

TYPO3

Anleitung

für

Menüstrukturen

Inhaltsverzeichnis

EINLEITUNG.....	3
ERSTELLUNG VON TEXTMENÜS	4
EINFACHE MENÜSTRUKTUREN ERSTELLEN.....	4
<i>Menüpunkte horizontal anzeigen</i>	4
<i>Menüpunkte vertikal anzeigen</i>	5
<i>Haupt- und Unternavigation anzeigen</i>	6
Alle Untermenüpunkte immer anzeigen	6
„onfocus=blurLink(this);“ beim Erstellen der Menüstruktur entfernen	7
<i>Haupt- und Unternavigation räumlich trennen</i>	8
<i>Breadcrumb Navigation erstellen</i>	9
<i>Footer Navigation erstellen</i>	9
Footer Navigation ohne Trennstrich am Ende (OptionSplit)	10
Eine Bereichüberschrift aus Menü erstellen.....	11
VERTIKALES TEXTMENÜ MIT LISTENFORMATIERUNG (TMENU).....	12
Formatklassen im Menü für CSS Formatierung festlegen	13
<i>Textmenü über zwei Ebenen mit Listenformatierung (TMENU)</i>	14
DROP-DOWN MENÜ ÜBER ZWEI EBENEN (TMENU)	15
ERSTELLUNG VON GRAPHISCHEN MENÜS	17
EINFACHES GRAPHISCHES MENÜS (GMENU)	17
<i>Vertikales GMENU erstellen</i>	17
Positionen und Abstände	18
Hintergrundgraphik in GMENU einfügen	18
<i>Rollover Effekt in GMENU einfügen</i>	19
<i>Aktive Seite im GMENU hervorheben</i>	20
Fehler angegebene Graphik für aktuelle Seite wird nicht angezeigt	21
Fehler „Unable to open file in...“ wenn jeder Menüpunkt das erste Mal angeklickt wird.....	21
<i>GMENU für 2 Ebenen mit Auswahlbalken</i>	21
<i>Horizontales GMENU erstellen</i>	23
KOMPLEXERE GRAPHISCHE MENÜS (GMENU)	24
<i>Schriftformatierung im GMENU vornehmen</i>	24
<i>Seitenbilder (Ressourcen) in Menüstruktur anzeigen</i>	25
ERSTELLUNG VON LAYER MENÜS.....	27
ERSTELLUNG EINES 2 STUFIGEN POP-UP MENÜ (GMENU_LAYERS)	27
<i>Erstellung 2. stufiges Pop-Up Menu mit Aktive und Rollover Effekt</i>	28
ERSTELLUNG EINES 2 STUFIGEN POP-UP MENÜ (TMENU_LAYERS).....	30
<i>Zweite Ebene von Pop-Up Menü als horizontales Menü</i>	30
Formatierung von „Active“ an „Rollover“ Eigenschaft durchführen.....	30
Einige Spielereien mit dem Pop-Up Menü	32
ERSTELLUNG EINES 3 STUFIGES POP-UP MENÜ (ÄHNLICH MS WINDOWS).....	33
KOMPLEXES 3 STUFIGES POP-UP MENÜ MIT FARBCODIERUNG.....	34
ERSTELLUNG VON JAVASCRIPT MENÜS.....	36
DROP-DOWN MENÜ MIT DEM JSMENU OBJEKT.....	36
Einzelnen Menüpunkt vom Menü entfernen.....	37
Mehrstufiges DropDown Menü	38
SPEZIELLE MENÜ KONSTRUKTIONEN	39
HORIZONTALES MENÜ WOBEI JEDER MENÜPUNKT EINE ANDERE FARBE HAT	39
MENÜ FÜR SPRACHUMSCHALTUNG MIT FLAGGEN	39
ZUSÄTZLICHE MENÜ MIT FARBCODIERUNG AUS „TT_NEWS_CAT“	40
KOMPLEXE MENÜS MIT GRAPHIKEN	41
ERSTELLE MENÜ MIT GIFBUILDER	41

Einleitung

Voraussetzungen zur Durcharbeitung dieser Anleitung sind, mindestens die TYPO3 Version 3.8.x und das Sie das Paket „Quickstart“ installiert haben.

Diese Anleitung setzt die Grundlagen von TYPO3 voraus, d.h. das Installieren von Extensions, die Anlage einer Seite, oder einen Record von Typ „plugin einfügen“, oder was ein TS (TypoScript) Template ist. Des weiterem wären Grundlagen von HTML und CSS von Vorteil. Die Details zu den einzelnen Parametern können sie im TYPO3 TSref Manual nachlesen.

Es wurden der Einfachheit halber keine weiteren Formatierungen über CSS Dateien vorgenommen, weil damit das komplette aussehen der Menüs nochmals geändert werden kann. In der Praxis wird die Menüerstellung aus einer Mischung von TypoScript und CSS Formatierungen erfolgen.

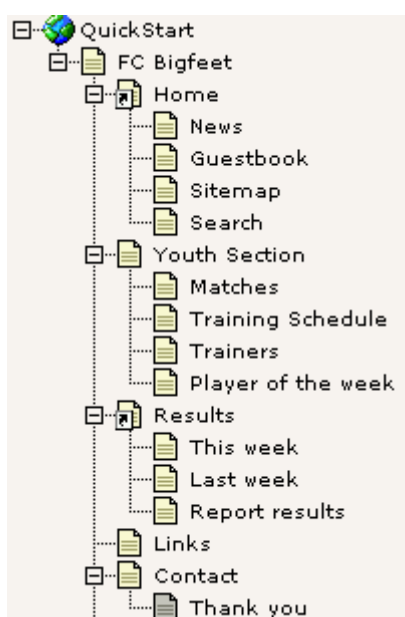
Damit die Struktur für die Menüerstellung nicht manuell angelegt werden muss, wurde einfach die Struktur aus dem Paket „Quickstart“ verwendet.

! WICHTIG !

Damit Sie die unten angeführten Beispiele innerhalb der „Quickstart“ Umgebung durchführen können, müssen Sie im Root Template „NEW SITE“ durch klicken auf dem Menüpunkt „Click here to edit the whole template record“ unter „Include static“ das „template; BUSINESS“ entfernen, wenn dieses vorhanden ist.

Entfernen Sie auch den vorgegeben Code aus dem Setup Bereich und ersetzen Sie diesen immer durch den Code in den Beispielen.

Die Ordnerstruktur von „Quickstart“



Erstellung von Textmenüs

Die Erstellung von Menüs erfolgt im TypoScript. Es ist dazu keine HTML Template Datei oder CSS Datei notwendig, sondern es kann alles in TypoScript formatiert werden.

Wenn Sie mit einer HTML Template Datei und einer CSS Datei („automaketemplate“ oder „templatevoila“) arbeiten können Sie Formatierung auch in einer CSS Datei vornehmen.

Einfache Menüstrukturen erstellen

Menüpunkte horizontal anzeigen

Einträge im Setup Bereich vom Template:

```
temp.menu = HMENU
temp.menu.1 = TMENU
temp.menu.1.NO = 1

page = PAGE
page.10 < temp.menu
```

Der Zustand NO ist der Normalzustand vom Menü. Durch die Zeile „...NO = 1“ im obigen Listing wird der Normalzustand vom Menü aktiviert, d.h. es wird das Menü angezeigt. Durch die letzten beiden Zeilen wird das erzeugte Menü an das Page Objekt zur Anzeige in der Webseite übergeben.

Anzeige im Browser:

[Home](#)[Youth](#)[Section](#)[Results](#)[Links](#)[Contact](#)

Wir sehen, dass die einzelnen Menüpunkte einfach hintereinander (ohne Abstände) angezeigt werden.

HTML Quelltext im Browser:

```
<body bgcolor="#FFFFFF">
<a href="index.php?id=6" onfocus="blurLink(this);">Home</a><a href="index.php?id=5"
onfocus="blurLink(this);">Youth Section</a><a href="index.php?id=4" onfocus="blurLink(this);">Results</a><a
href="index.php?id=3" onfocus="blurLink(this);">Links</a><a href="index.php?id=2"
onfocus="blurLink(this);">Contact</a>
</body>
```

Wir sehen im obigen Quelltext das TYPO3 beim wrappen der Menüstruktur automatisch den Eintrag „onfocus=“blurLink(this);“ in jedem Ankerlink hinzufügt.

Menüpunkte vertikal anzeigen

Wir verändern den Code im Setup Bereich folgenderweise ab:

```
temp.menu = HMENU
temp.menu.1 = TMENU
temp.menu.1.NO.linkWrap = | <br> // jetzt braucht NO nicht mehr aktiviert werden

page = PAGE
page.10 < temp.menu
```

Durch die Angabe von Tag
 wird jetzt nach jeden Menüpunkt ein Zeilenumbruch durchgeführt. Das Zeichen „|“ ist dabei ein Platzhalter, der beim wrappen durch die einzelnen Menüpunkte ersetzt wird.

Ansicht im Browser:



[Home](#)
[Youth Section](#)
[Results](#)
[Links](#)
[Contact](#)

HTML Quelltext im Browser:

```
<body bgcolor="#FFFFFF">
<a href="index.php?id=6" onfocus="blurLink(this);">Home</a> <br><a href="index.php?id=5"
onfocus="blurLink(this);">Youth Section</a> <br><a href="index.php?id=4"
onfocus="blurLink(this);">Results</a> <br><a href="index.php?id=3" onfocus="blurLink(this);">Links</a>
<br><a href="index.php?id=2" onfocus="blurLink(this);">Contact</a> <br>
</body>
```

Haupt- und Unternavigation anzeigen

Wir verändern den Code im Setup Bereich folgenderweise ab:

```
temp.menu = HMENU
temp.menu.1 = TMENU
temp.menu.1.NO.linkWrap = | <br>

temp.menu.2 = TMENU
temp.menu.2.NO.linkWrap = &nbsp;&nbsp;&nbsp;| <br>

page = PAGE
page.10 < temp.menu
```

Es wird mit „2“ ein zusätzliches Menü definiert. Das Wrapping wurde verändert indem vor der Anzeige des Menüpunktes zwei Leerschritte („ “) eingefügt wurden.

Anzeige im Browser:



```
Home
Youth Section
  Matches
  Training Schedule
  Trainers
  Player of the week
Results
Links
Contact
```

Es werden jetzt, wenn z.B. der Menüpunkt „Youth Section“ angeklickt wird, die Untermenüpunkte (Seiten die sich unterhalb befinden) vom „Youth Section“-Ordner angezeigt.

Alle Untermenüpunkte immer anzeigen

Durch folgende Änderung im Code des Setup Bereichs kann die Anzeige aller Untermenüpunkte permanent erzwungen werden:

```
temp.menu = HMENU
temp.menu.1 = TMENU
temp.menu.1.expAll = 1
temp.menu.1.NO.linkWrap = | <br>

temp.menu.2 = TMENU
temp.menu.2.NO.linkWrap = &nbsp;&nbsp;&nbsp;| <br>

page = PAGE
page.10 < temp.menu
```

Anzeige im Browser:

```
Home  
News  
Guestbook  
Sitemap  
Search  
Youth Section  
Matches  
Training Schedule  
Trainers  
Player of the week  
Results  
This week  
Last week  
Report results  
Links  
Contact
```

„onfocus=blurLink(this);“ beim Erstellen der Menüstruktur entfernen

Durch folgende Änderung im Code des Setup Bereich kann der Eintrag der „onfocus=blurLink(this);“ Definition beim Erzeugen der Menüs verhindert werden:

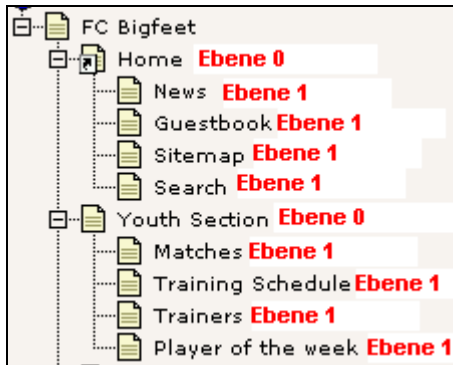
```
temp.menu = HMENU  
temp.menu.1 = TMENU  
temp.menu.1.noBlur = 1  
temp.menu.1.NO.linkWrap = | <br>  
  
temp.menu.2 = TMENU  
temp.menu.2.NO.linkWrap = &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;| <br>  
  
page = PAGE  
page.10 < temp.menu
```

HTML Quelltext ohne „onfocus=blurLink(this);“ im Browser:

```
<body bgcolor="#FFFFFF">  
<a href="index.php?id=6">Home</a> <br><a href="index.php?id=5">Youth Section</a> <br><a  
href="index.php?id=4">Results</a> <br><a href="index.php?id=3">Links</a> <br><a  
href="index.php?id=2">Contact</a> <br>  
</body>
```

Haupt- und Unternavigation räumlich trennen

Um die Haupt- und Unternavigation räumlich zu trennen, muss jeweils für die Haupt- und Unternavigation ein eigenes Objekt angelegt werden. Zusätzlich muss für das Objekt der Unternavigation ein „entryLevel“ festgelegt werden, ab dem diese Navigation starten soll.



In unserem Beispiel befindet sich die erste Unterseite „News“ die angezeigt werden soll auf Ebene 1. Daher wird „1“ als Startpunkt für die Unternavigation im „entryLevel“ eingetragen.

Wir Erstellen dazu folgenden Code im Setup Bereich:

```
temp.menu = HMENU
temp.menu.1 = TMENU
temp.menu.1.noBlur = 1
temp.menu.1.NO.linkWrap = | <br>

temp.submenu = HMENU
temp.submenu.entryLevel = 1
temp.submenu.1 = TMENU
temp.submenu.1.NO.linkWrap = &nbsp;&nbsp;&nbsp;| <br>

page = PAGE
page.10 < temp.menu
page.20 < temp.submenu
```

Ansicht im Browser:



Die Unternavigation von „Youth Section“ wird jetzt unterhalb der Hauptnavigation angezeigt. Sie könnte jetzt auch jederzeit an einer anderen Stelle durch Marker platziert werden.

Breadcrumb Navigation erstellen

Eine so genannte Breadcrumb Navigation zeigt immer den kompletten Pfad, wo sich die aktuelle Seite befindet, und kann als Navigationshilfe verwendet werden.

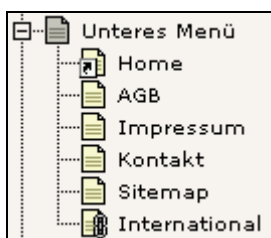
Im Setup Bereich wird eine Footer Navigation folgenderweise erstellt:

```
temp.breadcrumb = HMENU
temp.breadcrumb.special = rootline
temp.breadcrumb.special.value = 1
temp.breadcrumb.1 = TMENU
temp.breadcrumb.1.NO {
  noBlur = 1
  linkWrap = | &nbsp;&gt;&nbsp;&nbsp;|*|
  ATagParams = style="text-decoration:none;"
}
temp.breadcrumb.1.CUR = 1
temp.breadcrumb.1.CUR {
  linkWrap = <strong>|</strong>
  doNotLinkIt = 1
}

page = PAGE
page.30 < temp. breadcrumb
```

Footer Navigation erstellen

Eine so genannte Footer Navigation befindet sich meist am Ende einer Seite. Diese beinhaltet Verknüpfungen für „Impressum“, „AGB“, „Kontakt“ usw. Diese Menüpunkte sollten von jeder Seite aus erreichbar sein, aber nicht in der Hauptnavigation aufscheinen. Um eine solche Navigation zu erstellen wird eine Seite „Nicht im Menü“ verwendet, unter der alle diese Seiten angelegt werden.



Im Setup Bereich wird eine Footer Navigation folgenderweise erstellt:

```
temp.footer_menu = HMENU
temp.footer_menu.special = directory
temp.footer_menu.special.value = 1
temp.footer_menu.1 = TMENU
temp.footer_menu.1.NO {
  noBlur = 1
  linkWrap = | &#124;&nbsp;&nbsp;
  ATagParams = style="text-decoration:none;"
}

page = PAGE
page.30 < temp.footer_menu
```

Nur als Beispiel wird im obigen Code ein Menü aus allen Unterseiten der Seite „1“ erzeugt, normalerweise muss anstatt „1“ die Nummer des Ordners indem sich die Unterseiten befinden eingegeben werden.

Die einzelnen Menüpunkte werden horizontal dargestellt und durch „|“ getrennt. Das Trennzeichen wurde als ASCII Wert mit „|“ festgelegt.

Der Eintrag „...special = directory“ stellt eine spezielle Art der Menüerstellung dar. Durch Angabe vom „...special.value = 1“ wird der Ausgangspunkt für die Menüerstellung festgelegt. Zusätzlich wurde im obigen Beispiel durch den „ATagParams = style=“text-decoration“:none;“ festgelegt (direkte Style-Formatierung ohne CSS), dass die Links ohne den Unterstrich dargestellt werden.

Anzeige im Browser:

```
Home | Youth Section | Results | Links | Contact | Products |
```

Footer Navigation ohne Trennstrich am Ende (OptionSplit)

Damit am Ende der vorher erstellten Footer Navigation nicht nochmals ein Trennstrich angezeigt wird, muss eine spezielle Art der Formatierung die sich „OptionSplit“ nennt angewendet werden.

Mit „OptionSplit“ ist es möglich den einzelnen Menüpunkten der Navigation unterschiedliche Formatierungen zuzuweisen.

Unterschieden wird zwischen:

erste Ausgabe |*| alle mittleren Ausgaben |*| letzte Ausgabe

Im Setup Bereich wird folgendes für Footer-Navigation geändert:

```
temp.footer_menu = HMENU
temp.footer_menu.special = directory
temp.footer_menu.special.value = 1
temp.footer_menu.1 = TMENU
temp.footer_menu.1.NO {
  noBlur = 1
  linkWrap = | &#124;&nbsp;|*|*| |
  ATagParams = style="text-decoration:none;"
}

page = PAGE
page.30 < temp.footer_menu
```

Durch den ersten Teil der OptionSplit Angabe „| | |*|*|“ wird festgelegt, dass nach dem erstem Menüpunkt („|“ Ersetzungszeichen) ein Trennstrich angezeigt wird. Durch die anschließende OptionSplit Option „|*|“ (ohne Inhalt) wird festgelegt, dass auch alle mittleren Menüpunkte die Formatierung der ersten Menüoption erhalten. Die Angabe vom „|“ (Ersetzungszeichen) nach der zweiten OptionSplit Option besagt, dass der letzte Menüpunkt ohne besondere Formatierung angezeigt wird.

Anzeige im Browser:

```
Home | Youth Section | Results | Links | Contact | Products
```

Eine Bereichüberschrift aus Menü erstellen

Wenn ein bestimmter Menüpunkt ausgewählt wird, soll in einen anderen Bereich diese Auswahl als Bereichüberschrift angezeigt werden.

Mit folgendem Code im Setup Bereich lässt sich dies einfach erstellen:

```
temp.menu = HMENU
temp.menu.1 = TMENU
temp.menu.1.noBlur = 1
temp.menu.1.NO.linkWrap = | <br>

temp.bereich = HMENU
temp.bereich.special = rootline
temp.bereich.special.range = 1|1
temp.bereich.1 = TMENU
temp.bereich.1 {
  NO.linkWrap = <br><b>|</b><br>
  NO.doNotLinkIt = 1
}

page = PAGE
page.10 < temp.menu
page.20 < temp.bereich
```

Durch die Angabe „...special = rootline“ wird eine Navigation festgelegt die den kompletten Pfad (von Root ausgehend) des ausgewählten Menüpunktes enthält. Durch die Angabe von „...special.range = 1|1“ wird der Bericht der Pfadverfolgung eingegrenzt, d.h. es wird nur der Menüpfad nur bis zur ersten Ebene verfolgt. Eine Bereichüberschrift wird in diesem Beispiel immer aus den Seiten der ersten Ebene erzeugt und nie aus Seiten der Unterebenen. Durch den Eintrag „NO.doNotLinkIt = 1“ wird festgelegt, das dieser Menüpunkt nicht verlinkt werden soll.

Ansicht im Browser:



Bei Auswahl von „Youth Section“ wird unterhalb vom Hauptmenü der Name dieser Seite „Youth Section“ nochmals in Fett angezeigt.

Vertikales Textmenü mit Listenformatierung (TMENU)

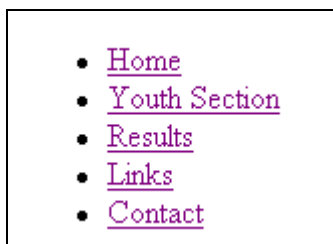
Um barrierefreie und standardkonforme Navigationen zu erstellen wird zur Formatierung der Navigation oft die HTML Listenformatierung (mit `` und `` Tags) verwendet.

Einträge im Setup Bereich vom Template:

```
temp.menu_main = HMENU
# set the entry level for the menu
temp.menu_main.entryLevel = 0
# First level menu-object, textual
temp.menu_main.1 = TMENU
temp.menu_main.1 {
  noBlur = 1
  expAll = 0
  # Normal state properties
  wrap = <ul> | </ul>
  NO.allWrap = <li> | </li>
  # Enabel active state and set properties
  ACT = 1
  ACT.allWrap = <li> | </li>
  CUR = 1
  CUR.allWrap = <li> | </li>
}
page = PAGE
page.10 < temp.menu_main
```

Hinweis In einem TMENU können Sie Zuständen „ACT“, „CUR“ keine direkten Formatierungen zuweisen. Erstellen Sie für diese Zustände eigene Klassen und formatieren Sie das Menü über eine CSS Datei. Der Zustand „RO“ tritt in einem TMENU nicht auf und eine Formatierung wird über den Zustand „NO“ oder der Ankerklasse „hover“ zugewiesen.

Ansicht im Browser:



Da die einzelnen Menüpunkte als Aufzählungsliste formatiert wurden, werde diese mit einen Aufzählungssymbol angezeigt.

HTML Quelltext im Browser:

```
<body bgcolor="#FFFFFF">
<ul><li><a href="index.php?id=6">Home</a></li><li><a href="index.php?id=5">Youth Section</a></li><li><a
href="index.php?id=4">Results</a></li><li><a href="index.php?id=3">Links</a></li><li><a
href="index.php?id=2">Contact</a></li></ul>
</body>
```

Formatklassen im Menü für CSS Formatierung festlegen

Bei einem TMENU können die einzelnen Menüeinträge auch über CSS formatiert werden. Sie müssen dazu für die einzelnen Menüzustände Klassen definieren die über eine CSS-Datei formatiert werden.

Um das Aussehen der Aufzählungsliste für das Menü verändern, definieren wir alle Zustände des Menüs eigene CSS Klassen in welchen die Formatierung festgelegt wird.

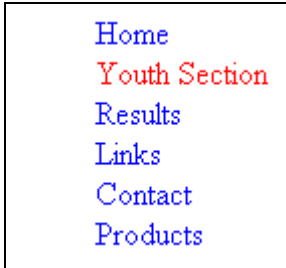
```
temp.menu_main = HMENU
# set the entry level for the menu
temp.menu_main.entryLevel = 0
# First level menu-object, textual
temp.menu_main.1 = TMENU
temp.menu_main.1 {
  noBlur = 1
  expAll = 0
  # Normal state properties
  wrap = <ul class="normal_list"> | </ul>
  NO.allWrap = <li> | </li>
  NO.ATagParams = class="normal_link"
  # Enabel active state and set properties
  ACT = 1
  ACT.allWrap = <li> | </li>
  ACT.ATagParams = class="active_link"
  CUR = 1
  CUR.allWrap = <li> | </li>
  CUR.ATagParams = class="active_link"
}
page = PAGE
page.10 < temp.menu_main
```

Definitionen in der CSS Datei für das obige Beispiel

```
...
.normal_list {
  list-style: none;
}
.normal_link:link, .normal_link:visited {
  text-decoration: none;
  color: #00f;
  background-color: fffff;
}
.normal_link:hover {
  text-decoration: none;
  color: #0f0;
  background-color: fffff;
}
.active_link {
  text-decoration: none;
  color: #f00;
  background-color: fffff;
}
...
```

Durch die obige CSS Definition werden die Listensymbole entfernt. Bei den Links wird der Unterstrich sowie die Farbe für „visited“ auf blau eingestellt. Aktive Links werden mit der Farbe rot dargestellt. Der Mouseover wird in grüner Farbe angezeigt. Die Hintergrundfarbe wurde bei allen Zuständen auf weiß eingestellt.

Anzeige im Browser:

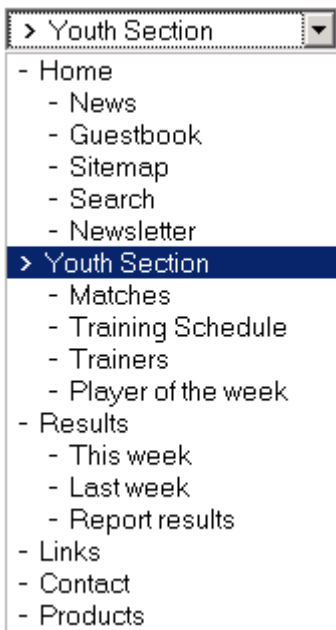


Textmenü über zwei Ebenen mit Listenformatierung (TMENU)

```
temp.menu_main = HMENU
# set the entry level for the menu
temp.menu_main {
  entryLevel = 0
  # First level menu-object, textual
  1 = TMENU
  1 {
    wrap = <ul class="UL_CLASS">|</ul>
    NO = 1
    NO {
      before {
        dataWrap = <li class="FESTE_CLASS_FUER_LI1" id="li_special_{field:uid}">|
      }
      wrapItemAndSub = |</li>
    }
  }
  2 < .1
  2 {
    wrap = <ul class="UL_CLASS_2">|</ul>
    NO {
      before {
        dataWrap = <li class="FESTE_CLASS_FUER_LI2" id="li_special_{field:uid}">|
      }
    }
  }
}
page = PAGE
page.10 < temp.menu_main
```

Damit können Sie auch hier mit Klassen arbeiten. Sie haben eine feste Klasse für alle Menüpunkte sowie für die Menüebenen und können über die dynamische ID jeden einzelnen Menüpunkt ansprechen.

Anzeige des Drop-Down Menüs im Browser:



Erstellung von graphischen Menüs

Einfaches graphisches Menüs (GMENU)

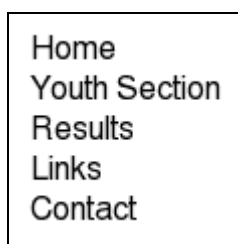
Vertikales GMENU erstellen

Bei dieser Menüart werden die einzelnen Auswahlpunkte beim Erzeugen des Menüs durch TYPO3 als Graphiken dargestellt. Diese Graphiken werden von TYPO3 anhand der Definitionen dynamisch (über GIFBUILDER) generiert. Zur Erstellung solcher Menüs mit dem GIDBUILDER wird das Grafikprogramm GDLib und FreeType verwendet, welche bei der Installation aktiviert sein sollten.

Mit folgendem Code wird ein einfaches GMENU erzeugt:

```
temp.menu = HMENU
temp.menu.1 = GMENU
temp.menu.1.noBlur = 1
temp.menu.1.NO {
  wrap = | <br>
  XY = 200,20
  backColor = white
  10 = TEXT
  10.text.field = title
  10.offset = 10,14
}
page = PAGE
page.10 < temp.menu
```

Ansicht im Browser:



HTML Quelltext im Browser:

```
<body bgcolor="#FFFFFF">
<a href="index.php?id=6"></a><br><a href="index.php?id=5"></a><br><a href="index.php?id=4"></a><br><a
href="index.php?id=3"></a><br><a href="index.php?id=2"></a><br>
</body>
```

Positionen und Abstände

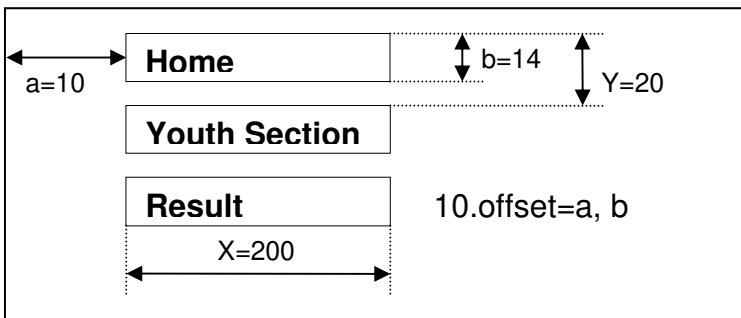
Die Positionen und Abstände, in dem oben erstellten Menü, werden über folgende Parameter festgelegt.

XY = 200,20

und

10.offset = 10,14

In einer Grafik kann dies wie folgt dargestellt werden:



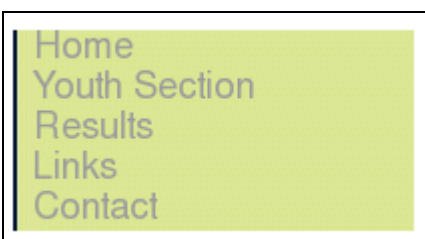
Hintergrundgraphik in GMENU einfügen

Hinter die einzelnen Menüpunkte kann eine Hintergrundgrafik (Bild) hinterlegt werden, dass bei jedem Menüpunkt angezeigt wird.

Folgende Änderung im Code des Setup Bereiches vornehmen:

```
temp.menu = HMENU
temp.menu.1 = GMENU
temp.menu.1.noBlur = 1
temp.menu.1.NO {
  wrap = | <br>
  XY = 200,20
  5 = IMAGE
  5.file = beckback*.jpg
  10 = TEXT
  10.text.field = title
  10.offset = 10,14
  10.fontSize = 14
  10.fontColor = #9B9B9B
}
page = PAGE
page.10 < temp.menu
```

Ansicht im Browser:



Sollte das hinterlegte Bild größer sein, als der Anzeigebereich eines Menüpunktes, so wird nur ein Teil des Bildes von oben und links beginnend ausgeschnitten und angezeigt.

Rollover Effekt in GMENU einfügen

Wenn Sie sich mit der Maus über einen Menüpunkt bewegen, soll die jeweilige Graphik des Menüpunktes ausgetauscht werden.

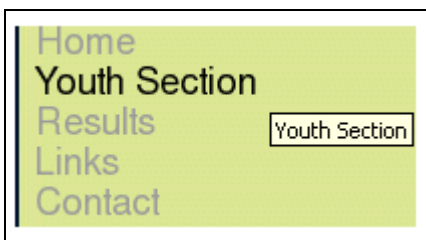
Änderungen im Setup Bereich für Rollover Effekt:

```
temp.menu = HMENU
temp.menu.1 = GMENU
temp.menu.1.noBlur = 1
temp.menu.1.NO {
  wrap = | <br>
  XY = 200,20
  5 = IMAGE
  5.file = beckback*.jpg
  # backColor = white
  10 = TEXT
  10.text.field = title
  10.offset = 10,14
  10.fontSize = 14
  10.fontColor = #9B9B9B
}
temp.menu.1.RO < temp.menu.1.NO
temp.menu.1.RO.10.fontColor = black
temp.menu.1.RO = 1

page = PAGE
page.10 < temp.menu
```

Löschen Sie den Frontend Cache bevor Sie die Einstellung testen.

Ansicht im Browser:



Wenn man sich jetzt mit der Maus über die einzelnen Menüpunkte bewegt wird die Textfarbe des Menüpunktes schwarz.

Aktive Seite im GMENU hervorheben

Die ausgewählte Seite im graphischen Menü soll besonders hervorgehoben werden. Kopieren Sie aus dem TYPO3 Verzeichnis „media/uploads/re/“ die Datei „menuback.gif“ in Ihr Image Verzeichnis kopieren z.B. das Verzeichnis „fileadmin/images/“. Sollte das Verzeichnis „images“ nicht vorhanden sein, so legen Sie dieses an.

Folgende Änderung im Code des Setup Bereiches durchführen:

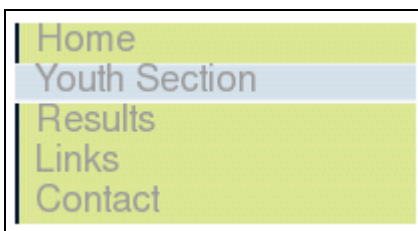
```
temp.menu = HMENU
temp.menu.1 = GMENU
temp.menu.1.noBlur = 1
temp.menu.1.NO {
  wrap = | <br>
  XY = 200,20
  5 = IMAGE
  5.file = beckback*.jpg
  10 = TEXT
  10.text.field = title
  10.offset = 10,14
  10.fontSize = 14
  10.fontColor = #9B9B9B
}
temp.menu.1.RO < temp.menu.1.NO
temp.menu.1.RO.10.fontColor = black
temp.menu.1.RO = 1

temp.menu.1.CUR < temp.menu.1.NO
temp.menu.1.CUR.5.file = fileadmin/images/menuback.gif
temp.menu.1.CUR = 1

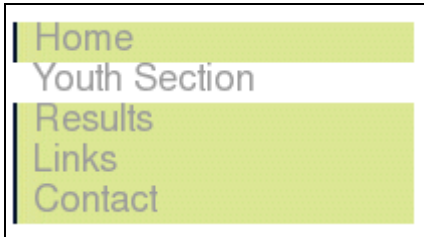
page = PAGE
page.10 < temp.menu
```

Löschen Sie den Frontend Cache bevor Sie die Einstellung testen.

Ansicht im Browser mit richtiger Graphik für aktive Seite:



Beim Menüpunkt „Home“ und „Result“ passiert nichts, weil diese Seiten „Shortcut“ auf andere Seiten darstellen, und somit selbst nicht die aktiven Seiten sind.

Fehler angegebene Graphik für aktuelle Seite wird nicht angezeigt

Das Bild welches für die aktuelle Seite hinterlegt wurde wird nicht angezeigt, sondern ein weißer Hintergrund. Diese Fehler könnte in Zusammenhang mit den nachfolgenden Fehler oder auch alleine auftreten. In unserem Fall wurde der Fehler verursacht, weil eine „*.png“ Graphik ausgewählt wurde, und die Unterstützung für PNG nicht in TYPO3 installiert ist.

Fehler „Unable to open file in....“ wenn jeder Menüpunkt das erste Mal angeklickt wird

Warning: md5_file() [[function.md5-file](#)]: Unable to open file in `/opt/lampp/htdocs/typo3_src-3.8.0/t3lib/class.t3lib_stdgraphic.php` on line 2160

Warning: Cannot modify header information - headers already sent by (output started at `/opt/lampp/htdocs/typo3_src-3.8.0/t3lib/class.t3lib_stdgraphic.php:2160`) in `/opt/lampp/htdocs/typo3_src-3.8.0/typo3/sysex/cms/tslib/class.tslib_fe.php` on line 2612

Abhilfe:

Die Definition für die Graphik die für die aktuelle Seite angezeigt werden soll ist falsch oder die Graphik ist nicht vorhanden, oder das Zugriffsrecht ist falsch. In unseren Fall war das Zugriffsrecht auf den Ordner „images“ und auf die Grafikdatei falsch.

GMENU für 2 Ebenen mit Auswahlbalken

Es wird ein Menü für 2 Ebene erstellt, wobei eine eigene Schriftart und ein Rollover Effekt integriert wurde. Durch den „expAll = 1“ wurde festgelegt das die zwei Ebenen automatisch geöffnet werden. Die Font Datei „arial.ttf“ können Sie sich von einer externen Quelle besorgen und wie hier in das Verzeichnis „fileadmin/fonts/“ kopieren. Sollte das Verzeichnis „fonts“ nicht vorhanden sein, legen Sie dieses an.

Folgender TypoScript Code wurde im Setup Bereiches angelegt:

```
temp.menu = HMENU
temp.menu {
  1 = GMENU
  1 {
    expAll = 1
    noBlur = 1
    NO {
      wrap = |<br>
      XY = 150,13
      backColor = #eeee88
      10 = TEXT
      10.text.field = title
      10.offset = 10,12
      10.fontFile = fileadmin/fonts/arial.ttf
      10.fontSize = 12
      10.fontColor = #101033
      10.align = left
    }
  }
  RO < .NO
```

```

RO = 1
RO.backColor = #dddddd
RO.10.fontColor = #666666
CUR < .NO
CUR = 1
CUR.backColor = #dddddd
CUR.10.fontColor = #666666
}

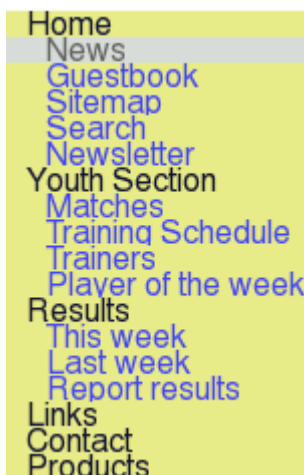
2 = GMENU
2 {
  noBlur = 1
  NO {
    wrap = |<br>
    XY = 150,13
    backColor = #eeee88
    10 = TEXT
    10.text.field = title
    10.offset = 20,12
    10.fontFile = fileadmin/styles/arial.ttf
    10.fontSize = 12
    10.fontColor = #4444ff
    10.align = left
  }
  RO < .NO
  RO = 1
  RO.backColor = #dddddd
  RO.10.fontColor = #666666
  CUR < .NO
  CUR = 1
  CUR.backColor = #dddddd
  CUR.10.fontColor = #666666
}
}

# Erstellung des PAGE Objektes und Übergabe der vorher erstellten „temp“ Objekte
# des Dokumentes „temp.mainTemplate“ und „temp.headTemplate“
page = PAGE
page.10 < temp.menu
page.10 < temp.menu

```

Löschen Sie den Frontend Cache bevor Sie die Einstellung testen.

Ansicht im Browser mit Auswahl einer für aktive Seite:



Horizontales GMENU erstellen

Bei einem horizontalen GMENU müssen die Graphiken je nach Textlänge der Menüpunkte verschieden lang erzeugt werden.

Geben Sie folgenden Code im Setup Bereich des Templates ein:

```
1. temp.menu = HMENU
2. temp.menu.1 = GMENU
3. temp.menu.1.noBlur = 1
4. temp.menu.1.NO {
5. wrap = &nbsp; | &nbsp;
6. XY = [10.w]+10,24
7. backColor = #009900
8. 10 = TEXT
9. 10.text.field = title
10. 10.offset = 5,16
11. 10.fontSize = 14
12. 10.fontColor = white
13. }
14. temp.menu.1.RO < temp.menu.1.NO
15. temp.menu.1.RO.10.fontColor = black
16. temp.menu.1.RO = 1

17. page = PAGE

18. page.10 < temp.menu
```

In der Zeile 14. wird dem „Rollover“ (RO) die Definition vom „Normal“ (NO) Zustand übergeben.

In der Zeile 15 wird die Font Farbe für RO festgelegt. Diese Definition überschreibt Font Farbe für den NO Zustand.

In der Zeile 16 wird das RO Objekt aktiviert. Die Aktivierung muss immer die letzte Anweisung nach der Definition des Objektes sein.

Ansicht im Browser:



Wenn man sich mit der Maus über einen Menüpunkt bewegt wird die Schrift des Menüpunktes schwarz. Eine eigene Hintergrundgraphik für die aktuelle Seite wurde in diesem Beispiel nicht hinterlegt.

Komplexere graphische Menüs (GMENU)

Schriftformatierung im GMENU vornehmen

Sie können die Schrift für das Menü über die GIFBUILDER Optionen mit allen möglichen Formatierungen versehen.

Eingaben im Setup Bereich:

```
temp.menu = HMENU
temp.menu.1 = GMENU
temp.menu.1.noBlur = 1
temp.menu.1 {
  NO {
    wrap = | <br>
    XY = 200,40
    backColor = black
    10 = TEXT
    10 {
      text.field = title
      offset = 1,35
      fontFile = fileadmin/fonts/verdana.ttf
      fontSize = 16
      fontColor = #ff0000
      align = center
      shadow {
        offset = 1,1
        color = yellow
        blur = 90
        intensity = 80
      }
    }
  }
}

RO < .NO
RO = 1
RO.10.fontColor = #008000
RO.10.shadow {
  offset = 1,1
  color = white
  blur = 90
  intensity = 80
}

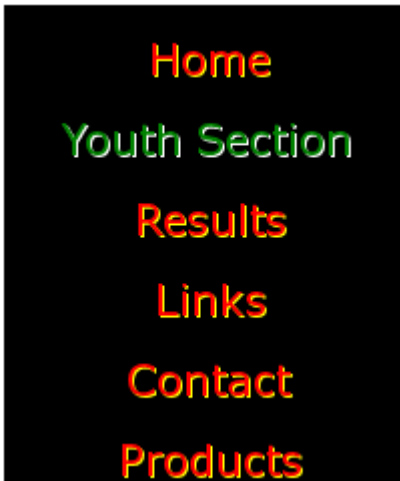
CUR < .NO
CUR = 1
CUR.doNotLinkIt = 1
CUR.10.fontColor = #008000
CUR.10.shadow {
  offset = 1,1
  color = white
  blur = 90
  intensity = 80
}
}

page = PAGE
page.10 < temp.menu
```


Im obigen Beispiel muss das Verzeichnis „font“ vorhanden sein und in diesem die Schrift „verdana.ttf“ abgelegt werden.

Durch die GIFBUILDER Option „shadow“ wird auf die definierte Menüschriftart ein zusätzlicher Schatteneffekt für den Normal- und Rollover-Zustand formatiert.

Anzeige im Browser:



Seitenbilder (Ressourcen) in Menüstruktur anzeigen

In diesem Beispiel werden die Bilder der einzelnen Menüpunkte auf die die einzelnen Seiten als Ressourcen hochgeladen. Durch das Hochladen werden die Bilder unter „/uploads/media/“ abgelegt und können von dort dann in der Menüstruktur angezeigt.

Das erste Bild das auf der Seite unter Ressourcen hochgeladen wird besitzt die „listNum=0“, das zweite Bild die „listNum=1“, usw.

Im normalen Menü wird hier das erste Bild aus dem Feld „Ressourcen“ als Menübild geladen, bei Maus-Rollover und bei aktiven Menüpunkt wird das zweite Bild aus dem Ressourcen Feld angezeigt.

Eingaben im Setup Bereich:

```
...
# The image menu with ressource images:
temp.menu = HMENU
temp.menu.1 = GMENU
temp.menu.1 {
    wrap = <ul>|</ul>
    noBlur = 0
    NO = 1
    NO {
        wrap = <li>|</li>
        # define normal image 'listNum=0'
        altImgResource.import = uploads/media/
        altImgResource.import.field = media
        altImgResource.import.listNum = 0

        ATagTitle.field = subtitle // title
    }
    # define alternative image 'listNum=1' for rollover and active menu
    RO < .NO
    RO = 1
}
```

```
altImgResource.import.listNum = 1  
ACT < .RO
```

```
}
```

```
page = PAGE  
page.10 < temp.menu
```

Durch die Angabe der "altImgResource" wird der das Bild das auf diese Seite geladen wurde im Menü als Menügrafik oder auch als Hintergrundgraphik angezeigt. Wenn Sie die Bilder als Menügrafik anzeigen, können Sie die Zeile „ATagTitle.field“ entfernen.

Dies ist eine alternative zur Anzeige von Menüs mit Bilder mit einer Fallback-Schleife wenn keine Bilder vorhanden sind.

```
# The menu:  
temp.customImageMenu = HMENU  
temp.customImageMenu.1 = GMENU  
temp.customImageMenu.1 {  
  noBlur = 0  
  NO = 1  
  NO {  
    altImgResource.import = uploads/media/  
    altImgResource.import.field = media  
    altImgResource.import.listNum = 0  
  
    # fall back GIFBUILDER item, if no result from altImgResource:  
    XY = 226,20  
    backColor = silver  
    10 = TEXT  
    10.text.field = nav_title // title  
    10.offset = 10,13  
    # wrap  
    wrap = |  
  }  
  RO < .NO  
  RO = 1  
  altImgResource.import.listNum = 1  
}  
  
page = PAGE  
page.10 < temp.menu
```

Erstellung von Layer Menüs

Erstellung eines 2 stufigen Pop-Up Menü (GMENU_LAYERS)

Es wird hier ein zweistufiges Pop-Up Menü angelegt. Wobei die erste Ebene als horizontales Menü und die zweite Ebene als vertikales GMENU ausgeführt wird.

Folgenden Code im Setup Bereich eingeben:

```

1. page = PAGE
2. page.includeLibs.gmenu_layers = media/scripts/gmenu_layers.php
3. page.10 = HMENU
4. page.10.1 = GMENU_LAYERS
5. page.10.1 {
6.   layerStyle = position:absolute;left:0px;top:30px;width:10px;VISIBILITY:hidden;
7.   xPosOffset = -10
8.   lockPosition = x
9.   topOffset = 36
10.  expAll=1
11.  NO {
12.    backColor = #cccccc
13.    XY = [10.w]+10,20
14.    10 = TEXT
15.    10.text.field = title
16.    10.offset = 5,15
17.    10.fontFace = fileadmin/verdana.ttf
18.    10.fontSize = 12
19.  }
20. }
21. page.10.2 = GMENU
22. page.10.2.wrap = |<br>
23. page.10.2.NO {
24.   backColor = #99cccc
25.   XY = 120,15
26.   10 = TEXT
27.   10.text.field = title
28.   10.offset = 5,12
29.   10.fontFace = fileadmin/verdana.ttf
30.   10.fontSize = 10
31. }

```

In dem obigen Beispiel wurden bewusst die „Rollover“ und „Aktive“ Eigenschaften der Menüs weggelassen um die Struktur relativ einfach zu halten.

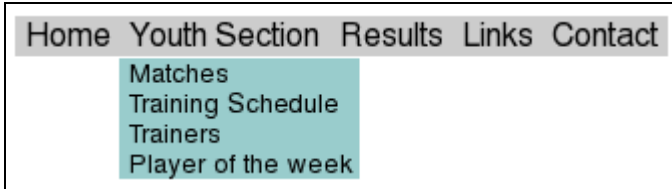
Der Eintrag in der Zeile 6. legt die horizontale Position des Pop-Up der zweiten Ebene fest. Diese Position ist abhängig vom selektierten Menüpunkt der ersten Ebene.

Der Eintrag in der Zeile 7. legt fest wie die Positionierung des Pop-Up Menüs der Zweiten Ebene berechnet werden soll. Mit „x“ wird dies von der Breite des Menüpunktes mit „y“ von der Höhe des Menüpunktes abhängig gemacht.

Mit dem Eintrag „topOffset“ in der Zeile 8. wurde der Abstand der Pop-Up Box vom zweiten Menü, gerechnet vom oberen Rand der Seite eingestellt. Ist dieser Eintrag nicht vorhanden wird das erste Menü von der Pop-Up Box überlappt.

In der Zeile 12. wird durch die Angabe von „XY = [10.w]+10,20“ die Höhe und die Breite der einzelnen Menüpunkte festgelegt die später als Graphik angezeigt werden. Durch die Angabe von „[10.w]+10“ wird die Berechnung der Breite dynamisch eingestellt, d.h. die Breite der späteren Graphik passt sich automatisch der Textbreite des Menüpunktes an.

Ansicht im Browser:



Wenn das Menü der 2 Ebene irgendwo anders im Browser angezeigt wird, wenn Sie das Menü über einen Marker platzieren, müssen Sie „relativeToTriggerItem = 1“ nach der 9. Zeilen in das obige Menü einfügen.

Erstellung 2. stufiges Pop-Up Menu mit Aktive und Rollover Effekt

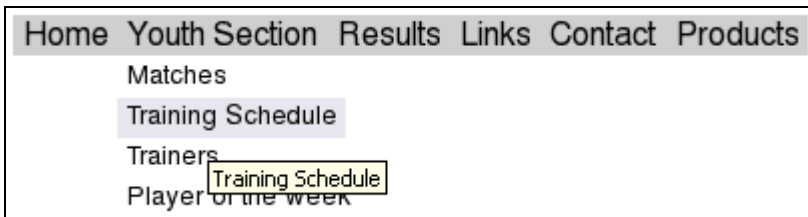
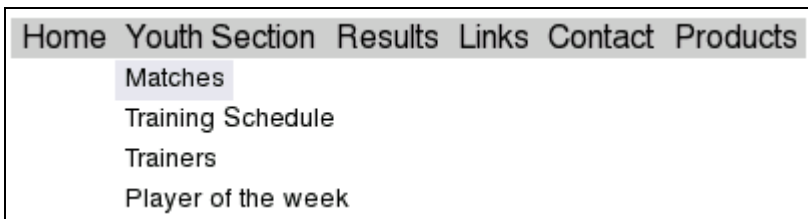
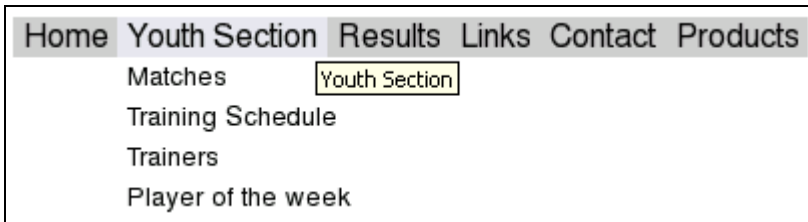
```
1. page = PAGE
2. page.includeLibs.gmenu_layers = media/scripts/gmenu_layers.php
3. page.10 = HMENU
4. page.10.1 = GMENU_LAYERS
5. page.10.1 {
6.   layerStyle = position:absolute;left:0px;top:30px;width:10px;VISIBILITY:hidden;
7.   xPosOffset = -10
8.   lockPosition = x
9.   topOffset = 36
10.  hideMenuWhenNotOver = 1
11.  expAll=1
12.  NO {
13.    backColor = #cccccc
14.    XY = [10.w]+10,20
15.    10 = TEXT
16.    10.text.field = title
17.    10.offset = 5,15
18.    10.fontFace = fileadmin/verdana.ttf
19.    10.fontSize = 12
20.  }
21.  RO < .NO
22.  RO.backColor = #e2e6e9
23.  RO = 1
24. }
25. page.10.2 = GMENU
26. page.10.2.noBlur = 1
27. page.10.2 {
28.  NO{
29.    backColor = white
30.    XY = [10.w]+10,20
31.    10 = TEXT
32.    10.text.field = title
33.    10.offset = 5,13
34.    10.fontFace = fileadmin/verdana.ttf
35.    10.fontSize = 10
36.  }
37.  RO < .NO
38.  RO.backColor = #e2e6e9
39.  RO = 1
```

40. }

Durch den Eintrag in Zeile 10. für „hideMenuWhenNotOver = 1“ wird das Menü der zweite Ebene sofort ausgeblendet wenn sich die Maus nicht auf den Menü der ersten oder zweiten Ebene befindet.

In der Zeile 29. wird durch den Wert „[10.w]+10“ die Breite für die dynamisch erzeugt Graphik festgelegt. Mit dem zweiten Wert „20“ wird die Höhe der Graphik festgelegt.

Ansicht im Browser bei „Mouseover“ Browser:



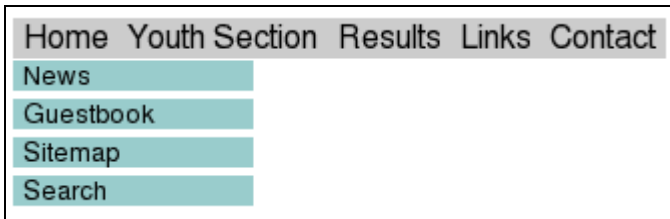
Wenn das Menü der 2 Ebene irgendwo anders im Browser angezeigt wird, wenn Sie das Menü über einen Marker platzieren, müssen Sie „relativeToTriggerItem = 1“ nach der 9. Zeilen in das obige Menü einfügen.

Einige Spielereien mit dem Pop-Up Menü

Es wurde versucht dem „wrap“ vom der zweiten Menüebene zu ändern. Eine Änderung des Eintrages „temp.menu.10.2.wrap = |
“ bewirkt nichts für den Aufbau des zweiten Menüs.

```
...
page.10.2.NO {
  wrap = | &nbsp;
  backColor = #99cccc
  ...
}
```

Ansicht im Browser:

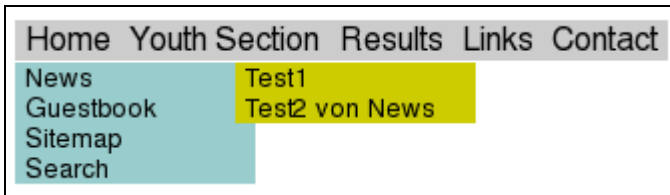


Erstellung eines 3 stufiges Pop-Up Menü (ähnlich MS Windows)

Mit folgend Code im Setup Bereich kann ein 3 stufiges Pop-Up Menü, ähnlich wie in Microsoft Windows Programmen erstellt werden. Außerdem können Sie durch die Klassendefinition „class“ jedem Menüpunkt eine eigene CSS Formatierung mitgeben:

```
page = PAGE
page.includeLibs.gmenu_layers = media/scripts/gmenu_layers.php
page.includeLibs.tmenu_layers = media/scripts/tmenu_layers.php
page.10 = HMENU
page.10.1 = GMENU_LAYERS
page.10.1 {
  layerStyle = position:absolute;left:0px;top:30px;width:10px;VISIBILITY:hidden;
  xPosOffset = -10
  lockPosition = x
  topOffset = 36
  expAll=1
  NO {
    backColor = #cccccc
    XY = [10.w]+10,20
    10 = TEXT
    10.text.field = title
    10.offset = 5,15
    10.fontFace = fileadmin/verdana.ttf
    10.fontSize = 12
  }
}
page.10.2 = GMENU_LAYERS
page.10.2 {
  layerStyle = position:absolute;left:0px;top:30px;width:10px;VISIBILITY:hidden;
  leftOffset = 110
  relativeToTriggerItem = 1
  relativeToParentLayer = 1
  expAll=1
  NO {
    backColor = #99cccc
    XY = 120,15
    10 = TEXT
    10.text.field = title
    10.offset = 5,12
    10.fontFace = fileadmin/verdana.ttf
    10.fontSize = 10
  }
}
page.10.3 = GMENU
page.10.3.NO {
  backColor = #cccc00
  XY = 120,15
  10 = TEXT
  10.text.field = title
  10.offset = 5,12
  10.fontFace = fileadmin/verdana.ttf
  10.fontSize = 10
}
```

Ansicht in Browser:



Durch Umstellung der „XY“ und „offset“ Werte können Sie die Position und die Größe der Popufenster beeinflussen.

Komplexes 3 stufiges Pop-Up Menü mit Farbcodierung

In diesem Pop-Up Menü werden können die einzelnen Menüpunkte in verschiedenen Farben dargestellt werden. Jedem Menüpunkt kann eine eigene Farbe vergeben werden.

Eingabe im Setup Bereich:

```

page = PAGE
page.includeLibs.gmenu_layers = media/scripts/gmenu_layers.php
page.includeLibs.tmenu_layers = media/scripts/tmenu_layers.php
page.10 = HMENU
page.10.1 = GMENU_LAYERS
page.10.1 {
  layerStyle = position:absolute;left:0px;top:30px;width:10px;VISIBILITY:hidden;
  xPosOffset = -10
  lockPosition = x
  topOffset = 36
  expAll=1
  NO {
    ATagParams = class = "main-menu-item{elementUid}"
    backColor = #cccccc
    XY = [10.w]+10,20
    10 = TEXT
    10.text.field = title
    10.offset = 5,15
    10.fontFace = fileadmin/verdana.ttf
    10.fontSize = 12
  }
}
page.10.2 = GMENU_LAYERS
page.10.2 {
  layerStyle = position:absolute;left:0px;top:30px;width:10px;VISIBILITY:hidden;
  leftOffset = 110
  relativeToTriggerItem = 1
  relativeToParentLayer = 1
  expAll=1
  NO {
    ATagParams = class = "sub-menu-item{elementUid}"
    backColor = #99cccc
    XY = 120,15
    10 = TEXT
    10.text.field = title
    10.offset = 5,12
    10.fontFace = fileadmin/verdana.ttf
    10.fontSize = 10
  }
}
page.10.3 = GMENU
page.10.3.NO {

```

```
ATagParams = class = "menu-item{elementUid}"  
backColor = #cccc00  
XY = 120,15  
10 = TEXT  
10.text.field = title  
10.offset = 5,12  
10.fontFace = fileadmin/verdana.ttf  
10.fontSize = 10  
}
```

Jetzt kann im CSS Style Sheet für jede Klasse „main-menu-item{elementUid}“, „sub-menu-item{elementUid}“ und für „menu-item{elementUid}“ das jeweilige Format und die Farbe festgelegt werden. Dabei wird „{elementUid}“ durch die „id“ Nummer des jeweiligen Menüelementes ersetzt, d.h. Sie können jedem Menüpunkt eine eigene CSS Definition zuweisen. Die angehängte Nummer entspricht dabei der „id“ im Seitenbaum. Die Position und Größe der Pop-up-Fenster können Sie wieder durch Umstellung der „XY“ und „offset“ Werte beeinflussen.

Erstellung von JavaScript Menüs

Drop-Down Menü mit dem JSMENU Objekt

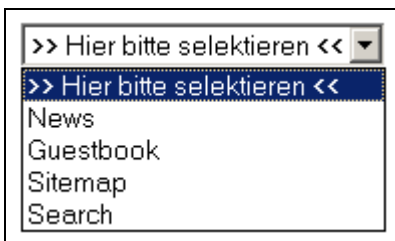
Mit dem JSMENU Objekt lassen sich sehr einfache JavaScript Menüs erstellen. Mit diesem Menütyp ist es möglich Menüs zu erstellen die mehrere Ebenen der Seitenstruktur erfassen, d.h. man kann damit mehrere Ebenen in einen Menü kombinieren.

Code im Setup Bereich für ein Drop-Down Menü:

```
temp.menu = HMENU
temp.menu.special = directory
temp.menu.special.value = 6
temp.menu.1 = JSMENU
temp.menu.1 {
  levels = 1
  1 {
    noLink = 0
    alwaysLink = 0
    showFirst = 0
    additionalParams = style="width: 200px;"
    firstLabel = >> Hier bitte selektieren <<
  }
}

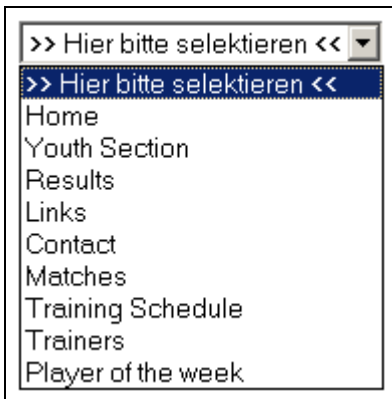
page = PAGE
page.30 < temp.menu
```

Ansicht im Browser:



Die Auswahl im Drop-Down Menü baut durch die Angabe von „...special.value = 6“ auf die erste Seitenstruktur unterhalb der Seite „Home“ auf.

Mit der Angabe „...special.value = 1,5“ wird die unten angezeigt Auswahl für das Drop-Down Menü zusammengestellt.



Einzelnen Menüpunkt vom Menü entfernen

Mit folgendem zusätzlichen Code kann ein einzelner Menüpunkt aus dem Menü entfernt werden:

```
temp.menu = HMENU
temp.menu.special = directory
temp.menu.special.value = 1,5
temp.menu.excludeUidList = 10
...
```

Durch die Angabe von „...excludeUidList = 10“ wird die Seite mit der ID nicht mehr im Menü angezeigt.

Ansicht im Browser:



Es fehlt jetzt der Menüpunkt „Matches“ in der Auswahlliste

Mehrstufiges DropDown Menü

```
temp.menu = HMENU
temp.menu.special = directory
temp.menu.special.value = 6
temp.menu.1 = JSMENU
temp.menu.1 {
  levels = 3
  1 {
    noLink = 0
    alwaysLink = 0
    showFirst = 0
    additionalParams = style="width: 200px;"
    firstLabel = >> Hier bitte selektieren <<
  }
  2 {
    noLink = 0
    alwaysLink = 0
    showFirst = 0
    additionalParams = style="width: 200px;"
    firstLabel = >> Hier bitte selektieren <<
  }
  3 {
    noLink = 0
    alwaysLink = 0
    showFirst = 0
    additionalParams = style="width: 200px;"
    firstLabel = >> Hier bitte selektieren <<
  }
}
```

Spezielle Menü Konstruktionen

Horizontales Menü wobei jeder Menüpunkt eine andere Farbe hat

```

MENU_TOP = HMENU
MENU_TOP {
  entryLevel = 1
  1 = TMENU
  1 {
    wrap =
    noBlur = 1
    NO = 1
    NO.allWrap = <span class=blue_menu_two> | </span> || <span class=pink_menu_two> | </span> ||
<span class=yellow_menu_two> | </span> || <span class=green_menu_two> | </span> || <span
class=blue_menu_two> | </span> || <span class=pink_menu_two> | </span> || <span
class=yellow_menu_two> | </span> || <span class=green_menu_two> | </span> || <span
class=blue_menu_two> | </span> || <span class=pink_menu_two> | </span>
    ACT = 1
    ACT.allWrap = <span class=blue_menu_one> | </span> || <span class=pink_menu_one> | </span> ||
<span class=yellow_menu_one> | </span> || <span class=green_menu_one> | </span> || <span
class=blue_menu_one> | </span> || <span class=pink_menu_one> | </span> || <span
class=yellow_menu_one> | </span> || <span class=green_menu_one> | </span> || <span
class=blue_menu_one> | </span> || <span class=pink_menu_one> | </span>
    ACT.doNotLinkIt = 1
  }
}

```

Menü für Sprachumschaltung mit Flaggen

```

##### Menü für Sprachumschaltung #####
temp.lang_menu = COA
temp.lang_menu {
  10 = HMENU
  10 {
    # deutsch und englisch
    special = language
    special.value = 0,1

    1 = GMENU
    1 {
      noBlur = 1
      NO = 1
      NO {
        XY = [10.w]+4,[10.h]+4
        backColor = white
        10 = IMAGE
        10.file = typo3/sysex/cms/tslib/media/flags/flag_de_f.gif ||
typo3/sysex/cms/tslib/media/flags/flag_uk_f.gif
        10.offset = 2,2
      }
      ACT < .NO
      ACT {
        10.file = typo3/sysex/cms/tslib/media/flags/flag_de_d.gif ||
typo3/sysex/cms/tslib/media/flags/flag_uk_d.gif
        noLink = 1
        #backColor = #ff0000
      }
    }
  }
}

```

```

USERDEF1 < .NO
USERDEF1 {
  10.file = typo3/sysex/cms/tslib/media/flags/flag_de_d.gif ||
typo3/sysex/cms/tslib/media/flags/flag_uk_d.gif
}
USERDEF2 < .ACT
}
}
}
}

```

Zusätzliche Menü mit Farbcodierung aus „tt_news_cat“

Es wurde in der Tabelle „tt_news_cat“ ein neues Feld mit dem Namen „tx_ttnewscatcolor_color“ angelegt als „TINYTEXT“ und mit den Farbwerten gefüllt die der Background des jeweiligen Menüs erhalten soll.

```

# Menu Products für ein horizontale Sub Menü
temp_menue_products = COA
temp_menue_products {
  20 = HMENU
  20 {
    # MENU with TWO LEVEL not Possible with social mode :- ( but when you make additional entry
level then it works)
    special = directory
    special.value = 9
    # Entry Level für das Product Menü (Hersteller)
    entryLevel = 32
    1 = TMENU
    1 {
      NO {
        linkWrap = 1|1
        allWrap = A|
        ATagBeforeWrap = 1
        beforeImg = GIFBUILDER
        beforeImg {
          XY = 60,20
          backColor.cObject = CONTENT
          backColor.cObject.table = tt_news_cat
          backColor.cObject.select.pidInList = 7
          backColor.cObject.select.andWhere = title LIKE '{field:title}'
          backColor.cObject.select.andWhere.insertData = 1

          backColor.cObject.renderObj = COA
          backColor.cObject.renderObj.10 = TEXT
          backColor.cObject.renderObj.10.field = tx_ttnewscatcolor_color
        }
      }
    ACT < .NO
    ACT = 1
  }
}
}

```


Komplexe Menüs mit Graphiken

Erstelle Menü mit GIFBUILDER

```
#####main menu#####

temp.menu_header = IMAGE
temp.menu_header.wrap = <div style="width:163px;margin-left:18px;margin-top:20px;"><div
style="background-color:#435293;width:163px;height:1px;margin-top:9px;"></div></div>
temp.menu_header.altText.text = Menü

temp.menu_header.file = GIFBUILDER
temp.menu_header.file {

  XY = [10.w]+3,[10.h]+7

  backColor = #FFFFFF

  10 = TEXT
  10.text.wrap = |
  10.text = Menü
  10.text.case = normal
  10.fontSize = 18
  10.fontColor = #435293
  10.fontFile = fileadmin/fonts/tt0250m.ttf
  10.niceText = 1
  10.offset = 0,23
  10.align = left

}

temp.main_menu = HMENU
temp.main_menu.stdWrap.wrap = <table border="0" cellpadding="0" cellspacing="0" width="163"
style="margin-top:20px;margin-left:18px;"></table>
temp.main_menu.entryLevel = 0

temp.main_menu.1 = TMENU
temp.main_menu.1.target = _top

temp.main_menu.1.NO {
  subst_elementUid = 1
  allWrap = <tr></tr><tr><td style="height:1px;width:20px;"></td><td style="height:1px;width:143px;"></td></tr>
  ATagBeforeWrap = 0
  linkWrap = <td style="width:143px;padding-top:2px;padding-bottom:2px;background-color:#E3E3E3;"
align="left" class="menuno" id="bg_col{elementUid}"></td>
  beforeImg = fileadmin/pics/icon_main.gif
  beforeROImg = fileadmin/pics/icon_main_a.gif
  beforeWrap = <td style="width:20px;background-color:#E3E3E3;" align="left" valign="top"
class="menuno"></td>

  RO_chBgColor = #3C5D90 | #E3E3E3 | bg_col
  RO = 1

}
```

```

temp.main_menu.1.ACT < temp.main_menu.1.NO
temp.main_menu.1.ACT = 1
temp.main_menu.1.ACT {

    allWrap = <tr></tr><tr><td style="height:1px;"></td><td style="height:1px;width:143px;"></td></tr>
    ATagBeforeWrap = 0
    linkWrap = <td style="width:143px;padding-top:2px;padding-bottom:2px;background-color:#3C5D90;"
align="left" class="menuact"></td>
    beforeImg = fileadmin/pics/icon_main_a.gif
    beforeWrap = <td style="width:20px;background-color:#3C5D90;" align="left" valign="top"
class="menuuno"></td>
}

temp.main_menu.2 = TMENU
temp.main_menu.2.target = _top
temp.main_menu.2.NO {

    subst_elementUid = 1
    allWrap = <tr><td colspan="2" width="163"><table border="0" cellpadding="0" cellspacing="0"
width="163"><tr><td style="height:1px;width:27px;"></td><td style="height:1px;width:136px;"></td></tr></table></td></tr>
    ATagBeforeWrap = 0
    linkWrap = <td style="width:137px;padding-top:2px;padding-bottom:2px;padding-left:1px;background-
color:#E3E3E3;" class="menuuno" id="bg_col{elementUid}"></td></tr>
    beforeImg = fileadmin/pics/icon_sub.gif
    beforeROImg = fileadmin/pics/icon_sub_a.gif
    beforeWrap = <tr><td style="width:27px;background-color:#6989B8;"></td>

    RO_chBgColor = #6989B8 | #E3E3E3 | bg_col
    RO = 1
}

temp.main_menu.2.ACT < temp.main_menu.2.NO
temp.main_menu.2.ACT = 1
temp.main_menu.2.ACT {

    allWrap = <tr><td colspan="2" width="163"><table border="0" cellpadding="0" cellspacing="0"
width="163"><tr><td style="height:1px;width:27px;"></td><td style="height:1px;width:136px;"></td></tr></table></td></tr>
    ATagBeforeWrap = 0
    linkWrap = <td style="width:137px;padding-top:2px;padding-bottom:2px;padding-left:1px;background-
color:#6989B8;" align="left" class="menuact"></td></tr>
    beforeImg = fileadmin/pics/icon_sub_a.gif
    beforeWrap = <tr><td style="width:27px;background-color:#6989B8;" valign="top"></td>
}

temp.main_menu.3 = TMENU
temp.main_menu.3.target = _top
temp.main_menu.3.NO {

    subst_elementUid = 1

```

```
allWrap = <tr><td colspan="2" width="163"><table border="0" cellpadding="0" cellspacing="0"
width="163"><tr><td style="height:1px;width:40px;"></td><td style="height:1px;width:123px;"></td></tr></table></td></tr>
  ATagBeforeWrap = 0
  beforeImg = fileadmin/pics/icon_subsub.gif
  beforeROImg = fileadmin/pics/icon_subsub_a.gif
  linkWrap = <td style="width:124px;padding-top:2px;padding-bottom:2px;padding-left:1px;background-
color:#E3E3E3;" class="menuuno" id="bg_col{elementUid}"></td></tr>
  beforeWrap = <tr><td style="width:40px;background-color:#899AB6;"></td>

  RO_chBgColor = #899AB6 | #E3E3E3 | bg_col
  RO = 1
}

temp.main_menu.3.ACT < temp.main_menu.3.NO
temp.main_menu.3.ACT = 1
temp.main_menu.3.ACT {

  allWrap = <tr><td colspan="2" width="163"><table border="0" cellpadding="0" cellspacing="0"
width="163"><tr><td style="height:1px;width:40px;"></td><td style="height:1px;width:123px;"></td></tr></table></td></tr>
  ATagBeforeWrap = 0
  beforeImg = fileadmin/pics/icon_subsub_a.gif
  linkWrap = <td style="width:124px;padding-top:2px;padding-bottom:2px;padding-left:1px;background-
color:#899AB6;" align="left" class="menuact"></td></tr>
  beforeWrap = <tr><td style="width:40px;background-color:#899AB6;" valign="top"></td>
}
```