.dat")

If statuszustand1 = 1 Then
    Label2.Text = "an"
    Label2.Background = Color.RGB(255, 0, 0)
    Shell "gpio -g Write 26 1" 'port 22 anschalten
Endif

If statusuhr = 1 Then
Label4.Text = "an"
    Label4.Background = Color.RGB(255, 0, 0)
    timer1.Enabled = True
Endif

End

Public Sub Form_Close()
    Shell "gpio -g Write 26 0" 'port 22 ausschalten
    ServerSocket1.close
    FMain.Close
End

Public Sub Timer1_Timer()
'Schaltzeiten auswerten und schalten

If statusuhr = 0 Then
    FMain.Label14.Text = ""
    Return 'Uhr abgeschaltet
Endif

FMain.Label14.Text = Format(Time, "hh:nn") 'Schaltuhrzeit anzeigen

If Time >= CDate(uhrlan) And Time <= CDate(uhrlaus) Then

```

```

betriebl = 1
Else
betriebl = 0
Endif

If Time >= CDate(uhr2an) And Time <= CDate(uhr2aus) Then
betrieb2 = 1
Else
betrieb2 = 0
Endif

If betriebl = 1 Or betriebl2 = 1 Then
uhrschalte = 1
Else
uhrschalte = 0
Endif

If uhrschalte = 1 And statusuhr = 1 Then Shell "gpio -g Write 26 1"

If uhrschalte = 0 And statuszustand1 = 0 Then Shell "gpio -g Write 26 0"
If statuszustand1 = 1 Then Shell "gpio -g Write 26 1" 'bleibt an

End

Public Sub Timer2_Timer() 'Zeit anzeigen
FMain.Label16.Text = Format(Now, "dd.mm.yyyy - hh:nn") 'Systemzeit

End

Public Sub Timer3_Timer() 'Intervall für Tempmessung
'Temperatur Messung
Dim tempinhalt As String
Dim pos As Integer
Dim gesamt As String

If FMain.CheckBox1.Value = False Then
FMain.Timer3.Enabled = False 'Timer anhalten
Return 'Haken nicht gesetzt für Tempmessung
Endif

```

```

Shell "/home/pi/Adafruit_Python_DHT/examples/AdafruitDHT.py 22 17" To tempinhalt 'Temp abrufen
Wait

pos = InStr(tempinhalt, "Temp=")
If pos <> 0 Then
    gesamt = Mid(tempinhalt, pos, 11) & " " & Format(Now, "dd.mm.yyyy hh:nn")
    FMain.ListBox1.Add(gesamt)
    tempfehler = 0 'Fehler zurück setzen
Endif
If pos = 0 Then
    tempfehler = tempfehler + 1 'Fehler erhöhen
    FMain.ListBox1.Add("Fehler " & Format(Now, "dd.mm.yyyy hh:nn"))
Endif

If FMain.ListBox1.Count > 4 Then
    FMain.ListBox1.Remove(0) 'ältestes Element löschen
    FMain.ListBox1.index = 3 'Element markieren für auslesen
    FMain.Label18.text = FMain.ListBox1.Text 'markiertes Element anzeigen
    letztetemp = FMain.ListBox1.Text 'markiertes Element in Glob Variable für Anzeige in HTML
Endif

If tempfehler > 3 Then 'nach 3 Fehlern Temp Messung abschalten
    FMain.CheckBox1.Value = False 'Haken aus Checkbox
    FMain.Timer3.Enabled = False 'Timer anhalten
Endif

End

Public Sub CheckBox1_Click()
    'Haken entfernen oder setzen

    If FMain.CheckBox1.Value = True Then 'Haken gesetzt
        FMain.Timer3.Enabled = True
        tempfehler = 0
    Endif

    If FMain.CheckBox1.Value = False Then 'Haken raus
        FMain.Timer3.Enabled = False

```

```
tempfehler = 0  
Endif  
End
```