28.2.1 Crypt.MD5

Die Funktion *Crypt.MD5* verschlüsselt eine Passwort-Zeichenkette nach dem MD5-Algorithmus mit folgender Syntax:

```
Static Function MD5 ( Password As String [ , Prefix As String ] ) As String
```

Es kann optional ein Präfix verwendet werden, der genau acht Zeichen lang ist. Die Zeichen stammen aus dem folgenden Zeichenvorrat:

```
0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ./
```

Wenn der Präfix nicht angegeben wird, dann wird ein zufälliger Wert für den Präfix gesetzt.

Bitte beachten Sie, dass es keine Methode gibt, um eine mit der Funktion *Crypt.MD5* verschlüsselte Passwort-Zeichenkette wieder zu entschlüsseln. Das hat Konsequenzen für den Einsatz der Komponente *Crypt*. Sie können nur mit der Methode *Crypt.Check(..)* prüfen, ob ein eingegebenes Passwort – das intern nach dem Algorithmus MD5 verschlüsselt wird – mit einem im Programm oder einer Datei hinterlegten verschlüsselten Passwort übereinstimmt oder nicht.

Aus diesem Grunde wird Ihnen ein MD5-Passwort-Generator vorgestellt, mit dem Sie *starke* Passwörter generieren können, um diese dann in Ihren Programmen zu verwenden.

Ein starkes Passwort kann m.E. so definiert werden:

- Das Passwort besteht aus mindestens 8 Zeichen.
- Das Passwort enthält mindestens 1 Großbuchstaben.
- Das Passwort enthält mindestens 1 Kleinbuchstaben.
- Das Passwort enthält mindestens 1 Ziffer.
- Das Passwort enthält mindestens ein Sonderzeichen aus einem definierten Zeichenvorrat.

28.2.1.1 MD5-Passwort-Generator

Der vorgestellte MD5-Passwort-Generator verwendet die Funktion *Crypt.MD5,* ermöglicht die Verwendung eines Präfixes, prüft den Präfix und die Stärke der eingegebenen Passwort-Zeichenkette.

Abbildung 28.2.1.1.1: MD5-Passwort-Generator mit zufälligem Präfix

Hier eine Auswahl von verschlüsselten Passwörtern – ohne fest vorgegebenem Präfix – für das Klartext-Passwort '#GAMbas+340' :

- $1$2usFQfDf$xOBo.OWT0gYqf7r4d8fRM1
- $1$6RtPjUJj$NK3goi13w3KOOhGvWswTe1
- $1$1yNAMhZe$.LXOmkwREuvL7i7NFsmU3.
- 1Amre/.4B$YA6j2M0TcjXURXvYtXS6p/

Wenn es sich beim Klartext-Passwort nicht um ein starkes Passwort handelt, dann wird das signalisiert und die Kriterien für ein starkes Passwort angezeigt.

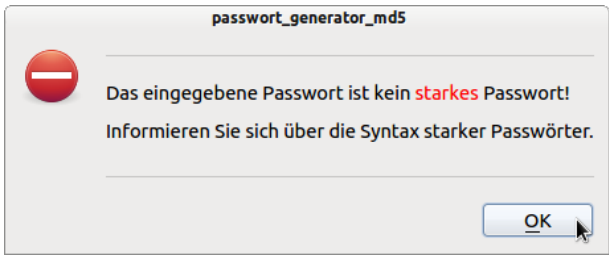

Abbildung 28.2.1.1.2: Schwaches Passwort

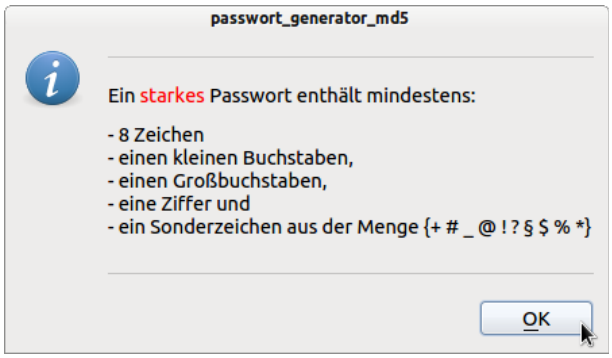

Abbildung 28.2.1.1.3: Hinweise für ein starkes Passwort

Sie können auch einen Präfix verwenden. Dann gibt es für diesen einen Präfix genau ein verschlüsseltes Passwort:

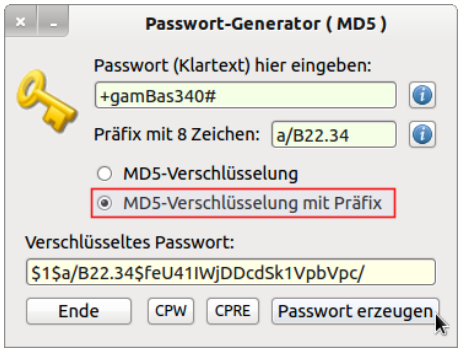

Abbildung 28.2.1.1.4: Verschlüsselung mit einem Präfix

Die Syntax des Präfixes wird geprüft und bei einem Fehler werden entsprechende Meldungen und Hinweise angezeigt:

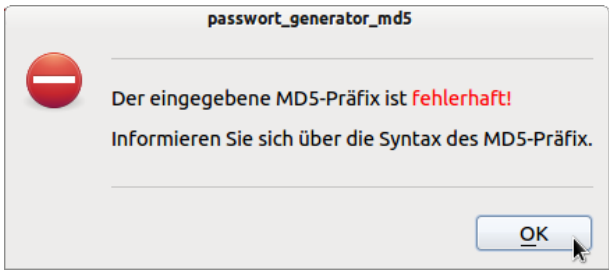

Abbildung 28.2.1.1.5: Fehlermeldung

Sie erhalten danach Hinweise darauf, welche Kriterien für einen korrekten Präfix einzuhalten sind:

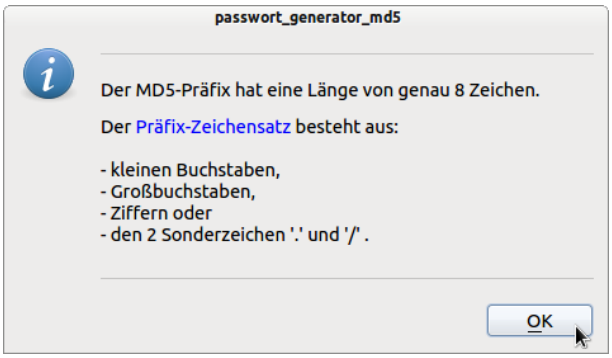

Abbildung 28.2.1.1.6: Hinweise für einen korrekten Präfix

Im Quelltext werden für die Prüfung starker Passwörter oder korrekter Präfixe u.a. auch reguläre Ausdrücke eingesetzt. Ansonsten ist der Quelltext ohne Überraschungen und wird nur in Auszügen angegeben:

```
' Gambas class file
' Die Komponenten gb.crypt und gb.pcre müssen eingebunden werden

Public Sub Form_Open()
  FPasswort.Center
  ...
End ' Form_Open()

Public Sub btnPasswortGenerieren_Click()
  Dim sStrongPassword, sValidPrefix, sMessage As String

  If Len(txtPasswortEingabe.Text) = 0 Then
     Message.Error("Geben Sie ein Passwort ein!")
     txtPasswortEingabe.SetFocus
     Return
  Endif ' Len(txtPasswortEingabe.Text) = 0 ?

  If InStr(txtPasswortEingabe.Text, Chr(32)) <> 0 Then
     Message.Error("<hr><br>Das Passwort darf <font color='red'>kein</font> Leerzeichen \
                    enthalten!<br><hr>")
     txtPasswortEingabe.SetFocus
     Return
  Endif ' InStr(txtPasswortEingabe.Text, Chr(32)) <> 0 ?

  If CheckStrongPassword(txtPasswortEingabe.Text) = True Then
     sStrongPassword = txtPasswortEingabe.Text
  Else
     sMessage = "<hr>"
     sMessage &= "<p>Das eingegebene Passwort ist kein <font color='red'>starkes</font> \
                  Passwort!</p>"
     sMessage &= "Informieren Sie sich über die Syntax starker Passwörter.<br>"
     sMessage &= "<hr>"
     Message.Error(sMessage)
     btnInformation_Click()
     txtPasswortEingabe.SetFocus
     Return
  Endif ' CheckStrongPassword(...)

  If optMD5.Value = True Then
     txtKeyAusgabe.Text = Crypt.MD5(sStrongPassword)
  Else
     If CheckPrefix(txtPrefixEingabe.Text) = True Then
        sValidPrefix = txtPrefixEingabe.Text
        txtKeyAusgabe.Text = Crypt.MD5(sStrongPassword, sValidPrefix)
     Else
        sMessage = "<hr>"
        sMessage &= "<p>Der eingegebene MD5-Präfix ist <font color='red'>fehlerhaft! \
                     </font></p>"
        sMessage &= "Informieren Sie sich über die Syntax des MD5-Präfix.<br>"
        sMessage &= "<hr>"
        Message.Error(sMessage)
```

```
        btnPrefixInformation_Click()
        txtPrefixEingabe.SetFocus
     Endif ' CheckPrefix(txtPrefixEingabe.Text) = True ?
  Endif ' optMD5.Value = True ?
End ' btnPasswortGenerieren_Click()

Private Function CheckPrefix(sPrefix As String) As Boolean
  Dim sSubject, sPattern As String

  sSubject = sPrefix
  sPattern = "^([a-zA-Z0-9./]{8})$"

  If Match(sSubject, sPattern) = True Then
     Return True
  Else
     Return False
  Endif ' Match(sSubject, sPattern) = True ?

End ' CheckPrefix(sPrefix As String) As Boolean

Private Function CheckStrongPassword(sPasswort As String) As Boolean
  Dim sSubject, sPattern As String

  sSubject = sPasswort
  sPattern = "(?=^.{8,}$)(?=.*[A-Z])(?=.*[a-z])(?=.*\\d)(?![.\n])(?=.*[+#_@!?$$%*]).*$"

  If Match(sSubject, sPattern) = True Then
     Return True
  Else
     Return False
  Endif ' Match(sSubject, sPattern) = True ?

End ' CheckStrongPassword(sPasswort As String) As Boolean

Public Function Match(Subject As String, Pattern As String) As Boolean
  Dim rRegex As Regexp

  rRegex = New Regexp(Subject, Pattern)

  If rRegex.Offset = -1 Then
     Return False
  Else
     Return True
  Endif ' rRegex.Offset = -1
End ' Match(...)

...

Public Sub btnClose_Click()
  FPasswort.Close
End ' btnClose_Click()
```

Das vollständige Projekt finden Sie im Download-Bereich und eine Anwendung des MD5-Passwort-Generators im folgenden Kapitel.