

23.12.5 Projekt PDF-Reader

In diesem Kapitel wird ein erprobtes Projekt vorgestellt. Es ermöglicht die Auswahl und Anzeige einer PDF-Datei. Das Projekt nutzt die Komponente gb.poppler.

Das Projekt 'PDF-Reader' verfügt über folgende Funktionalität:

- Auswahl der PDF-Datei in einem Dateiauswahl-Dialog
- Anzeige von ausgewählten Eigenschaften der PDF-Datei in einem separaten Fenster
- Suchfunktion mit Anzeige der Fundstellen in einem separaten Fenster
- Druckfunktion
- Navigation im Dokument zur Auswahl der einzelnen Seiten
- In/Out-Zoom über Strg + Mausrad

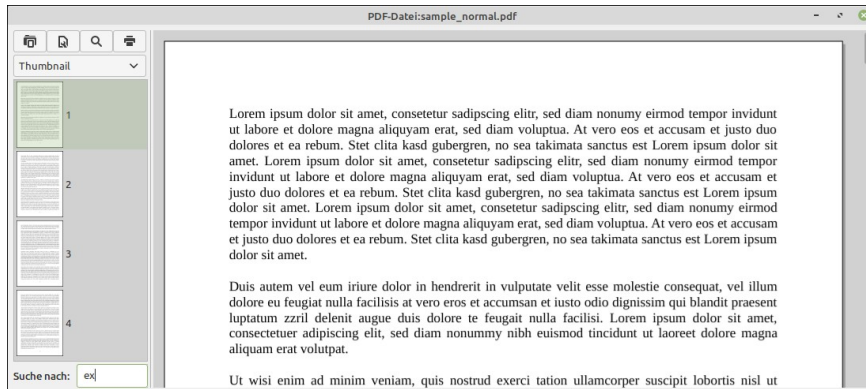


Abbildung 23.12.5.1: Programm-Oberfläche PDF-Reader mit Suchtext

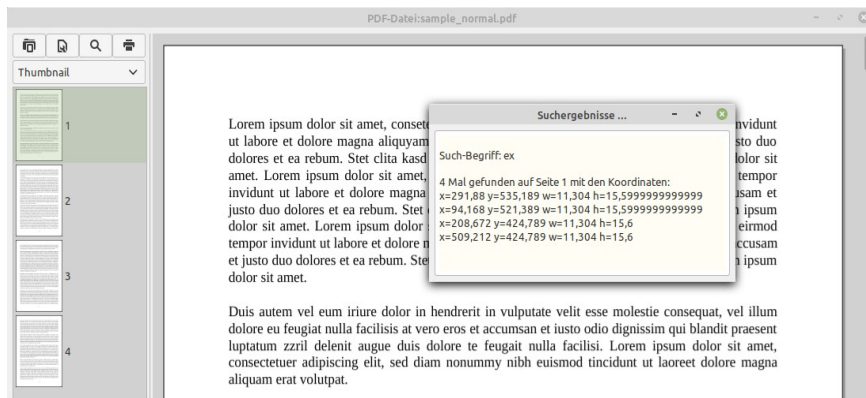


Abbildung 23.12.5.2: Programm-Oberfläche mit Anzeige der Fundstellen-Koordinaten

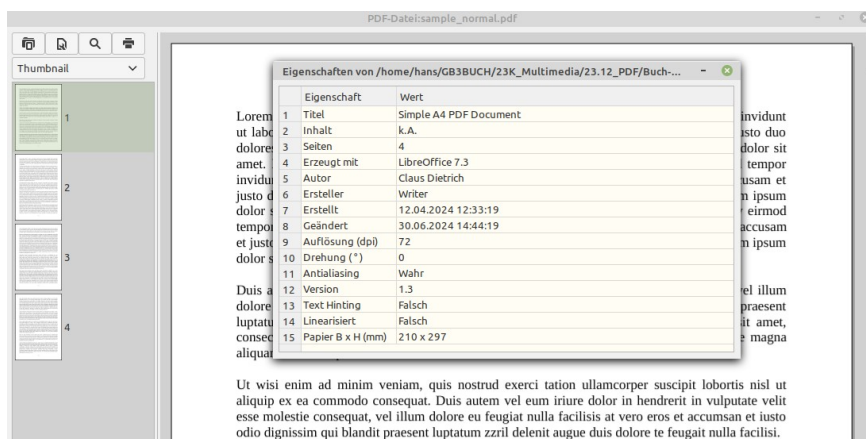


Abbildung 23.12.5.3: Programm-Oberfläche mit Anzeige von Eigenschaften des PDF-Dokumentes

Quelltext CreateThumbnails():

```

[1] Public Sub CreateThumbnails()
[2]
[3]     Dim hImage, hThumbnail As New Image
[4]     Dim i As Integer
[5]     Dim fFactor As Float = 0.10
[6]
[7]     ListViewPages.Expand = True
[8]     ListViewPages.Clear()
[9]
[10]    For i = 0 To hPdfDocument.Max
[11]        '-- Try reading embedded thumbnails first
[12]        hThumbnail = hPdfDocument[i].Thumbnail
[13]        '-- Make a thumbnail if no embedded thumbnail is available
[14]        If IsNull(hThumbnail) Then
[15]            hImage = hPdfDocument[i].Render(,,,,, 10)
[16]            hThumbnail = hImage.Stretch(hPdfDocument[i].W * fFactor, hPdfDocument[i].H * fFactor)
[17]        End If
[18]        Paint.Begin(hThumbnail)
[19]        Paint.LineWidth = 1
[20]        Paint.Brush = Paint.Color(color.Black)
[21]        Paint.Rectangle(0, 0, hThumbnail.W, hThumbnail.H)
[22]        Paint.Stroke()
[23]        Paint.End()
[24]
[25]        '-- Label instead of index+1, because the page labels of PDFs can be changed.
[26]        '-- XViewer also shows the page label.
[27]        ListViewPages.Add(i, hPdfDocument[i].Label, hThumbnail.Picture)
[28]    Next
[29]
[30]    ListViewPages[0].Selected = True
[31]
[32] End

```

Quelltext ShowIndex(Optional ExpandParent As Boolean = False)

```

[1] Public Sub ShowIndex(Optional ExpandParent As Boolean = False)
[2]
[3]     Dim hPdfIndex As PdfIndex
[4]     Dim i, iLastParent As Integer
[5]     Dim sIndexText As String
[6]
[7]     Dim pic, hpic As Picture
[8]
[9]     TreeViewIndex.Expand = True
[10]    TreeViewIndex.Clear()
[11]
[12]    cCollection.Clear()
[13]
[14]    '-- Load TreeView with table of contents
[15]    If hPdfDocument.Index.Count > 0 Then
[16]        ComboBoxSelectMode.Clear()
[17]        ComboBoxSelectMode.Add("Index & Thumbnail")
[18]        ComboBoxSelectMode.Add("Thumbnail")
[19]        ComboBoxSelectMode.Add("Index")
[20]        ComboBoxSelectMode.Index = 0
[21]        TreeViewIndex.Visible = True
[22]        For Each hPdfIndex In hPdfDocument.Index
[23]            sIndexText = Trim(hPdfIndex.Text)
[24]            hpic = Null
[25]            '-- Dummy picture (required to use the paint methods)
[26]            hpic = New Picture(1, 1)
[27]            Paint.Begin(hpic)
[28]            Paint.Font = TreeViewIndex.font
[29]            pic = Null
[30]            '-- Generate an invisible pic which enlarges the treeview line if required
[31]            pic = New Picture(1, Paint.TextSize(sIndexText).H, True)
[32]            Paint.End()
[33]
[34]            '-- Root item?
[35]            If hPdfIndex.Parent = -1 Then
[36]                TreeViewIndex.Add(i, sIndexText, pic)
[37]                iLastParent = i
[38]                '-- Expand the node
[39]                TreeViewIndex[i].Expanded = ExpandParent
[40]            '-- Non-Root parent item?
[41]            Else If hPdfIndex.Children > 0 Then
[42]                TreeViewIndex.Add(i, sIndexText, pic, hPdfIndex.Parent)
[43]                iLastParent = i
[44]                TreeViewIndex[i].Expanded = ExpandParent '-- Expand the node ...
[45]            Else '-- Child item?

```

```
[46]         TreeViewIndex.Add(i, sIndexText, pic, iLastParent)
[47]         Endif
[48]         '-- Put it into a collection to make the page accessible from the treeview
[49]         cCollection.Add(hPdfIndex.Action.Goto.Page, i)
[50]         Inc i
[51]     Next
[52] Else
[53]     ComboBoxSelectMode.Clear()
[54]     ComboBoxSelectMode.Add("Thumbnail")
[55]     ComboBoxSelectMode.Index = 0
[56]     TreeViewIndex.Visible = False
[57] Endif
[58]
[59] End
```

Die folgenden zwei Ereignisbehandlungsroutinen vervollständigen die Navigation und bieten die Möglichkeit, eine bestimmte PDF-Seite in der DocumentView anzuzeigen:

```
[1] Public Sub TreeViewIndex_Click()
[2]
[3]     DocumentView1.Goto(cCollection[TreeViewIndex.Current.Key] - 1)
[4]     iFirstVisiblePage = cCollection[TreeViewIndex.Current.Key] - 1
[5]
[6] End
[7]
[8] ...
[9]
[10] Public Sub ListViewPages_Select()
[11]
[12]     DocumentView1.Goto(ListViewPages.Current.Key)
[13]
[14] End
```

Im Download-Bereich finden Sie den Quelltext des PDF-Reader-Projekts als Quelltext-Archiv sowie Beispiel-PDF-Dateien.