

Tvorba aplikácie v PHP

Tvorba webovej databázovej aplikácie by sa mala začať načrtnutím hlavných rysov jej koncepcie, určením hlavných funkcií a ich (aspoň predbežnej) dekompozície na zodpovedajúce skripty. Predpokladáme, že (aspoň principiálne) je vytvorený koncept databázovej infraštruktúry a máme vytvorené príslušné databázové tabuľky.

Riešenie rozloženia stránky

Je samozrejmosťou riešiť vzhľad a rozloženie stránky s využitím kaskádových štýlov. Preto si buď navrhujeme a vytvoríme vlastný súbor kaskádového štýlu, alebo využijeme niektoré z veľkého množstva hotových riešení, dostupných na rôznych vývojárskych portáloch (napr. www.interval.cz). Medzi najjednoduchšie a veľmi obľúbené patrí dvojstĺpcový layout stránky, ktorý má v ľavom paneli navigáciu stránky, obsahuje hlavičku aj päť stránky. Na uvedenej stránke (<http://interval.cz/clanky/dvousloupcovy-layout-s-pevnou-sirkou/>) nájdeme kaskádový štýl v tvare

```
body
{
margin:0;
padding:0;
background:#ffffff;
text-align:center;
}
#hlavicka
{
padding: 10px;
border: 1px solid blue;
background: #dddddd;
}
#obsah
{
width: 750px;
margin:1em auto;
border: 1px solid blue;
text-align:left;
}
#levypanel
{
float: left;
width: 200px;
margin:0;
padding:0;
}
#levypanel-obsah
```

```

{
margin:0;
padding: 0.5em 1em;
}
#hlavnipanel
{
margin:0 0 0 200px;
padding:0;
}
#hlavnipanel-obsah
{
margin:0;
padding: 0.5em 1em;
}
#paticka
{
padding: 10px;
border-top: 1px solid blue;
background: #dddddd;
}
hr.cleaner
{
clear:both;
display: none
}

```

Aplikácia uvedeného štýlu na html má tvar

```

<html>
<body>
<div id="obsah">
<div id="hlavicka">
Zde se nachází hlavička...
<a href="http://css.interval.cz/clanky/dvousloupcovy-layout-s-pevnou-sirkou/">Zpět na
článek: Dvousloupcový layout s pevnou šířkou</a>
</div>
<hr class="cleaner" />
<div id="levypanel">
<div id="levypanel-obsah">
Menu s pevnou šířkou...
    <ul>
        <li><a href="index.html">Odkaz 1</a></li>
        <li><a href="index.html">Odkaz 2</a></li>
        <li><a href="index.html">Odkaz 3</a></li>
        <li><a href="index.html">Odkaz 4</a></li>
        <li><a href="index.html">Odkaz 5</a></li>
    </ul>
</div>
</div>
<div id="hlavnipanel">
<div id="hlavnipanel-obsah">
<h1>Obsah s pevnou šířkou</h1>
<p>Lorem ipsum dolor ...</p>

```

```
</div>
</div>
<hr class="cleaner" />
<div id="paticka">
    Patička
</div>
</div>
</body>
</html>
```

Jeho výsledkom je výstup v tvare

[Zpět na článek: Dvousloupcový layout s pevnou šířkou](#)

Menu s pevnou šířkou

- [Odkaz 1](#)
- [Odkaz 2](#)
- [Odkaz 3](#)
- [Odkaz 4](#)
- [Odkaz 5](#)

Obsah s pevnou šířkou

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut nec dui. Integer tortor wisi, adipiscing id, lacinia in, dapibus ut, leo. Suspendisse rhoncus sapien et est aliquam vulputate. Nam non nibh. Praesent posuere tortor at risus. Nam facilisis elementum elit. Quisque condimentum congue quam. Fusce fringilla malesuada lectus. Nam tristique, mauris eget faucibus ullamcorper, neque tellus elementum metus, sit amet iaculis pede leo eget diam. Proin malesuada, magna vel vehicula pretium, tortor ipsum ultricies augue, sit amet ultrices wisi ipsum non nulla. Fusce blandit, velit et lacinia auctor, purus magna condimentum ligula, a facilisis mi mi nec est.

Aenean eu augue sit amet velit suscipit vulputate. Ut augue quam, tempus tristique, vulputate in, vehicula ac, odio. Quisque vulputate diam a dolor. Maecenas molestie justo eu sapien. Donec pede metus, pulvinar ac, nonummy vitae, ultricies a, urna. Nunc porta. Vestibulum augue. Aenean mattis sapien non sem. Nullam justo est, aliquam in, facilisis eget, mattis id, mi. Mauris ac lorem facilisis urna convallis consectetur. Vivamus id ipsum quis odio aliquet adipiscing. Vestibulum mattis est accumsan quam. Suspendisse at ligula sed ante faucibus bibendum. Vivamus scelerisque, pede ac mattis sagittis, justo nisl feugiat libero, ut adipiscing ligula lorem a tellus. Fusce magna pede, tempor nec, placerat vel, interdum et, leo. Integer odio. Sed luctus. Quisque nec ligula eu mauris ornare placerat. Mauris mi sem, lacinia vel, vehicula ut, porta ac, metus. Proin sollicitudin laoreet augue.

Nunc a lacus. Phasellus cursus, mi ac commodo ultricies, velit magna pellentesque lorem, et auctor diam nibh ac est. Phasellus mattis egestas risus. Sed dolor felis, molestie ut, lobortis dictum, posuere ac, orci. Ut a libero in leo sagittis dignissim. In hac habitasse platea dictumst. Sed tempor lectus ac mauris. Aenean turpis erat, ultricies sed, sollicitudin eu, porttitor dictum, mauris. Praesent quam lacus, malesuada tempor, blandit vitae, condimentum sed, ante. Vestibulum id nulla quis nulla volutpat viverra. Morbi tincidunt. Cras neque lectus, tincidunt sed, imperdiet a, mollis vitae, orci. Curabitur semper, leo eu egestas euismod, ante arcu malesuada diam, id commodo est lorem sollicitudin nibh. Praesent magna dolor, pellentesque vel, facilisis vel, aliquet eu, neque. Ut quis dui vitae justo sagittis egestas. Pellentesque metus leo, placerat non, hendrerit a, tristique quis, turpis.

Nad týmto štýlom vykonáme ešte niekoľko drobných úprav, doplnkov a zmien podľa vlastného vkusu a potreby, resp. požiadaviek aplikácie – do pozadia obsahu umiestnime obrázok a jeho šírku nastavíme na 100%, maximálnu šírku na 1024px. Výsledný tvar css bude

```
body
{
margin:0; padding:0;
background:#ffffff;
text-align:center;
}

#hlavicka
{
padding: 1px;
border-bottom: 1px solid blue;
text-align:center;
background-color: rgb(221,221,221);
}

#obsah
{
width: 100%;
margin:1em auto;
border: 1px solid blue;
text-align:left;
max-width: 1024px;
background: url('greenyellow2.gif') top left repeat;
}

#levypanel
{
float: left;
width: 100px;
margin:0; padding:0;
}

#levypanel-obsah
{
margin:0;
padding: 0.5em 1em;
}

#hlavnipanel
{
margin:0 0 0 120px;
padding:0;
}

#hlavnipanel-obsah
{
margin:0;
padding: 0.5em 1em;
```

```

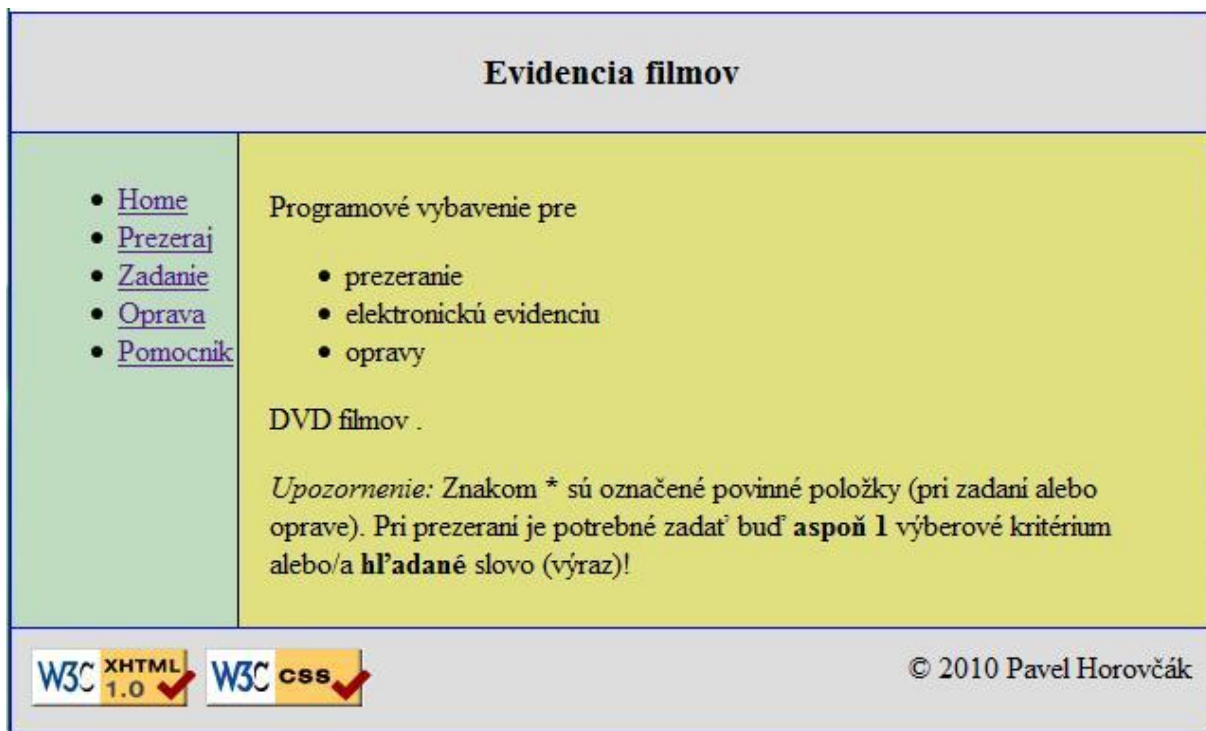
}

#paticka
{
padding: 10px;
border-top: 1px solid blue;
background: #dddddd;
}

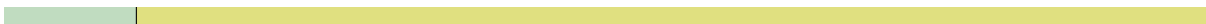
hr.cleaner
{
clear:both;
display: none
}

```

Výsledok má tvar



Tvar súboru *greenyellow2.gif* je nasledovný (môžeme ho vytvoriť napr. aj v Skicári):



Okrem toho ešte bude použitý súbor kaskádového štýlu *styles.css* v tvare

```

.Table1 {
border: rgb(0,100,0) 1px solid;
background-color: rgb(144,238,144);
text-align:center;
}

```

```

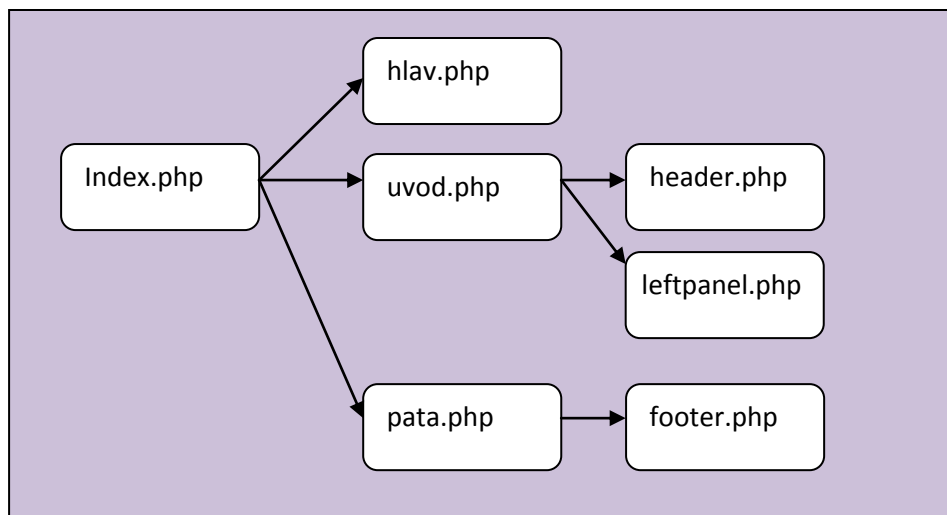
.TableHead1 {
  background-color: rgb(11,238,144);
  font-family: Verdana, Arial;
  font-weight: bold;
  font-size: 10pt;
}
.TableEven{
  background-color: #ddb; /* rgb(144,192,0); rgb(210,255,156); #ddb; */
}

.TableContent1 {
  font-family: Verdana, Arial;
  font-size: 10pt;
}
}
.Table2 {
  border: Blue 1px solid;
  background-color: Blue;
}
}
.TableHead2 {
  font-family: Verdana, Arial;
  font-weight: bold;
  font-size: 10pt;
}
}
.TableContent2 {
  font-family: Verdana, Arial;
  font-size: 10pt;
}
}

```

Riešenie kompozície stránky

Riešenie úzko súvisí so zvoleným rozložením stránky. To znamená, že pre jednotlivé útvary stránky budú vytvorené zodpovedajúce skripty, resp. úseky



kódu a výsledná stránka budekomponovaná ich skladaním. Schéma úvodnej stránky aplikácie má tvar:

Súbor *hlav.php* označuje začiatok XHTML súboru:

```

<?php //hlav.php
print("<?xml version='1.0' encoding='utf-8' ?>\n
<!DOCTYPE html PUBLIC '-//W3C//DTD XHTML 1.0 Strict//EN'
'http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd'>
<html xmlns='http://www.w3.org/1999/xhtml' xml:lang='sk' lang='sk'>
<head>
");
?>

```

Súbor *uvod.php* ukončuje úvodnú časť a začína obsahovú časť aplikácie

```

<?php //uvod.php
printf("<meta http-equiv='content-language' content='sk' />
<meta http-equiv='content-type' content='text/html; charset=utf-8' />
<meta name='Author' content='Pavel Horovcak' />
<script type='text/javascript' src='gw.js'></script>
<link rel='shortcut icon' href='./styly/filmy.ico' />
<link rel='stylesheet' href='./styly/stzel.css' type='text/css' />
<link rel='stylesheet' type='text/css' href='styly/styl.css' />
<link rel='stylesheet' type='text/css' href='styly/styles.css' /> </head>
<body onload='gw()'>
");
printf("<div id='obsah'>");
require("header.php");
require("leftpanel.php");
//error_reporting(E_ALL ^ E_NOTICE); //potlacsenie notice
printf("<div id='hlavnipanel'>
<div id='hlavnipanel-obsah'>
");
?>

```

Je doplnený hlavičkou stránky *header.php*

```

<?php //header.php
printf("<div id='hlavicka'>
<span style='float:left;'>
<!--a href='http://www.sacr.sk'><abbr title='www.sacr.sk'><img
src='./styly/logosacr7.gif' alt='www.sacr.sk' /></abbr></a-->
</span>
<span style='float:right;'>
<!--abbr title='miesto pre obrazok'><img src='./styly/*.jpg' alt='' /></abbr-->
</span>
<h3>Evidencia filmov </h3>
</div>
<hr class='cleaner' />
");
?>

```

a navigačným panelom, označeným ako *leftpanel.php*

```

<?php //leftpanel.php
printf("
<div id=\"levypanel\">
<div id=\"levypanel-obsah\">
    <ul>
        <li><a href=\"index.php\">Home</a></li>
        <li><a href=\"see_film.php\">Prezeraj</a></li>
        <li><a href=\"new_film.php\">Zadanie</a></li>
        <li><a href=\"update_film.php\">Oprava</a></li>
        <li><a href=\"sum_film.php\">Sumár</a></li>
        <li><a href=\"pomoc.php\">Pomocník</a></li>
    </ul>

</div>
</div>
");
?>

```

Riešenie päty stránky *pata.php* má tvar

```

<?php //pata.php
printf("</div>
</div>");
require("footer.php");
printf("</div>
</body>
</html> ");
?>

```

a jeho obsah je uložený v súbore *footer.php*

```

<?php //footer.php
print("
<hr class=\"cleaner\" />
<div id=\"paticka\">
<span style=\"float:right;text-align: right;\">
&copy; 2010
    Pavel Horov&#269;&aacute;k
</span>
    <a href=\"http://validator.w3.org/check?uri=referer\"><img
        style=\"border:0;width:88px;height:31px;\"
        src=\"./styly/valid-xhtml10.gif\"
        alt=\"XHTML 1.0 Strict\" height=\"32\" width=\"88\" /></a>

    <a href=\"http://jigsaw.w3.org/css-validator/check?uri=referer\">
    <img style=\"border:0;width:88px;height:31px;\" src=\"./styly/vcss.png\"
        alt=\"Valid CSS!\" /> </a>

</div>
");
?>

```

Kompozícia týchto zložiek vytvára úvodný skript stránky *index.php* v tvare

```
<?php require("hlav.php"); ?>
<title>Horovčák's films</title>

<?php require("uvod.php"); ?>

<br />Programové vybavenie pre <ul>
<li>prezeranie</li>
<li>elektronickú evidenciu</li>
<li>opravy</li>
</ul>DVD filmov
<br /><p><em>Upozornenie:</em> Znakom <b>*</b> sú označené povinné položky
(pri zadaní alebo oprave). Pri prezeraní je potrebné zadať
    buď <b>aspoň 1</b> výberové kritérium alebo/a <b>hľadané</b> slovo (výraz)!</p>

<?php require("pata.php"); ?>
```

Rovnakým spôsobom sú vytvárané aj všetky ostatné stránky, ktoré vykonávajú jednotlivé špecifické funkcionality aplikácie.

Prístup k databáze

Je riešený s využitím dvoch súborov. Prvý súbor *db_const.php* obsahuje autentifikačné údaje na prístup do databázy, v druhom súbore *db.php* je realizované pripojenie na databázu a kontrola prítomnosti rozšírenia mysqli, s ktorým jednotlivé skripty a funkcie pracujú. Súbory majú tvar:

```
<?php //db_const.php
$host="localhost";
$user="filmy";
$db="filmy";
$tb="film";
$psw="";
$siz=6; //max. veľkosť selectov - počet riadkov na obrazovke
?>
```

```
<?php //db.php
    require "../db_const.php";
    $em='mysqli';

    $mysqli = new mysqli($host,$user,$psw,$db);
    /* check connection */
    if (mysqli_connect_errno()) {
        printf("Chyba pripojenia: %s.\n<br>", mysqli_connect_error());
        printf("Connect failed: %s.\n", mysqli_connect_error());
        exit();
    }
    /*check extension */
```

```

if(!extension_loaded($em)){
    echo "PHP ".phpversion()."<br>";
    printf("MySQL server: %s\n<br>", $mysqli->server_info);
    printf("MySQL client: %d\n<br>", $mysqli->client_info);
    printf("Extension $em NOT loaded. Exit!<br>");
    exit();
}
?>

```

Pomocné funkcie aplikácie

Pretože sa jedná o pomerne komplexnejšiu aplikáciu, kde sa viaceré činnosti a algoritmy opakovane využívajú s rôznymi hodnotami parametrov, je vhodné pripraviť pre tieto činnosti úseky kódu vo forme podprogramov – funkcií. V praxi sa osvedčilo umiestnenie týchto podprogramov do jedného spoločného súboru, ktorý je zaradzovaný do všetkých skriptov, využívajúcich niektorú z jeho funkcií. V našej aplikácii je tento spoločný súbor označený ako *common.php*. Jednou z najfrekvencovanejších funkcií je funkcia *zoznamt()*, ktorá zabezpečí načítanie obsahu dvoch zadaných stĺpcov tabuľky (ide o všetky číselníky aplikácie a aj o hlavnú tabuľku), ich naplnenie do HTML príkazu `<select>` a jeho položiek `<option>` a jeho vygenerovanie na stránku. Využíva pritom skutočnosť, že primárny kľúč tabuľky je odvodený od názvu stĺpca predponou `id_`, takže napr. pre číselník jazykov platí 1. stĺpec: `id_lang`, 2. stĺpec: `lang` (pozri upravený dátový model aplikácie v časti [Tvorba tabuliek v PHP](#)). Vstupnými parametrami funkcie sú názov tabuľky (`$tb0`), identifikátor konekcie na databázu (`$mysqli`), názov stĺpca tabuľky, ktorý bude umiestnený do príkazu `<select>` (`$name`), identifikátor režimu činnosti funkcie (`$tb1`) a pole položiek, ktoré budú v príkaze `<select>` zvýraznené - `selected` (`$slist`). Funkcia a začiatok súboru *common.php* má tvar:

```

// common.php functions
$ar[]=null; //prazdne pole pre potreby volania funkcie zoznamt
$h1="h3";
$h2="h3";
// require "tb_names.php"; //nazvy tabuliek
require ("get_param.php"); //prenos parametrov

function zoznamt($tb0,$mysqli,$name,$tb1,array $slist){
    //naplnenie zoznamu obsahom tabulky $tb0 atribut $name
    //$tb1=0 -> len $tb0 bez where, 1 - multiple
    //$tb1=2 - pre subnames a subname, $slist[0] obsahuje id_film
    //$tb1=4 - pre subnames a sublength
    //$tb1=3 - oznac selected v langs,titles,actors,serials podľa pola $slist
    require "db.php";
    //urcit parameter multiple
    $multiple = ($tb1>0) ? "multiple=\"multiple\" : \"\"";
    //urcit ci nazov selectu je pole alebo jednoduchá premenná
    $namea = ($tb1>0) ? $name."[]" : $name;
    //nazov prvych dvoch stlpcov tabulky

```

```

$names="id_".$name." ".$name;
$cp = 'utf8'; $mysqli->set_charset($cp);
$sql = "select distinct $names from $tb0 order by $name";
if($tb1==2)$sql="select $names from $tb0 where id_film=$slist[0]";
if($tb1==4)$sql="select id_subname,$name from $tb0 where id_film=$slist[0]";
if($result = $mysqli->query($sql)){
    //pocet riadkov
    $rows = $result->num_rows;
    //korekcia na $siz
    $size = $rows<$siz ? $rows : $siz;
    echo "<select name=\"\$namea\" size=\"\$size\" class=\"select\" $multiple>";
    for ($i = 0; $i < $rows; $i++) {
        //vyber riadkov z matice
        $row = $result->fetch_array(MYSQLI_NUM);
        //urcit ci je id v zozname (poli) $slist a nastavit parameter selected
        $sel = in_array($row[0],$slist) ? " selected=\"selected\"" : "";
        if(($tb1==2)||($tb1==4))$sel= " selected=\"selected\"";
        //pre tb1=3 vypis so $sel, inak bez
        if($tb1>1)print "<option value=\"".$row[0]."$sel\">$row[1]</option>";
        else print "<option value=\"".$row[0]."$sel\">$row[1]</option>";
    }
}
//ukoncit select
print "</select>";
$mysqli->close();
}

```

Režim činnosti funkcie (\$tb1) definuje tieto možnosti:

- tb1=0 vytvára príkaz <select> s možnosťou iba jednej voľby položky, čo platí napríklad pre číselník žánrov alebo seriálov
- tb1=1 vytvára príkaz <select multiple > doplnený parametrom multiple s možnosťou viacnásobnej voľby položiek (stlačením klávesa Ctrl), čo platí pre číselníky jazykov, titulkov a hercov a súčasne vytvára názov príkazu <select> v tvare pol'a s hodnotou parametra \$name (pre 'genre' to bude name="genre", pre 'lang' bude name="lang[]")
- tb=3 vytvára príkaz <select multiple > s označením (selected) zvolených položiek <option>v tabuľkách langs,titles,actors,serials podľa pol'a \$slist
- tb=2 vytvára príkaz <select multiple > s označením (selected) všetkých položiek <option>pre tabuľku subnames a stĺpec subname, pričom v poli \$slist na nulte pozícii je umiestnená hodnota id_film
- tb=4 vytvára príkaz <select multiple > s označením (selected) všetkých položiek <option>pre tabuľku subnames a stĺpec sublength, pričom v poli \$slist na nulte pozícii je umiestnená hodnota id_film

Okrem toho je vytváraný príkaz <select> doplnený aj o potrebné odkazy na položky kaskádového štýlu. Vytvorený príkaz <select> je následne inkorporovaný do príslušnej stránky.

Tvar súboru *get_param.php*

```
<?php //get_param.php

foreach($_REQUEST as $ind => $val){
    //$val=addslashes($val); //prida spatne lomitka k ',\ a NULL
    if(!is_array($val)){ //okrem poli
        eval("$".$ind."=\"".$val."\""); //vyhodnoti retazec ako PHP kod
        //echo "$ind=$val<br>";
    }
}
?>
```