

UNIVERZITA PARDUBICE

Fakulta elektrotechniky a informatiky

Aplikace pro tvorbu Javascriptové webové fotogalerie

Lukáš Kupr

Bakalářská práce

2011

Univerzita Pardubice  
Fakulta elektrotechniky a informatiky  
Akademický rok: 2010/2011

## ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Lukáš KUPR**  
Osobní číslo: **I08100**  
Studijní program: **B2646 Informační technologie**  
Studijní obor: **Informační technologie**  
Název tématu: **Aplikace pro tvorbu Javascriptové webové fotogalerie**  
Zadávající katedra: **Katedra informačních technologií**

### Z á s a d y p r o v y p r a c o v á n í :

Cílem bakalářské práce je vytvoření aplikace pro generování webových fotografických galerií založených na využití HTML a Javascriptu. Výsledná galerie musí být snadno přenositelná na různé typy webů. V současnosti existuje velké množství Javascriptových galerií - např. Highslide.

#### Teoretická část

Cílem teoretické části je rešerše současných softwarových generátorů a popis technologií vhodných k jejich tvorbě. Dále bude teoretické část obsahovat popis možných omezení jazyka Javascript a popis problémů s různými typy internetových prohlížečů.

#### Praktická část

Cílem praktické části je vytvoření aplikace v jazyce Javascript pro tvorbu webové galerie z vlastních obrázků a její srovnání s dostupnými aplikacemi.

Rozsah grafických prací:

Rozsah pracovní zprávy:

Forma zpracování bakalářské práce: tištěná/elektronická

Seznam odborné literatury:

\*Slavoj Písek: HTML a XHTML začínáme programovat, Grada 2003,  
ISBN 80-247-0571-0

\*Dave Thau: Velký průvodce JavaScriptem - tvorba interaktivních webo-  
vých stránek v praxi, Grada 2009, ISBN 978-80-247-2211-5

Vedoucí bakalářské práce:

RNDr. Josef Rak  
Katedra informačních technologií

Datum zadání bakalářské práce: 17. prosince 2010


Termín odevzdání bakalářské práce: 13. května 2011



prof. Ing. Simeon Karamazov, Dr.  
děkan



L.S.



Ing. Lukáš Čegan, Ph.D.  
vedoucí katedry

V Pardubicích dne 31. března 2011

## **Prohlášení autora**

Prohlašuji, že jsem tuto práci vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Souhlasím s prezenčním zpřístupněním své práce v Univerzitní knihovně.

V Pardubicích dne 10.8.2011

Lukáš Kupr

## **Poděkování**

Tímto bych chtěl poděkovat panu RNDr. Josefu Rakovi za rady a připomínky, kterými mi pomohl při tvorbě této bakalářské práce. Dále bych rád poděkoval své rodině za trpělivost, kterou se mnou měli po dobu pracování na bakalářské práci.

## **Anotace**

Bakalářská práce se zabývá problematikou softwarových generátorů webových galerií založených na Javascriptu. Práce se snaží čtenáři přiblížit základní technologie používané v oblasti internetových stránek. Také popisuje možné problémy s implementací Javascriptu v jednotlivých prohlížečích. Dále jsou představeny dostupné softwarové generátory a vhodné technologie k jejich vývoji. V praktické části je vytvořen program pro generování javascriptových galerií. Tento program byl naprogramován v jazyku Java. Dále je součástí této práce skript napsaný v jazyku Javascript, který se stará o zobrazení obrázků galerie na webových stránkách.

## **Klíčová slova**

Javascript, Java, C++, HTML, XHTML, CSS, webová galerie, internetové prohlížeče, generátor webových galerií

## **Title**

Application for creating Javascript web photo gallery

## **Annotation**

Bachelor thesis deals with software generators web galleries based on JavaScript. The thesis tries to bring readers the basic technologies used in web pages. The thesis also describes possible problems with implementation of JavaScript in different browsers. There are also introduced software generators available on the Internet and appropriate technologies for their development. In the practical part of the thesis there is a developed program for generating JavaScript web galleries. The program was programmed in Java language. In the practical part of the thesis is a script written in JavaScript also, that provides displaying images on web pages.

## **Keywords**

Javascript, Java, C++, HTML, XHTML, CSS, web gallery, internet browsers, web gallery generator

# 1 Obsah

<b>1</b>	<b>Obsah</b> .....	<b>6</b>
	Seznam zkratk.....	8
	Seznam obrázků.....	9
	Seznam tabulek .....	9
<b>2</b>	<b>Technologie zobrazování obrázků na webu</b> .....	<b>10</b>
<b>3</b>	<b>Základní pojmy</b> .....	<b>12</b>
3.1	HTML .....	12
3.1.1	HTML tágy .....	12
3.1.2	Vnořené elementy .....	12
3.1.3	Validita HTML dokumentu .....	12
3.1.4	XHTML .....	13
3.2	CSS .....	14
3.2.1	Selektory CSS .....	14
3.2.2	Pravidla CSS .....	15
3.2.3	Vkládání CSS do webové stránky .....	15
3.2.4	Souhrn .....	15
3.3	Javascript .....	16
3.3.1	Úvod .....	16
3.3.2	Alternativa Javascriptu .....	16
3.3.3	Využití Javascriptu .....	16
3.3.4	Reakce na události .....	17
3.3.5	Vypnutý Javascript .....	18
3.3.6	Implementace Javascriptu ve webových prohlížečích.....	18
3.3.7	Javascriptové frameworky .....	19
<b>4</b>	<b>Webové galerie využívající Javascript</b> .....	<b>20</b>
<b>5</b>	<b>Generátory webových galerií</b> .....	<b>21</b>
5.1	Online generátory .....	21
5.1.1	Highslide JS .....	21
5.2	Softwarové generátory.....	24
5.2.1	STGThumb .....	24
5.2.2	LightBox Video Web Gallery Creator.....	26

5.2.3	VisualLightBox .....	28
5.2.4	jAlbum .....	30
<b>6</b>	<b>Programovací jazyky vhodné pro vývoj softwarových generátorů .....</b>	<b>33</b>
6.1	Vlastnosti objektově orientovaných jazyků .....	33
6.2	C++ .....	34
6.3	Java .....	34
<b>7</b>	<b>kGenerator .....</b>	<b>36</b>
7.1	kGallery .....	36
7.1.1	Ruční použití javascriptové aplikace .....	36
7.1.2	Hash v URL adrese .....	37
7.1.3	Další užitečné funkce .....	38
7.1.4	Shrnutí .....	38
7.2	kGenerator – generátor galerie kGallery .....	38
7.2.1	Práce s programem .....	39
7.2.2	Shrnutí .....	40
7.3	Možná zlepšení do budoucna .....	41
<b>8</b>	<b>Porovnání výše zmíněných generátorů galerií.....</b>	<b>42</b>
<b>9</b>	<b>Závěr .....</b>	<b>44</b>
	<b>Literatura .....</b>	<b>45</b>
	<b>Příloha A – Ukázka validního HTML dokumentu.....</b>	<b>47</b>
	<b>Příloha B – Javascript a přehled událostí .....</b>	<b>48</b>
	<b>Příloha C – Uživatelské rozhraní kGallery .....</b>	<b>51</b>
	<b>Příloha D – Obsah CD přiloženého k bakalářské práci.....</b>	<b>54</b>



## Seznam zkratk

HTML	HyperText Markup Language
XHTML	Extensible HyperText Markup Language
CSS	Cascading Style Sheets
PHP	Hypertext Preprocessor
API	Application Programming Interface
FTP	File Transfer Protocol

## Seznam obrázků

Obrázek 1 – Náhled webového rozhraní projektu Highslide .....	23
Obrázek 2 – Náhled uživatelského prostředí programu STGThumb .....	26
Obrázek 3 – Náhled uživatelského prostředí programu LightBox Video Web Gallery Creator .....	28
Obrázek 4 – Náhled uživatelského prostředí programu VisualLightBox.....	30
Obrázek 5 – Náhled uživatelského prostředí programu jAlbum .....	32
Obrázek 6 – Náhled uživatelského prostředí programu kGenerator .....	40

## Seznam tabulek

Tabulka 1 – Porovnání dostupných generátorů .....	42
---	----

## 2 Technologie zobrazování obrázků na webu

Digitální fotoaparát v dnešní době pořídíte i za částku, kterou necháte na benzínové pumpě při natankování plné nádrže. V posledních letech se znatelně zvýšil počet uživatelů digitálních fotoaparátů. S tímto trendem se také zvyšuje počet lidí, kteří chtějí své fotografie prezentovat na Internetu. Ovšem své obrázky nechtějí prezentovat jen fotografové, ale například grafik také uvítá možnost zveřejnění své práce, aby měl větší šanci na získání nových zakázek. Nyní se nabízí otázka: „*Jak obrázky publikovat?*“

Na Internetu existuje nesčetné množství serverů pro sdílení souborů, kam můžete obrázky bezplatně nahrát a následně je šířit pomocí unikátního odkazu, který každý obrázek dostane. Toto řešení ovšem není moc praktické z hlediska uživatele, který si bude obrázky prohlížet. Musí na každý odkaz kliknout, aby se mu zobrazil obrázek, poté se vrátit zpět a navštívit odkaz další. Tento nešvar eliminují servery, které se přímo specializují na sdílení obrázků, takže poskytují uživateli jistý komfort při prohlížení. Umožňují prohlížení obrázků pomocí tlačítek „*Následující*“ a „*Předchozí*“, prohlížení ve smyčce, atd. Tento způsob sdílení obrázků je v dnešní době velmi využíván, ale někteří uživatelé, respektive správci stránek, chtějí víc. Chtějí vlastní galerii a nemít vzhled stejný s milionem dalších uživatelů, kteří vystavují obrázky na stejném serveru. Těmto uživatelům nezbývá nic jiného, než si vytvořit galerii vlastní.

Pro poslední zmíněnou kategorii uživatelů existuje několik možností, jak své obrázky prezentovat, respektive jakou zvolit technologii pro prezentování obrázků. Tyto technologie se liší jak z hlediska náročnosti implementace, tak z hlediska správy, tak z hlediska komfortu prohlížení obrázků.

Nejjednodušší možností je prezentace obrázků pomocí statických HTML stránek. Pro správce stránek to představuje pouze základní znalosti v oblasti HTML, případně CSS, ručně upravené obrázky v obrázkovém editoru a následné nahrání na FTP server. Další výhodou tohoto řešení je jistota, že se galerie zobrazí v každém prohlížeči, pokud je napsaná validně. Mezi nevýhody tohoto jednoduchého řešení patří nutnost načítat celou stránku s veškerým obsahem znovu při každém zobrazení nového obrázku. To je problém zejména u datově omezeného připojení k Internetu. Další nevýhodou je správa galerie. Přidání obrázku do galerie obnáší ruční úpravu HTML souborů, což může někdy zabrat mnoho drahocenného času. Stejný problém nastává i u odebrání obrázků z galerie. Poslední nevýhoda je nemožnost oživit galerii nějakými efekty.

Pro zjednodušení správy galerie se nabízí možnost využití programovacích jazyků vhodných pro generování dynamického obsahu webových stránek, například PHP. Tato varianta je oproti předchozí variantě velmi náročná z hlediska implementace. Naučit se programovací jazyk PHP není otázka jednoho večera. Tato náročnost se dá eliminovat tím, že oslovíte jednu z mnoha firem s vaším požadavkem a oslovená firma se postará o vyhotovení aplikace ušité přímo na míru vašim stránkám, respektive vaší galerii. Pro uživatele, který si přišel na váš web prohlédnout obrázky, to nepřináší žádnou změnu oproti předchozí variantě. Velké zjednodušení nastává z pohledu správce galerie, který

může jednoduše přidávat a odebírat obrázky pomocí administračního rozhraní, které musí být navrženo dle požadavků galerie.

Využití PHP je z pohledu správce stránek velké zjednodušení, nicméně pro uživatele to nepřináší žádné zlepšení, stále musí načítat celý obsah stránky znovu při každém zobrazení nového obrázku. Aby se nemusel celý obsah stránky načítat pokaždé znovu, potřebujeme nějaký nástroj, který umožní dynamickou změnu obsahu webové stránky bez kompletního načtení. To znamená, že potřebujeme nástroj, který pracuje na straně klienta v internetovém prohlížeči a ne na straně serveru, jako je tomu u PHP. Tímto nástrojem je programovací jazyk Javascript. Javascript umožňuje změnu obsahu webové stránky bez opětovného načítání a tudíž uživatel dostane větší komfort při prohlížení obrázků. Použití Javascriptu přináší mnoho výhod, ale, jak už to v životě chodí, nic není dokonalé a Javascript není výjimkou. O výhodách a nevýhodách Javascriptu si povíme dále.

Předmětem této bakalářské práce je programovací jazyk Javascript, HTML a CSS, takže se zde dále nebudeme zabývat variantou použití PHP.

## 3 Základní pojmy

### 3.1 HTML

HTML je zkratka z názvu HyperText Markup Language. Jedná se o značkovací jazyk určený pro tvorbu internetových stránek. HTML bylo vyvinuto ze staršího jazyka SGML (Standard Generalized Markup Language).

Pomocí HTML jazyka se vytváří statické webové stránky. Tyto webové stránky nejsou nic jiného než textové soubory napsané podle určitých pravidel, která říkají, jakým stylem zapisovat HTML značky neboli tagy a jejich atributy. Každý tag má přesně daný význam a poskládáním těchto tagů, do určité textové formy, vznikne webová stránka. Pomocí atributů jednotlivých tagů můžeme měnit jejich vlastnosti.

#### 3.1.1 HTML tagy

Tagy mohou být buď párové, nebo nepárové. Párový tag znamená, že do HTML dokumentu vložíte počáteční tag, mezi to nějaký text, či obrázek a poté ukončíte koncovým tagem. Zde máte jednoduchý příklad párového tagu v HTML:

```
<cite>Citovaný text</cite>
```

Nepárový tag znamená, že se žádným způsobem neukončuje. Příklad tagu, který udělá horizontální čáru přes celou šířku elementu, do kterého bude vložen:

```
<hr>
```

Výše byl zmíněn pojem „*element*“. V případě HTML je element celý tag i se svým obsahem. Když se podíváte na výše zmíněný příklad citovaného textu, tak celý text, včetně začínajícího a ukončovacího tagu, je jeden element.

#### 3.1.2 Vnořené elementy

Elementy v HTML můžeme vnořovat do sebe, ale nesmí se křížit. To znamená, že v případě vnořených elementů musíte nejprve ukončit element vnitřní a poté element vnější. Internetové prohlížeče tuto chybu ignorují a většinou si s ní dokáží poradit, ale nedá se na to spoléhat a je lepší psát stránky validní.

#### 3.1.3 Validita HTML dokumentu

Validita dokumentu spočívá v tom, že jsou dodržena určitá pravidla pravidla pro psaní webových stránek, respektive HTML dokumentu. Mezi tato pravidla patří nekřížení elementů mezi sebou, správná struktura HTML dokumentu, atp.

I zkušený vývojář někdy udělá chybu v HTML kódu, a proto existují validační nástroje, kam vložíte přímo HTML kód, nebo jen odkaz na stránku, kterou chcete validovat. Po validaci zdrojového kódu dostanete výpis chyb, které by měly být odstraněny.

Nejrozšířenější nástroj pro validaci je od W3C, neboli World Wide Web Consortium. W3C je mezinárodní konsorcium, které vyvíjí webové standardy. Nástroj pro validaci HTML dokumentu naleznete na stránkách <http://validator.w3.org>, kam můžete

vložit adresu stránky, kterou chcete validovat, nebo přepnout na záložku „*Direct input*“ a vložit tam zdrojový kód celé stránky.

Validace dokumentu je jedna věc, ale druhá věc je ta, že ani s naprosto validním dokumentem není zaručeno, že se stránka zobrazí ve všech prohlížečích stejně. Každý prohlížeč zpracovává stránku jiným způsobem a u některých elementů dochází k odlišnostem. Například zmíněná horizontální čára pomocí tágu „*hr*“ vypadá v každém prohlížeči jinak a dokonce má i různou tloušťku. Při vytváření webu se doporučuje provádět kontrolu vzhledu ve všech nejpoužívanějších prohlížečích jako je Internet Explorer, Opera, Google Chrome, Firefox, atd. Tím zjistíte, zda se stránky zobrazují dostatečně správně, případně můžete stránku poupravit.

Příklad validního dokumentu naleznete v příloze A

### 3.1.4 XHTML

Extensible HyperText Markup Language, neboli XHTML, je značkovací jazyk založený na XML se zpětnou kompatibilitou s HTML. Původně mělo být XHTML nástupcem staršího HTML, které se dlouho nevyvíjelo, ale v dnešní době se stále vyvíjí oba standardy a bez problému koexistují. Podpora v moderních prohlížečích je na stejné úrovni pro HTML i XHTML.

Hlavním úkolem XHTML bylo nastolit jasně daná pravidla pro psaní validních webových stránek. HTML byl jazyk velmi tolerantní, až to bylo někdy nepřehledné, proto vzniklo XHTML, které je podstatně striktnější.

V HTML bylo možné psát nepárové značky tak, že prostě nebyly ukončené. V XHTML neexistuje neukončená značka. To znamená, že i nepárové značky musí být ukončeny. Lze to vyřešit dvěma způsoby, které vidíte v následujícím příkladu.

```
<br />  
<br></br>
```

Obě předchozí varianty jsou správné, ale nejčastěji se používá varianta první.

V HTML bylo možné psát atributy a jejich parametry velkými i malými písmeny a také bylo možné psát parametry bez uvozovek. V XHTML musí být veškeré atributy a jejich parametry psány malým písmem, navíc musí být parametry uzavřeny do uvozovek.

V XHTML jsou zakázány všechny atributy, které určují vzhled jednotlivých elementů, protože se počítá s použitím CSS. Pokud potřebujete vyvíjet web pro starší prohlížeče nepodporující CSS, musíte použít standard XHTML 1.0 Transitional, které je tolerantní vůči atributům určujícím vzhled elementu.

Mezi další rozdíly XHTML oproti HTML patří MIME typ odesílaný na server, kde u HTML se jedná o hodnotu „*text/html*“ a u XHTML je to „*application/xhtml+xml*“. XHTML dokument by měl začínat XML deklarací, není-li kódován v UTF-8 a mnoho dalších rozdílů, které ovšem nejsou tak podstatné.

Použité zdroje: (Grimmich, 2008), (Wikipedie, 2004), (Žďárek, 2011)

## 3.2 CSS

Kaskádové styly (CSS - Cascading Style Sheets) je jazyk, který se používá pro stylování HTML, XHTML a XML dokumentů. Hlavním důvodem vývoje tohoto jazyka bylo oddělit veškeré prvky, které stylují dokument, od struktury obsahu. Původní koncept HTML spočíval v tom, že v sobě nesl obsah i styl, jak se má prvek zobrazovat. Možnosti byly velmi omezené, dala se upravovat barva písma, velikost a font. Navíc se musel každý prvek v dokumentu stylovat zvlášť i když se měly zobrazit stejně. To při případné změně znamenalo, že musíte projít celý web a upravit staré hodnoty na hodnoty nové.

Smyslem CSS je rozšíření možností, které může vývojář webu nastavit. Dále přináší možnost zjednodušení v podobě napsání jednoho CSS dokumentu, který pak využijete na všech stránkách webu. To znamená, že při změně vzhledu na webu upravíte pouze pár řádků v jednom CSS souboru a změna se provede pro celý web. Například se rozhodnete, že nechcete mít všechny nadpisy „*h1*“ černé, ale zelené a na to postačuje pouze jeden řádek zápisu CSS stylu v externím souboru. V samotném HTML bylo nutné upravit každý nadpis, který na stránce máte, zvlášť, a to u rozsáhlejších portálů může zabrat i několik hodin. CSS styl je také mocný nástroj ve spojení s Javascriptem, kdy můžete pomocí akcí měnit hodnoty CSS stylů u jednotlivých elementů v dokumentu.

### 3.2.1 Selektory CSS

Pro určení prvku, nebo množiny prvků, které chceme stylovat, používáme selektory, kterých existuje několik:

- *span* – Selektor, který říká, že pravidlo bude použito pro všechny elementy *span*.
- *span a* – Toto platí pro všechny elementy *a*, které jsou v elementu *span* a to v jakékoli hloubce.
- *span>a* – Toto pravidlo opět platí pro všechny elementy *a* v elementu *span*, ale pouze ty elementy *a*, které jsou přímo v elementu *span* a ne elementy, které jsou dále vnořené do jiných elementů.
- *.nazev\_tridy* – Pravidlo, které platí pro všechny elementy, které mají atribut *class* nastavený na „*nazev\_tridy*“.
- *#nazev\_identifikatoru* – V tomto případě pravidlo platí pro všechny elementy, které mají atribut *id* nastaven na hodnotu „*nazev\_identifikatoru*“. Identifikátorů můžeme vytvořit, kolik chceme, ale každý by se měl na jedné stránce vyskytovat pouze jednou.

### 3.2.2 Pravidla CSS

Pro vytvoření pravidla zobrazování dokumentu napíšeme do CSS souboru selektor, který chceme použít, dále vytvoříme blok pomocí složených závorek a do tohoto bloku vepíšeme pravidla pro daný selektor. Příklad pravidla:

```
p {
    margin: 0px;
    padding: 0px;
    background-color: blue;
}
```

Výše uvedené pravidlo nám říká, že všechny elementy p, nebudou mít žádné vnitřní i vnější odsazení a pozadí bude modré.

### 3.2.3 Vkládání CSS do webové stránky

CSS styl se dá do HTML dokumentu vložit několika způsoby. První způsob je přímé vkládání CSS stylu k jednotlivým elementům. Toto řešení je nepraktické v případě, že stejný styl budete používat pro více elementů, protože je pak nutné při případné změně upravit každý element zvlášť.

```
<p style="margin: 0px; padding: 0px; background-color: blue;">upravený text</p>
```

Další způsob už je praktičtější. Opět se pravidlo vkládá přímo do HTML dokumentu, ale tentokrát takovým způsobem, že do hlavičky HTML zapisujeme pravidla pro jednotlivé selektory.

```
<style type="text/css">
  p {
    margin: 0px;
    padding: 0px;
    background-color: blue;
  }
</style>
```

Předchozí způsob nás omezuje pouze na jednu HTML stránku, takže si ukážeme poslední 2 způsoby, které importují externí CSS soubory do HTML dokumentu. Tyto soubory můžeme importovat do každého HTML dokumentu na webu a případná změna v CSS souboru se promítne na celém webu.

```
<link rel="stylesheet" type="text/css" href="muj_styl.css" />
Nebo
```

```
<style type="text/css">@import
url("http://www.nejakyserver.cz/muj_styl.css");</style>
```

Poslední zmíněný způsob importu CSS souboru umožňuje použití stylu, který je umístěn na jiném serveru, než webové stránky, pro který chceme styl použít.

### 3.2.4 Souhrn

V dnešní době se pro stylování stránky používá výhradně CSS. Standard XHTML to dokonce vyžaduje, protože atributy a tágy, které měly na starosti stylování dokumentů, jsou zakázány, tudíž musí být použit CSS styl. CSS přináší velkou řadu výhod, jako



například mnohem snazší úprava vzhledu webových stránek, možnost mít vše, co se týče vzhledu, v jednom souboru, mnohem větší možnosti formátování atd.

Nic není dokonalé a také CSS má jednu zásadní nevýhodu. Lépe řečeno, nejedná se o nevýhodu CSS, ale o způsobu implementace CSS do jednotlivých webových prohlížečů. Každý prohlížeč obsahuje nějaké chyby v implementaci CSS a mnohdy se stane, že se stejný web zobrazuje v různých prohlížečích odlišně. To se dá napravit pomocí takzvaných CSS *hacků*, u kterých ale bohužel nemáte jistotu, že budou fungovat v nové verzi prohlížeče.

Stejně tak, jako musí být validní HTML stránka, musí být validní i CSS dokument. Opět existuje CSS validátor, který zjistí možné nedostatky v CSS dokumentu. Najdete jej na stránkách organizace w3c, nebo přímo na tomto odkazu <http://jigsaw.w3.org/css-validator/>.

Použité zdroje: (Grimmich, 2008), (Wikipedie, 2004), (Žďárek, 2011)

### **3.3 Javascript**

#### **3.3.1 Úvod**

Jak již bylo řečeno, Javascript je programovací jazyk, který pracuje na straně klienta. Je to jediný programovací jazyk, který má přímý přístup k jednotlivým elementům na stránce a umožňuje dynamicky měnit jejich obsah na základě nějaké akce. Naučit se programovat v jazyku Javascript není tak náročné, jako se naučit programovat, například, v jazyku Java, ale také to není zcela triviální, zvláště nemáte-li žádné zkušenosti s programováním. V předchozí větě byl zmíněn programovací jazyk Java. Jedná se o jazyk, který nemá nic společného s Javascriptem a jsou zcela odlišné. Javascript byl primárně vytvořen pro implementaci ve webových prohlížečích, kdežto Java je všeobecný programovací jazyk.

#### **3.3.2 Alternativa Javascriptu**

Pokud nechcete používat Javascript, existuje alternativa zvaná VBScript, která také funguje ve webovém prohlížeči a umožňuje vytvářet dynamické webové stránky. VBScript byl vyvinut společností Microsoft a funguje pouze na operačním systému Windows v kombinaci s prohlížečem Internet Explorer. Pokud tedy chcete na svých stránkách navštívníky i s jinými operačními systémy, než je Windows, případně s jinými internetovými prohlížeči, zůstaňte věrní Javascriptu. Zdroj (THAU, 2009).

#### **3.3.3 Využití Javascriptu**

Jazyk Javascript se nepoužívá pouze na obrázkové galerie. Tento jazyk se nejvíce využívá při práci s formulářovými poli. Můžete dynamicky přidávat, či odebrat pole. Obrovská výhoda je možnost kontroly vyplněného formuláře před samotným odesláním na server. Typický příklad je formulářové pole pro zadání e-mailové adresy, kdy se kontroluje, zda je e-mail v platném tvaru. Výhoda kontroly formuláře ještě před odesláním spočívá v tom, že uživatel nemusí čekat na odpověď serveru, který vyhodnotí, zda je

formulář správně vyplněn a poté pošle uživateli odpověď, ale kontrola je provedena ještě před odesláním, tudíž je uživatel informován téměř okamžitě. Kromě ušetřeného času také redukuje množství přenesených dat po síti.

Kontrola formulářů na straně klienta, před odesláním, je velmi dobrý nástroj, nicméně se na tuto kontrolu nemůžeme, jako správci stránek, spolehnout. A to ze dvou důvodů. Zkušenější uživatel si umí vytvořit formulář vlastní a následně ho odeslat na náš server. To znamená, že kontrola na straně klienta selže, respektive se vůbec nekoná. Další důvod, proč se nespolehat na kontrolu na straně klienta je ten, že uživatel si může funkci Javascriptu ve svém prohlížeči jednoduše vypnout. Takže se opět žádná kontrola u klienta nekoná a musí být provedena na straně serveru. Z výše uvedeného plyne, že veškerou kontrolu formulářů u klienta provádíme jen proto, abychom snížili zátěž serveru a uživateli ušetřili trochu času v případě, že formulář nevyplní správně. Konečnou kontrolu musíme provést i na serveru, aby se předešlo chybně zpracovanému formuláři v případě, že uživatel vypne Javascript.

### 3.3.4 Reakce na události

Kontrolou vyplněných formulářů zdaleka možnosti Javascriptu nekončí. Další věc, která způsobila velkou oblíbenost tohoto programovacího jazyku, je možnost reakce na události vyvolané uživatelem. Můžete reagovat například, když uživatel stiskne tlačítko myši, nebo klávesnice, dále můžete vyvolat nějakou akci při posunutí stránky, atp.

Reakci na událost přidáme elementu HTML nastavením patřičného atributu. Například, chceme-li vyvolat reakci při stisku myši, na určitý element, v podobě vyskakovacího okénka, které ukáže hlášku, můžeme to provést více způsoby. První způsob se používá pro velmi jednoduché reakce na události a vypadá následovně.

```
<p onclick="window.alert('Právě jste kliknul na odstavec');">Kliknutím na tento odstavec vyvoláte reakci</p>
```

Předchozí ukázka části zdrojového kódu HTML stránky způsobí zobrazení okna s nápisem „Právě jste kliknul na odstavec“ v případě, že uživatel klikne na text, který odstavec obsahuje. Jedná se o přímý zápis kódu Javascriptu do atributu „*onclick*“. Druhý způsob, jak můžeme přiřadit reakci nějaké události je, že si vytvoříme funkci a při patřičné události ji zavoláme pomocí názvu dané funkce. Použití v praxi znázorňuje následující ukázka zdrojového kódu.

```
<p onclick="kliknutiNaOdstavec()">Kliknutím na tento odstavec vyvoláte reakci</p>
<script type="text/javascript">
    function kliknutiNaOdstavec() {
        window.alert('Právě jste kliknul na odstavec');
    }
</script>
```

Po kliknutí na odstavec s textem je zavolána metoda s názvem „*kliknutiNaOdstavec()*“, která provede to, co jí řeknete, v tomto případě opět vyskočí hláška s textem. Tuto metodu lze vložit přímo do HTML stránky, nejčastěji se vkládá do hlavičky a musí být uvozena tágem „*script*“ s atributem „*type*“ nastaveným na hodnotu „*text/javascript*“, jak ukazuje

předchozí část zdrojového kódu. Další možnost je vložit metodu do externího souboru, který následně importujete do HTML stránky. Import souboru se opět provádí pomocí tagu „*script*“ a vypadá to následovně.

```
<script src="soubor_javascriptu.js" type="text/javascript"></script>
```

Varianta s externím souborem se používá nejčastěji, protože případný zdrojový kód Javascriptu, vložený přímo do HTML stránky, může překážet při správě webu. Toto řešení má navíc výhodu v tom, že kód nemusíte kopírovat na každou HTML stránku zvlášť, ale pouze znovu importujete soubor a případné změny v javascriptu provádíte jen na jednom místě.

Všechny události, které lze pomocí Javascriptu vyvolat, jsou vypsané v příloze B

### 3.3.5 Vypnutý Javascript

Nyní se dostáváme k prvnímu problému při používání Javascriptu na webových stránkách. Možnost vypnutí Javascriptu existuje v každém prohlížeči a někteří uživatelé to využívají. Jedná se sice o zhruba jedno procento lidí, ale z deseti tisíc návštěv na webových stránkách denně, je to sto uživatelů, což není malé číslo. Z tohoto důvodu by měly být stránky, které používají Javascript navrženy tak, aby bylo co nejméně nápadné, když má uživatel Javascript vypnutý. To znamená, že by stránka měla zůstat stále čitelná i bez zapnutého Javascriptu.

V případě navržení stránek vyžadujících spuštěný Javascript, můžete uživatele upozornit, že se mu nyní obsah nemusí zobrazit korektně. Pak nastává riziko, že si uživatel návštěvu takovýchto stránek rozmyslí a opustí je.

Možnost vypnutí Javascriptu není jediná překážka pro programátory.

### 3.3.6 Implementace Javascriptu ve webových prohlížečích

Práce s Javascriptem by nebyla až tak složitá záležitost, kdyby ovšem implementace v jednotlivých prohlížečích nebyly odlišné. Co to znamená? Znamená to, že se dá snadno napsat aplikace, která funguje pouze v prohlížeči, pro který byla optimalizována a v dalších prohlížečích nepoběží. Implementace v jednotlivých prohlížečích provádí některé akce odlišným způsobem, než implementace v prohlížečích ostatních. Pokud chcete web, který je přístupný co největšímu množství uživatelů, měla by být vaše aplikace validována pro všechny nejpoužívanější webové prohlížeče.

Pokud nemáte jistotu, že je daný objekt implementován ve všech prohlížečích, můžete nejdříve provést test, zda objekt v prohlížeči uživatele, který aktuálně navštívil vaši stránku, implementován je a na základě výsledku testu přizpůsobit chování aplikace. Toto řešení je doporučované, protože uživatel může být upozorněn, že jeho prohlížeč není podporován. Test, zda je možné použít v aktuálním prohlížeči objekt „*images*“ vypadá následovně.

```
if (document.images) {  
    // kód pracující s objektem „images“  
} else {
```

```
    // alternativní kód, který se obejde bez objektu images  
}
```

### 3.3.7 Javascriptové frameworky

Pro usnadnění programování v jazyku Javascript byly vytvořeny různé frameworky. Tyto frameworky se snaží zapouzdřit rozdíly mezi prohlížeči, a tím ušetřit programátorovi práci s optimalizací pro nepoužívanější webové prohlížeče. Další výhodou těchto frameworků je množství funkcí, které přidávají nad rámec „obyčejného“ Javascriptu. Umožňují přidat různé efekty na webové stránky pomocí pár řádků kódu a nemusíte celý efekt psát sáhodlouze samy. Malou nevýhodou je, že při použití nějakého z frameworků, jej musíte importovat do webové stránky, ve které byl použit, jinak výsledná aplikace nebude fungovat.

Mezi nepoužívanější frameworky patří například jQuery, Moo Tools a Prototype. Samozřejmě, frameworků existuje obrovské množství a každý programátor preferuje jiný.

Použité zdroje: (Grimmich, 2008), (Žďárek, 2011), (Milde, 2011)

## 4 Webové galerie využívající Javascript

Webové galerie založené na Javascriptu dodávají prohlížení obrázků na webu další rozměr. Prohlížení je mnohem zábavnější a hlavně rychlejší. Ale programování vlastní aplikace k zobrazování obrázků na webu se stává náročným úkolem, chceme-li obrázky ukázat co nejširšímu publiku. Kromě znalosti samotného programovacího jazyka Javascript, potřebujete také nainstalovat všechny prohlížeče, pro které bude aplikace vyvíjena a každý kousek kódu, který napíšete, patřičně vyzkoušet ve všech těchto prohlížečích. Při programování javascriptové galerie není, kupodivu, problém jen s Javascriptem, ale i se stylováním pomocí CSS. Zde platí stejný problém, jako u Javascriptu a to, že každý prohlížeč se opět chová odlišně. Mnohdy je optimalizace CSS stylu obtížnější než optimalizace samotné aplikace v jazyku Javascript.

Pokud chcete předejít pracnému vývoji kvalitní aplikace, nabízí se možnost implementovat na web nějaké z hotových řešení, kterých na internetu naleznete velké množství. Tyto aplikace jsou většinou vyvíjeny zkušenými programátory a jsou optimalizovány pro většinu moderních prohlížečů. Zároveň jejich vývoj nebývá jednorázový, ale vycházejí novější verze reagující na zpětnou vazbu od uživatelů, kteří nahlásí chyby, na které narazili při prohlížení obrázků.

Javascriptové aplikace pro webové galerie dostupné na Internetu podléhají licenci, kterou určuje autor projektu. Na to je nutné dát si pozor, protože dané licence se musíte držet. Některé aplikace jsou zcela zdarma. Jiné jsou zdarma jen pro osobní účely, a pokud chcete aplikaci využít pro komerční účely, musíte si zaplatit licenci.

V případě, že jste náročný uživatel, či firma a potřebujete aplikaci na míru, protože vám žádná z dostupných na Internetu plně nevyhovuje, můžete oslovit firmu, která se postará o sestavení aplikace dle přesných požadavků a kritérií. Tato varianta je nejlepší, ale bývá samozřejmě také nejnákladnější.

## 5 Generátory webových galerií

Hledáte-li snadný způsob zveřejňování obrázků na Internetu, můžete sáhnout po nějakém generátoru webových galerií. Jedná se o program, který vám vytvoří jednoúčelové stránky pro prezentaci obrázků na webu. Nemusíte znát žádný programovací jazyk. Jediné, co je od vás vyžadováno, je základní znalost práce na počítači. Generátor může být buď v podobě software na počítači, nebo webové rozhraní.

Tyto generátory většinou pracují na stejném principu. Zvolíte si obrázky, které chcete zveřejnit, nastavíte si různé volby, které generátor umožňuje a následně je vygenerována galerie. Generátory se liší množstvím možností, které můžete nastavit, a hlavně se liší výslednou galerií. Některé generátory generují galerie založené pouze na HTML a CSS, jiné generují i webové galerie založené na Javascriptu. Záleží na požadavcích, které máte a na základě toho vybrat vhodný generátor.

Do zvláštní skupiny generátorů patří ty, které ukládají obrázky rovnou na nějaký, předem určený, server, kde jsou uloženy. Typickým příkladem je český projekt zvaný Rajče<sup>1</sup>. Zaregistrujete se na stránkách, kde vám je přidělen prostor pro vaše obrázky, v tomto případě je prostor datově neomezený. Stáhnete si program, který spolupracuje s projektem Rajče, navolíte obrázky, které se následně samy upraví a nahrají na server [www.rajce.indes.cz](http://www.rajce.indes.cz). Tyto obrázky jsou uveřejněny na stránkách projektu a vy je můžete šířit pomocí odkazu, který je automaticky vygenerován. Dalším podobným projektem je Picasa<sup>2</sup> od společnosti Google. Tato skupina generátorů galerií byla zmíněna jen pro úplnost a dále se jimi nebudeme zabývat.

### 5.1 Online generátory

Nespornou výhodou online generátorů je fakt, že jsou přístupné odkudkoli, kde je Internet. Tuto výhodu využijete zejména, když jste na dovolené, nafotíte nějaké fotografie a chcete je co nejdříve publikovat. Zajdete do internetové kavárny, na stránku nějakého projektu s online generátorem a necháte vygenerovat galerii. Další výhodou je ta, že nemusíte do počítače nic stahovat, nebo zdlouhavě instalovat. To se hodí v případě, že chcete jednorázově vytvořit galerii a víte, že to nikdy více nebudete potřebovat. Online řešení generátoru galerií ovšem přináší i jednu zásadní nevýhodu. Veškeré obrázky, které chcete vložit do galerie, musí být nejprve nahrány na server, kde jsou následně zpracovány. To s sebou přináší mnohdy velký objem dat, který musí být přenesen mezi uživatelem a serverem. Některé fotografie mohou být velké i několik megabajtů a při pomalejším připojení může nahrávání trvat i několik hodin.

#### 5.1.1 Highslide JS

Highslide<sup>3</sup> JS je nepochybně jeden z největších hráčů na trhu javascriptových galerií. Javascriptové aplikace, které nabízí, jsou propracované do nejmenšího detailu.

---

<sup>1</sup> Obrázková galerie od společnosti MAFRA a. s. <http://www.rajce.idnes.cz/>

<sup>2</sup> Obrázková galerie od společnosti Google [www.picasaweb.google.com](http://www.picasaweb.google.com)

<sup>3</sup> Stránky projektu Highslide JS <http://highslide.com>

Seznam kompatibilních prohlížečů pokrývá všechny dnes používané a dokonce zaručují kompatibilitu i se staršími prohlížeči, jako je například Internet Explorer 5.5, Firefox 1.5, atd. Tento projekt je opravdu zdařilý a autor si ho také cení. Pokud chcete nějaký skript pouze vyzkoušet, nebo jej použít pro nekomerční účely, neplatíte žádnou částku. V případě, že chcete skript použít na komerčních internetových stránkách, musíte si koupit licenci, která vás vyjde na 29<sup>4</sup> dolarů. Tato licence vám platí na jeden web. Jste-li vývojář a víte, že skripty Highslide budete využívat častěji na několika webech, vyplatí se vám zvolit licenci nelimitovanou počtem webů. Tato licence stojí ovšem poněkud více, 179 dolarů.

Galerii založenou na Highslide skriptech můžete vytvořit zcela ručně tak, že si na stránkách projektu nakonfigurujete soubor Javascriptu dle vašich požadavků a ten si následně stáhnete. V tomto případě musíte ručně upravit obrázky, vytvořit vlastní HTML stránku, importovat soubor javascriptu a mnoho dalších věcí. Toto řešení využijete zejména v případě, že vytváříte rozsáhlejší webové stránky a ne pouze jednocelovou prezentaci obrázků.

Nechcete-li zdlouhavě upravovat obrázky, vytvářet HTML stránku, stylovat rozmístění miniatur na webu, nejspíše zvolíte editor<sup>5</sup> galerie, který Highslide nabízí ve verzi online. Při vstupu na stránku s editorem se vám zobrazí krátký průvodce, který vysvětluje základní kroky nutné ke zdárnému dokončení galerie. Po potvrzení tlačítkem „*Get started*“ máte v levé části obrazovky záložky, které postupně projdete, abyste nastavili parametry výsledné galerie. V pravé části, která dominuje obrazovce, vidíte náhled výsledné galerie.

Na úvodní záložce „*General*“ volíte jazykovou verzi galerie a základní styl, od kterého se bude výsledná galerie odvíjet. Další záložka, „*Gallery*“, slouží především pro výběr obrázků, které chceme prezentovat. Dále se zde volí vzhled ovládacích tlačítek, informace o počtu obrázků, zda se má zobrazit proužek s miniaturami při otevřeném náhledu obrázku a chování galerie. Další dvě záložky slouží pro nastavení vzhledu galerie, jako jsou například ovládací tlačítka, styl rámečku obrázku a mnoho dalších možností. Předposlední záložka „*Behavior*“ umožňuje nastavení stylu animace přechodu při spouštění/vypínání galerie, pozice obrázku v okně při otevřeném náhledu, povolení přizpůsobení velikosti obrázku oknu prohlížeči a mnoho dalších možností. Poslední zmíněná možnost nastavení „*přizpůsobení velikosti obrázku oknu prohlížeče*“ je užitečná zejména pro uživatele s menším monitorem, protože se jim obrázek vždy ukáže celý a nemusí rolovat stránku nahoru, dolů. Poslední záložka, „*Overlays*“ umožňuje nastavení pozice titulku a názvu obrázku. Dále pak styl tlačítka pro vypnutí náhledu.

Při procházení jednotlivých záložek konfigurace máte k dispozici pohled na aktuální podobu kódu Javascriptu, CSS stylu a HTML. Jednotlivé zdrojové kódy zobrazíte pomocí tlačítek umístěných v horní části pravého okna editoru. Dokonce máte možnost

---

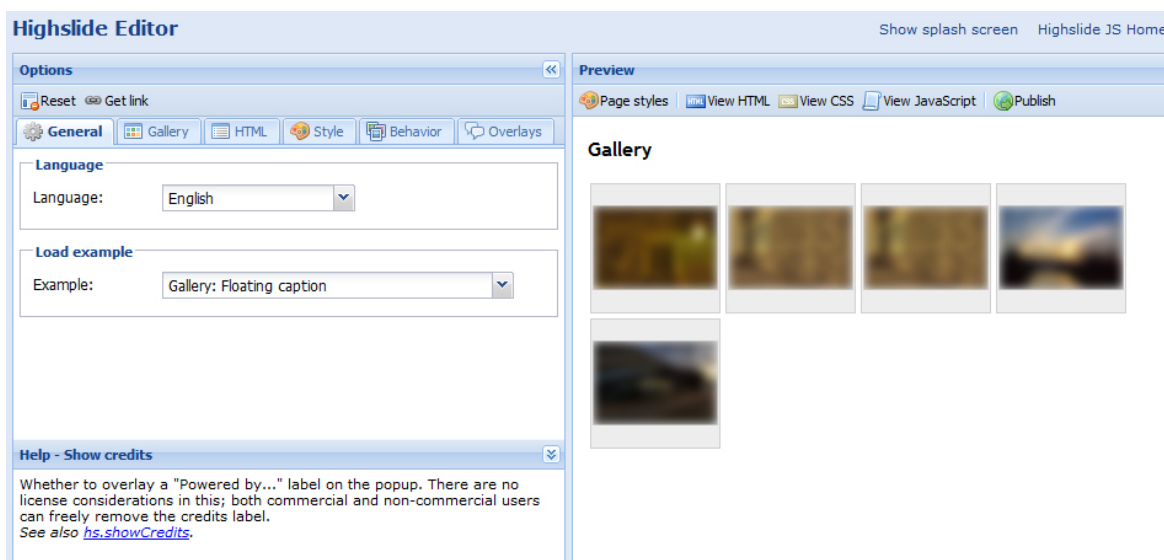
<sup>4</sup> Srpen 2011

<sup>5</sup> Online generátor webových galerií Highslide <http://www.highslide.com/editor/>

provést zásah do CSS stylu galerie kliknutím na tlačítko „Page styles“, kam můžete napsat vlastní styl CSS.

Poslední úkol je získání výsledné galerie. Toho docílíme stiskem tlačítka „Publish“. Zde se nám nabízí dvě možnosti. První možností je přímé nahrání na FTP server, což je výborná věc, kterou využijete v případě, že nepotřebujete výslednou galerii u sebe v počítači. Zároveň tím ušetříte množství přenesených dat, protože nahrávání probíhá přímo ze serveru, kde sestavujete galerii na FTP server, který vyplníte. Druhou možností je stažení „zip“ souboru, který obsahuje veškeré soubory galerie, a nahrání na FTP server provedete ručně. Výslednou galerii také můžete vložit do již hotové webové stránky. Pro zjednodušení vám je, při stahování „zip“ souboru, poskytnut průvodce, který ukazuje krok za krokem, co musíte udělat, aby vložení galerie do existující stránky bylo úspěšné.

Následující obrázek zobrazuje náhled webového editoru galerií projektu Highslide (Obrázek 1)



**Obrázek 1 – Náhled webového rozhraní projektu Highslide**

Výsledná galerie pracuje velmi svižně a není autorům co vytýkat. V nastavení ovšem nějaké věci, které by se u takto rozsáhlého projektu mohly objevit, chybí. Například není možnost si v editoru galerie nastavit přesnou šířku stránky, takže se stránka automaticky přizpůsobuje šířce okna. Při vkládání výsledné galerie do existující stránky to není problém, ale v případě, že chcete vygenerovanou galerii pouze jako jednoúčelovou stránku a vyžadujete přesnou šířku galerie, musíte provést zásah do zdrojového kódu. Další věc, která v editoru nenašla své místo, je možnost stránkování. Pokud se chystáte prezentovat 500 obrázků, tak je generátor všechny vloží na jednu stránku.

Více informací se můžete dozvědět na stránkách projektu (Hønsi, a další, 2011).



## 5.2 Softwarové generátory

Jedná se o programy napsané za účelem generování webových galerií. Oproti online generátorům veškerá práce probíhá přímo ve vašem počítači, kde si vyberete fotografie a následně provedete konfiguraci. Softwarové generátory většinou nabízí širší možnosti konfigurace, než generátory online, kde jsou možnosti omezenější.

### 5.2.1 STGThumb

Softwarový generátor, jehož vývoj začal již v roce 2001 a poslední verze (3.10) byla vydána v červnu roku 2011<sup>6</sup>. Program můžete stáhnout z Internetu zdarma pro jeho vyzkoušení a poté zakoupit licenci za 19,95 dolarů. Bez zakoupení licence generátor vkládá do obrázků vodoznak s nápisem „*STGTHUMB*“, takže je výsledná galerie víceméně nepoužitelná.

Tento generátor je navržen pro jednoduché vytvoření obrázkové galerie. Můžete si zvolit ze 4 základních typů galerie, kde pouze typ „*Lytebox*“ funguje ve spojení s Javascriptem. Ostatní možnosti galerie využívají pouze HTML, případně ještě CSS, takže se jimi nebudeme zabývat a zaměříme se na typ „*Lytebox*“.

Pomocí tohoto generátoru můžete prezentovat obrázky mnoha formátů (JPG, JP2, J2K, J2C, TIF, PNG, BMP, PCX, WMF, EMF, ICO, CUR, PSD), které budou při generování galerie převedeny na JPG. Generátor umožňuje prezentaci videí ve formátu „*avi*“ a „*mpg*“, ale to pro naše potřeby není podstatné. Dále podporuje vkládání vlastních vodoznaků do obrázků, které prezentujete. Toho se využije v případě, že mají být obrázky chráněny proti odcizení. Například, jedná-li se o prezentaci obrázků k prodeji.

Celý proces vytváření galerie v tomto programu není obtížný. První věc, která se nabízí, je volba, zda budete prezentovat pouze obrázky, nebo videa, případně oboje. Dále si vyberete adresář s obrázky, které chcete prezentovat a zvolíte „*Scan*“. Vyberou se všechny podporované formáty z adresáře a poté můžete upravit seznam obrázků tím, že označíte obrázek v seznamu vpravo a kliknete na znaménko „*minus*“, čímž obrázek odstraníte ze seznamu. Také můžete obrázky přidat kliknutím na znaménko „*plus*“. Při odstraňování obrázku ze seznamu se program neptá, zda opravdu chcete obrázek vymazat, ale akci provede okamžitě.

Nastavení miniatur je hned další úkon, který se nabízí po volbě obrázků. Nastavit si můžete maximální šířku a maximální výšku miniatury a „*prefix*“ nebo „*suffix*“ názvu miniatury pro odlišení názvů od velkých obrázků. Další volbou je kopírování velkých obrázků do cílového adresáře. Je tu vcelku nepochopitelně a dokonce je ve výchozím stavu odškrtnuta. Pokud tuto možnost nezaškrtnete, tak se vytvoří pouze miniatury a velké obrázky si musíte upravit a přemístit do galerie samy. Vodoznak do obrázků můžete přidat buď vložením textu, nebo sloučením výsledného obrázku s obrázkem vodoznaku. Dále zde máme základní nastavení HTML v podobě výběru typu výsledné galerie. Jak jsem již psal,

---

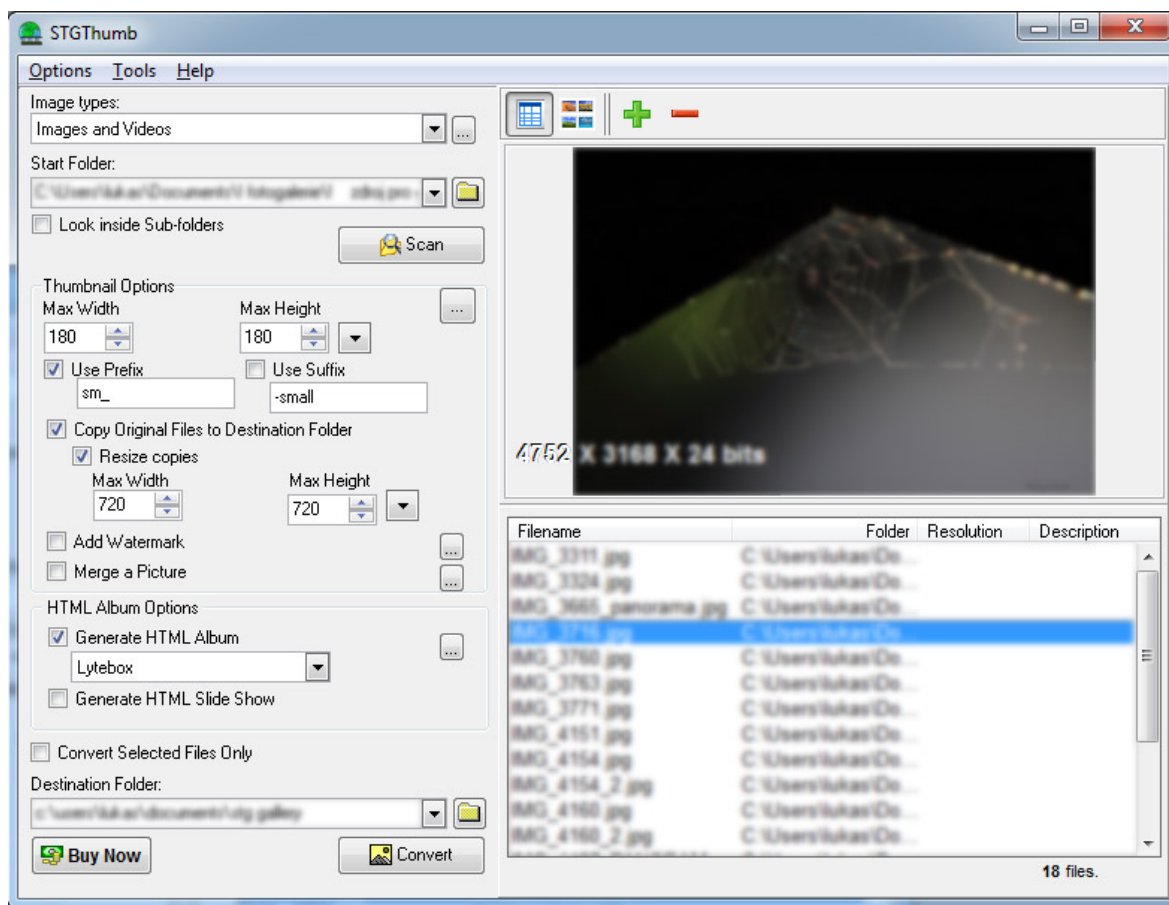
<sup>6</sup> Informace ze dne 31. 7. 2011

nás především zajímá typ „*Lytebox*“. Samozřejmě musí zůstat zaškrtnutá položka generování HTML alba. Možnost generování HTML slideshow je volitelná a znamená to, že se vygeneruje samostatný HTML soubor, který funguje jako samostatná slideshow, ovšem bez možnosti jakéhokoli ovládání (obrázky se mění po určitém intervalu). Poslední položka, kterou je nutné vyplnit pro vytvoření galerie je umístění na disku.

Tato základní konfigurace nepokrývá všechny možnosti programu. V nabídce „*Options*“ máte, kromě jiného, možnost detailněji nastavit výstup. Volbou „*Output settings*“ toto nastavení zobrazíte. Je zde volba kvality výsledných obrázků, možnost zachování proporcí, dále můžete aplikovat filtr pro konvertování obrázků. Vyskytuje se zde i volba přepsání existujících souborů, pokud v cílovém adresáři nějaké existují. Tento dotaz by bylo lepší zobrazit, až když daná situace nastane, protože ne každý uživatel bude detailně zkoumat nastavení a může se stát, že si přepíše existující soubory bez jakéhokoliv upozornění.

Poslední rozšířená konfigurace je opět v nabídce „*Options*“ a je pod názvem „*HTML Configuration*“. Zde na první záložce „*General*“ provedete základní nastavení, jako je počet sloupců na řádku (počet řádků na stránce se nastavuje na jedné z dalších záložek), šířka rámečku kolem celé galerie, zobrazované informace o miniaturách a v neposlední řadě název úvodního (index) souboru. Nastavení, která se ukrývají na další záložce, „*Image Location*“, nejsou zcela dotažená do konce, protože v případě, že si nastavíte relativní umístění obrázků a miniatur do určitého adresáře, tak generátor vygeneruje všechny obrázky do stejného adresáře společně se soubory HTML a dalšími, poté musíte podadresáře vytvořit a vložit do nich soubory ručně. Dalo by se očekávat, že v projektu, který je vyvíjen deset let, budou takovéto případy ošetřeny a automaticky generátor vytvoří potřebné podadresáře, ale není tomu tak. Předposlední záložka „*Multiple pages*“ se dotazuje, zda chcete rozdělit galerii na více stránek. V případě, že chcete galerii rozdělit, nastavte si počet řádků na stránku. Další konfigurační záložka „*Slide Show*“ slouží pro nastavení hodnot automatické slideshow, která je vygenerována jako samostatný soubor, o kterém jsem se zmiňoval již v základní konfiguraci. Pokud slideshow k ničemu nepotřebujete, můžete tuto záložku konfigurace přeskočit.

Výsledná galerie založená na šabloně „*Lytebox*“ obsahuje při prohlížení základní navigační prvky (předchozí a následující obrázek) umístěné přímo pod obrázkem. Zobrazování obrázků nemá žádné výrazné efekty, což alespoň napomůže k rychlejšímu prohlížení. V případě, že zobrazíte obrázek, automaticky se začne načítat obrázek v pořadí následující a předchozí, aby prohlížení bylo rychlejší. Nebylo by špatné zakomponovat do programu nabídku, aby si uživatel, který vytváří galerii, zvolil, zda chce načítat nějaké obrázky předem, nebo až při požadavku na zobrazení.



Obrázek 2 – Náhled uživatelského prostředí programu STGThumb

### Klady:

- Intuitivní a jednoduché provedení základního nastavení.
- Kvalitní výsledná galerie.

### Zápory:

- Volba přepsání existujících souborů při generování galerie je až v rozšířené nabídce konfigurace. Dotaz by mohl být zobrazen v okamžiku, kdy se shodují názvy.
- Nefunkční vytváření podsložek, nutno provést ručně.

Stránky projektu naleznete na této adrese <http://www.stgsys.com/stgthumb.asp>.

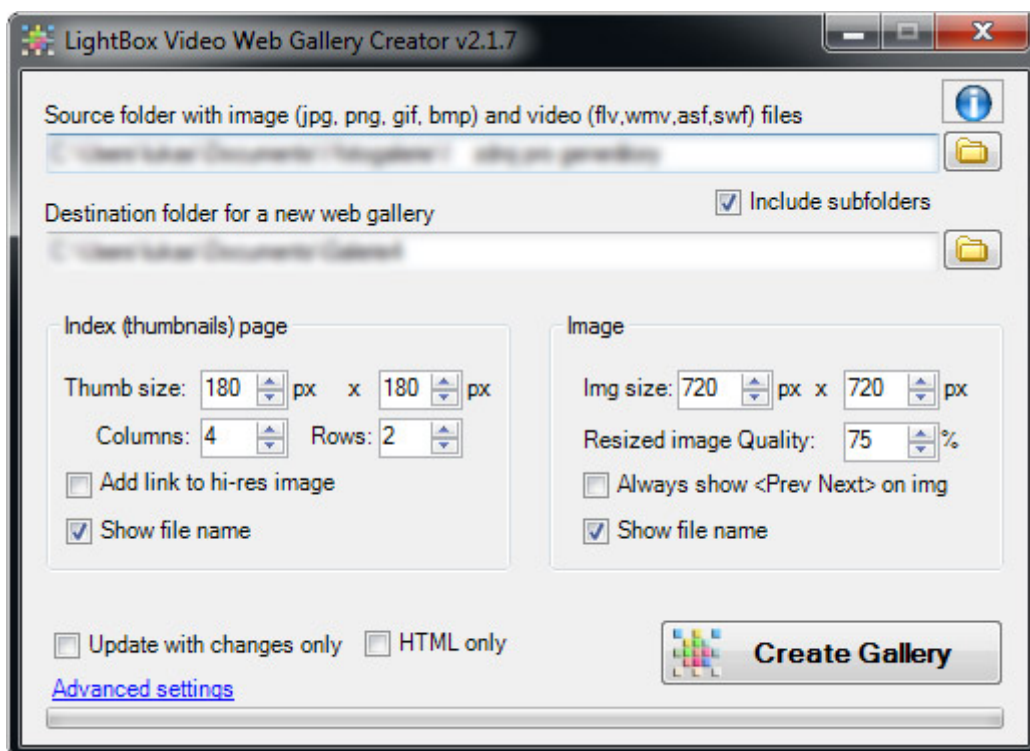
### 5.2.2 LightBox Video Web Gallery Creator

Další softwarový generátor, který generuje galerie založené na „Lytebox“. Výsledná galerie je oproti předchozímu popsanému generátoru téměř shodná. Liší se akorát umístěním ovládacích prvků (předchozí, následující), kde u této galerie jsou prvky umístěny přímo v zobrazovaném obrázku a zobrazují se pouze při přiblížení ukazatele myši poblíž ovládacího prvku. Podporované obrázkové formáty jsou JPG, PNG, GIF, BMP. Také podporuje některé video-formáty (FLV, WMV, ASF, SWF).

Generátor se může používat zcela zdarma pod podmínkou, že zůstane zachován odkaz pod vygenerovanou galerií, který odkazuje na stránku vývojářů. Nebo můžete zaplatit licenci, která umožňuje odstranění tohoto odkazu. Cena licence je 39 dolarů a galerii smíte využívat na neomezeném počtu webů.

Při spuštění programu máte zobrazenou pouze základní nabídku, která bohatě postačuje k vytvoření pěkné galerie. První pole, které musíme vyplnit, je adresář s obrázky, které chceme prezentovat. Zde hned narazíte na první omezení tohoto programu, a sice, musíte vybrat pouze celý adresář a nemůžete si zvolit jednotlivé obrázky do výsledné galerie. Není to zásadní nevýhoda, ale nutí to uživatele vytvářet adresář s obrázky v jiném programu (například Průzkumník systému Windows). Naopak, výhodou je možnost vytvořit strukturu adresářů, která bude zachována ve výsledné galerii, pokud zaškrtnete možnost „*Include subfolders*“. Dále volíte cílový adresář na disku, kam bude galerie uložena. Poté už zbývá jen nastavit základní parametry miniatur a velkých obrázků a můžete generovat galerii. Jediná věc, která je zde jiná, je ta, že pod každou miniaturu můžete vložit odkaz na obrázek v originální velikosti a to tak, že zaškrtnete „*Add link to hi-res image*“. Musíte, ovšem, brát na vědomí, že se velikost výsledné galerie několikanásobně zvýší, protože se na server musí nahrát i obrázky v plném rozlišení.

Pokud toto základní nastavení z nějakého důvodu nedostačuje, program nabízí i rozšířené možnosti konfigurace galerie. Tato rozšířená nabídka se zobrazí kliknutím na nápis „*Advanced settings*“. Na první záložce „*CSS*“ máte možnost provést ruční zásah do CSS stylu galerie (pouze pro uživatele, kterým není jazyk CSS cizí). Na další záložce „*Link Home*“ si můžete nastavit text, který se zobrazí nad miniaturami a bude odkazovat na domovský adresář webu. Záložka „*Combine*“ slouží pro vložení galerie do existující webové stránky, což je velmi zajímavý nástroj, který dokáže ulehčit práci s ručním vkládáním do hotových webových stránek. Na záložce s názvem „*Strip*“ máte možnost zvolit vygenerování jen galerie, která je vhodná pro vložení do existující stránky, protože výsledný HTML soubor neobsahuje kompletní HTML stránku, ale jen prvky důležité pro galerii. Kód z tohoto souboru jednoduše zkopírujete do existující stránky a galerie je hotová (samozřejmě musí být importován Javascript soubor, atd). Výsledná galerie z tohoto generátoru umožňuje automatické prohlížení obrázků v nastaveném intervalu. Tento interval si nastavíte právě na následující záložce s názvem „*Slide Show*“. Poslední záložka slouží na obnovení výchozích hodnot programu.



Obrázek 3 – Náhled uživatelského prostředí programu LightBox Video Web Gallery Creator

#### Klady:

- Velmi jednoduché základní ovládání pro laiky a pro pokročilejší doplňující funkce.
- Velmi dobrá výsledná galerie s funkcí automatického prohlížení.

#### Zápory:

- Nelze vybrat jednotlivé obrázky, pouze celý adresář.

Stránky projektu naleznete na této adrese <http://pranas.net/webgallerycreator/>.

### 5.2.3 VisualLightBox

Další vydařený projekt, který prozkoumáme detailněji, nese název „VisualLightBox“. Aktuální<sup>7</sup> vydání je ve verzi 4.8. Tento generátor se vyznačuje velkým množstvím různých šablon pro miniatury i pro velké obrázky, které mezi sebou můžete kombinovat. Použití pro nekomerční účely je zcela zdarma. V případě komerčního využití zaplatíte za licenci pro jeden web 49 dolarů a za neomezené množství webů 69 dolarů. Podporované formáty obrázků jsou BMP, DIB, RLE, JPG, JPEG, JPE, JFIF, GIF, EMF, WMF, TIF, TIFF, PNG a ICO. Generátor je dostupný v několika jazykových mutacích, které můžete za běhu programu přepínat volbou „Možnosti“ a poté „Jazyky“, kde si vyberete vámi preferovanou verzi. V české verzi je mnoho výrazů nepřeložených.

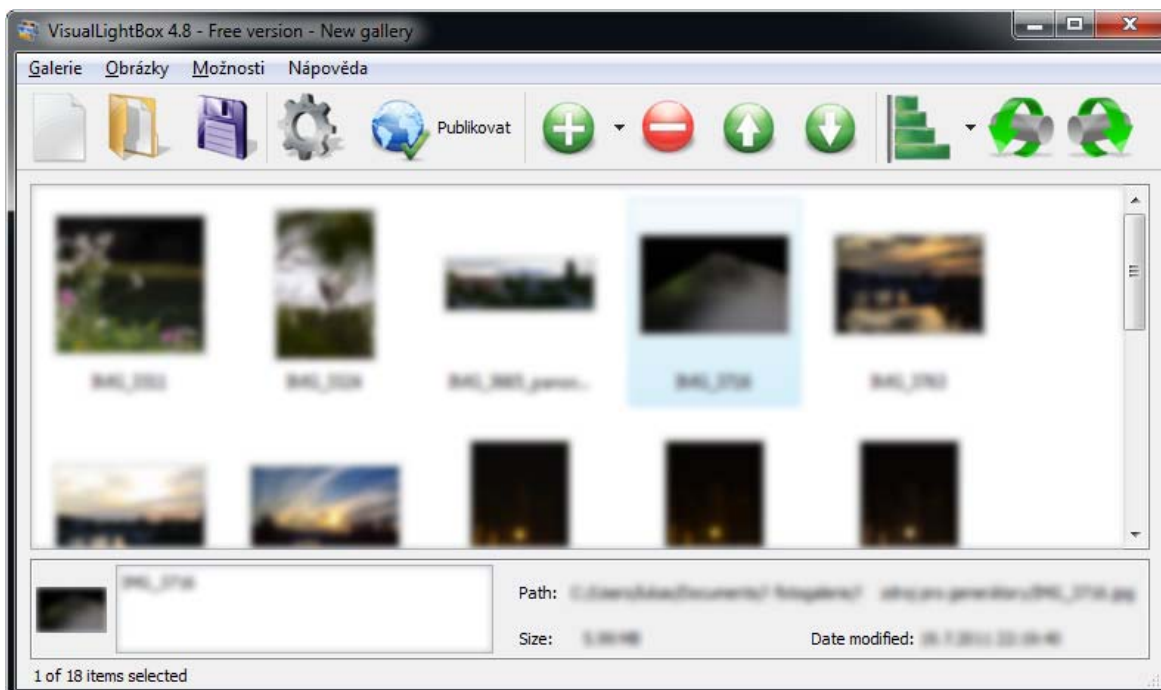
Po spuštění generátoru vidíte přes celé okno programu nápis, který nabádá k vložení obrázků. Kromě klasického tlačítka „Přidat“, tento generátor umí vkládat

<sup>7</sup> Psáno dne 1. 8. 2011

obrázky i tím způsobem, že v Průzkumníku Windows chytnete obrázky a přesunete je do okna s generátorem. Po vložení obrázků můžete změnit jejich pořadí přesouváním z jednoho místa na místo jiné. Také můžete obrázek přetočit v krocích po 90 stupních vpravo či vlevo.

Nyní zbývá provést konfiguraci výsledné galerie. Ta se aktivuje kliknutím na obrázek ozubeného kola. Otevře se okno se čtyřmi záložkami. V první záložce, „*Obecné*“, se skrývá základní nastavení galerie. Nejprve se nastavuje název galerie, poté jsou zde zaškrťovací tlačítka, která nastavují například povolení slideshow, povolení „*zoom*“ efektu při zobrazování obrázku, zda chcete zobrazit pozadí přes celou stránku v případě, že je zobrazený velký obrázek a také rychlost slideshow (pokud je povolena). Na této záložce se také nastavuje, jaký Javascript Framework bude výsledná galerie používat. Na výběr je jQuery a Prototype. Další záložka „*Miniatury*“ nastavuje, jak lze z názvu odvodit, parametry pro miniatury. Je zde na výběr několik vzhledových šablon pro stránku s miniaturami, dále rozměry miniatur, kvalita, formát a také počet sloupců na stránce. Nelze zvolit počet řádků na stránku, takže je patrné, že generátor umístí všechny obrázky na jednu stránku. Poslední konfigurační záložka „*Templáty*“ by mohla mít lepší překlad, ale to neubírá na možnostech programu. Opět se zde vybírá šablona, tentokrát pro zobrazení velkého obrázku, dále kvalita obrázku, rozměry a případně vodoznak. Poslední záložka, „*Publikovat*“ již není konfigurační, ale nastavuje výstup galerie. Je zde na výběr několik možností. První možností je uložení galerie do zvoleného adresáře na disk. Dále možnost přímého nahrání na FTP. Třetí možnost je vložení do existující stránky. A poslední možnost je velmi specifická, jedná se o vytvoření modulu pro redakční systém Joomla. Po zvolení jedné z těchto možností zbývá už jen kliknout na tlačítko „*Publikovat*“. V případě volby uložení galerie na pevný disk se generátor neptá, zda chcete přepsat existující adresář. Pokud již existuje, přepíše jej.

Výsledná galerie pracuje svižně. Při prohlížení načítá vždy předem následující a předchozí prvek pro urychlení načítání. Opět chybí v možnostech konfigurace, zda se bude načítat předem nějaký obrázek, nebo ne.



Obrázek 4 – Náhled uživatelského prostředí programu VisualLightBox

#### Klady:

- Jednoduché ovládání generátoru.
- Intuitivní ovládání galerie.
- Možnost automatické slideshow.
- Mnoho šablon pro vzhled miniatur a prohlíženého obrázku.

#### Zápory:

- Výslednou galerii nelze automaticky stránkovat.
- Při nastavení omezeného počtu sloupců není galerie vycentrována.
- Automaticky přepíše existující adresář bez jakéhokoli dotazu.

Stránky projektu naleznete na této adrese <http://visuallightbox.com/>

#### 5.2.4 jAlbum

Jeden z nejvydařenějších projektů generátoru webových galerií. Má rozsáhlé možnosti konfigurace. Dokonce umožňuje uživatelům přiložit ruku k dílu tím způsobem, že má každý možnost vytvořit si svou vlastní šablonu pro generování galerie. Projekt jAlbum poskytuje programovací rozhraní (API – Application programming interface), proti kterému se šablony programují. Výsledné galerie některých šablon jsou založeny pouze na HTML, ty pokročilejší využívají Javascript. V případě, že vytvoříte vlastní šablonu, můžete ji publikovat na webu projektu<sup>8</sup>, kde ji zároveň uživatelé, kteří vaši šablonu použijí, mohou ohodnotit. Díky tomu mají ostatní přehled, které šablony jsou vydařené a které nemá smysl stahovat. Některé šablony jsou šířeny zdarma a jiné podléhají

<sup>8</sup> <http://jalbum.net/cs/skins>

licenci, kterou musíte, v případě užití skinu, zaplatit. Program samotný je volně šiřitelný bez jakéhokoli poplatku.

Autoři projektu jAlbum se neomezili pouze na generování obrázkové galerie, ale do svého programu zakomponovali i základní grafické operace s obrázky. Je možné provádět operace jako je ořez obrázku, korekce červených očí, použít fotografický filtr, vložit text, a mnoho dalších úprav. Tyto operace s obrázky jsou opět vytvářeny formou plug-inů, na které existuje programovací rozhraní a uživatelé mají možnost i v tomto případě ovlivnit vývoj programu.

Po vložení obrázků pro generování je uživatel k zadání názvu a popisu alba, které se následně vytvoří. Jednotlivé obrázky mohou být upraveny tím, že na daný obrázek najedete a kliknete na nápis „*Editace*“. V pravé části programu se zobrazí nabídka, která ukazuje možnosti, které pro editaci máte. Nemusíte mít starost o své originální obrázky, ty se neupravují. Na matici s obrázky se vrátíte tak, že najedete na náhled obrázku, kde se zobrazí tlačítko s maticí, na které kliknete. Dále je vhodné zvolit si šablonu galerie, která bude použita pro generování. Šablona se vybírá v levé spodní části programu „*Vzhled a styl*“. Na stránkách projektu jAlbum lze stáhnout další šablony programované uživateli. Hned pod výběrem šablony je možnost volby barevného schématu dané šablony (pokud šablona obsahuje více schémat).

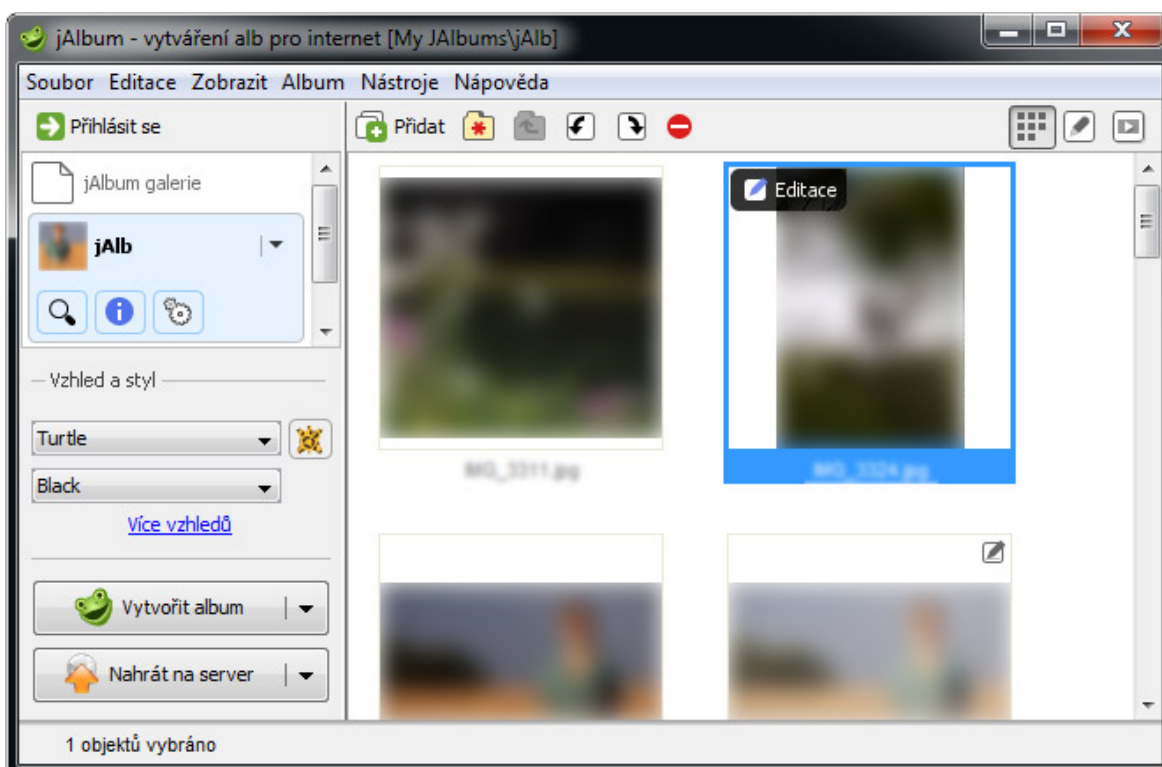
Důležitá věc je také nastavení výsledné galerie. To se provádí kliknutím v horní liště na nabídku „*Album*“ a poté „*Nastavení alba*“, nebo můžete stisknout klávesu F4, která také aktivuje nastavení.

První záložka „*Základní volby*“ umožňuje zvolit obrázek, který bude použit jako náhled alba, dále pak název a popis alba a nakonec upravit cílové adresáře pro uložení do počítače. Další záložka, „*Stránky*“, umožňuje zvolit různé propojení obrázků. Pracovat můžete se zmenšenými obrázky nebo s originály, které jsou datově mnohem objemnější. Na této záložce se nastavuje počet sloupců a řádků galerie, to znamená, že se výsledná galerie bude stránkovat v případě, že se nevejde na jednu stránku. Dále se zde nastavuje řazení obrázků a v pokročilých volbách je možnost „*Kopírovat původní obrázky*“, což by mělo originální soubory, ze kterých je vytvářena galerie, zkopírovat do cílového adresáře galerie, ale při psaní této práce to nechtělo fungovat. Třetí záložka, „*Obrázky*“, obsahuje položky pro konfiguraci kvality a rozměrů výsledných obrázků. V pokročilých volbách je, mimo jiné, možnost zvolit grafickou akceleraci pro zpracování obrázků, ale podržením kurzoru myši na této volbě se zobrazí nápis, který varuje před možnou zhoršenou kvalitou obrázků na některých grafických kartách. V záložce číslo čtyři, „*Pokročilé volby*“, se nastavuje například kódování, zachování meta informací u obrázků, pojmenování index souboru a mnoho dalších. Poslední záložka je záložka konfigurace dané šablony, takže je s každou šablonou jiná. Obsahuje dostupné možnosti konfigurace výsledné galerie, jako je například prodleva mezi obrázky při automatickém prohlížení, barva pozadí při zobrazeném obrázku a další konfigurace tohoto typu. Některé šablony obsahují opravdu



velké množství konfiguračních položek, jiné zase obsahují třeba jen dvě a není možnost do výsledné galerie moc zasáhnout.

Po provedení konfigurace už zbývá poslední věc, kterou je vygenerování galerie. Tlačítko pro spuštění generování je umístěné v levé dolní části programu „Vytvořit album“. Také můžete výslednou galerii přímo nahrát na FTP server pomocí tlačítka „Nahrát na server“.



Obrázek 5 – Náhled uživatelského prostředí programu jAlbum

#### Klady:

- Možnost vytvoření vlastních šablon a plug-inů
- Program je zcela zdarma
- Možnost editace obrázků

## 6 Programovací jazyky vhodné pro vývoj softwarových generátorů

Existuje velké množství programovacích jazyků, ale ne každý se dá snadno použít pro vývoj aplikace s grafickým rozhraním. Největší využití v oblasti grafických programů mají jazyky objektově orientované. Protipól objektově orientovaným jazykům jsou jazyky založené na procedurálním programování.

### 6.1 Vlastnosti objektově orientovaných jazyků

Objektově orientované programování se snaží prvky z reálného světa převést do počítače, respektive do programu, kde s těmito prvky chceme pracovat. Těmto prvkům se říká **objekty**. Objekty obsahují nějaké atributy, které odrážejí jejich aktuální stav. Přístup k těmto atributům z vnějšku objektu je realizován pomocí veřejných přístupových metod (tzv. „*setter*“ a „*getter*“). Objekt je instance třídy. Třída je datová struktura popisující chování objektů.

Programátor, který chce s objektem pracovat, nepotřebuje vědět, jak se konkrétní věc provádí, ale stačí mu pouze výsledek na základě vstupních hodnot. Tomuto jevu se říká **abstrakce** (abstrahuje od detailů implementace jednotlivých objektů).

Další vlastnost objektově orientovaných jazyků je **zapouzdření**, což zaručuje, že objekt nemůže přímo přistupovat k atributům jiných objektů, ale musí měnit hodnoty těchto atributů pomocí, již zmíněných, přístupových metod („*setter*“ a „*getter*“). Tím zamezíme nežádoucí změně hodnot atributů.

**Skládání** je další vlastnost charakterizující objektově orientované jazyky. Znamená to, že můžeme jako atribut objektu zvolit jiný objekt.

Mezi další vlastnosti objektově orientovaných jazyků patří **dědičnost**. Dědičnost umožňuje vytváření objektů, které jsou potomky jiných objektů. Tím, vytvořený objekt, získá všechny vlastnosti svého rodičovského objektu a může k nim přidat vlastnosti další. Toho se využívá zejména, chceme-li použít třídu napsanou někým jiným a určitým způsobem změnit její chování.

Poslední vlastností objektového programování je **polymorfismus**. Objekty, které jsou instancemi různých tříd, poskytují svému okolí určitou množinu stejných operací (metod), ale každý objekt tyto vlastnosti metody provádí jiným způsobem. Všechny tyto objekty mají společného rodiče a v programu s nimi pracujeme jako s instancemi právě tohoto rodiče. Příklad polymorfismu je, že chceme v programu počítat obvod trojúhelníků a čtverců, takže si vytvoříme rodičovskou třídu, která bude společná všem prvkům, s názvem „*Geometricky\_utvar*“. V této třídě bude metoda pro výpočet obvodu. Dále si vytvoříme třídy „*Ctverec*“ a „*Trojuhelnik*“, které budou potomky třídy „*Geometricky\_utvar*“, což nás nutí do těchto potomků implementovat metodu pro výpočet obvodu. To znamená, že jinak bude pracovat metoda pro výpočet obvodu čtverce a jinak

trojúhelníku. V programu vytváříme instance tříd „*Ctverec*“ a „*Trojuhelnik*“, ale pracujeme s nimi jako s třídou „*Geometricky\_utvar*“. Tento způsob programování umožňuje pozdější přidání dalšího geometrického útvaru, například kruhu, do programu bez jakékoli další úpravy programu.

## 6.2 C++

Jazyk C++ je jeden z jazyků, které jsou vhodné pro vytváření aplikací s grafickým uživatelským prostředím. Tento jazyk se začal vyvíjet v roce 1980 z jazyka C, který rozšiřuje o vlastnosti objektově orientovaného programování. Z tohoto vyplývá, že veškeré programy fungující v jazyku C, fungují i v C++, až na jasně vymezené výjimky.

Název C++ má značit vylepšení původního jazyka C, protože v syntaxi těchto jazyků znamenají dvě znaménka „+“ za sebou zvýšení celočíselného datového typu o hodnotu jedna.

Jazyk C++ je velmi složitý a pro začátečníky je mnohem náročnější, než například Java. Může za to spousta faktorů. Mezi ně patří ukazatelé, neboli pointery, ve kterých udělá každý začínající programátor spousty chyb a nejednoho programátora to otráví do takové míry, že přestane jazyk C++ používat. Další záležitostí tohoto jazyka je nutnost ruční správy paměti. Každý nově vytvořený objekt musíte smazat, pokud ho nadále nepotřebujete ve svém programu. V případě, že zapomenete na smazání nějakého objektu, může se stát, že po chvíli používání programu zabere veškerou dostupnou operační paměť počítače. Na druhou stranu, tyto faktory, které činí jazyk C++ tak náročným, umožňují vytvářet velmi rychlé programy s menšími nároky na operační paměť. Ale důležité je správně pracovat s pamětí.

Použité zdroje: (Pipošiar, 2003), (Wikipedie, 2004)

## 6.3 Java

Java je další jazyk vhodný pro vývoj software s grafickým prostředím. Tento jazyk byl poprvé představen v roce 1995 firmou Sun Microsystems a později prodána firmě Oracle. Od roku 2007 je Java vyvíjena jako open source (zdrojové kódy jsou volně dostupné na Internetu).

Tento jazyk má velmi podobnou syntaxi s jazyky C a C++. Nicméně Java je o poznání jednodušší pro začátečníky a to hlavně z toho důvodu, že nemusíte ručně mazat nepotřebné objekty, protože se o to stará tzv. Garbage collector, který automaticky objekt smaže a tím uvolní paměť dalším objektům. Dále odpadá práce s ukazateli, protože tu žádné nejsou.

Obrovskou výhodou Javy je přenositelnost. Jednou napsaný program můžete využívat na různých architekturách strojů (mobilní telefony, čipové karty, osobní počítače, servery, atd.) i na různých operačních systémech (Windows, Linux, atd.). Při kompilaci programu se totiž nevytváří přímo strojový kód, ale tzv. bajtkód, který je nezávislý na

architektuře stroje i na operačním systému. Jedinou věc, kterou potřebujete na stroji, kde bude využíván program napsaný v Javě, je interpret Javy, neboli Java Virtual Machine (JVM). JVM se stará o převod bajtkódu na strojové instrukce daného hardware.

Použité zdroje (Studenti Západočeské univerzity) a (Wikipedie, 2004).

## 7 kGenerator

kGenerator, takto byl nazván program, který je předmětem praktické části této bakalářské práce. Další praktická část bylo vytvoření javascriptové galerie, na které bude založena výsledná galerie vygenerovaná kGenerátorem. Tato aplikace dostala název kGallery.

### 7.1 kGallery

kGallery je javascriptová aplikace pro prohlížení obrázků na webových stránkách. Je napsaná čistě javascriptovým kódem bez jakýchkoli externích knihoven, či frameworků. Toto řešení bylo zvoleno pro úplnou kontrolu nad aplikací a také proto, že použití frameworku s sebou přináší nutnost importovat soubor s tímto frameworkem do webové stránky, na které bude umístěna aplikace, tím pádem více dat pro přenos mezi serverem a uživatelem. Výsledný soubor zdrojového kódu Javascriptu má, po použití komprimovacího nástroje <http://jscompress.com/>, 40 kB. Rozhodně by šlo napsat mnohem kvalitnější aplikaci v mnohem menším objemu dat. Nicméně já, jakožto autor této aplikace, jsem se s Javascriptem v praxi setkal poprvé při programování této bakalářské práce, takže se nedá očekávat plně optimální kód.

Aplikace poskytuje uživatelské rozhraní pro konfiguraci galerie. Možnosti konfigurace nejsou až tak rozsáhlé, ale základní konfiguraci poskytuje. Kompletní popis uživatelsky konfigurovatelných položek naleznete v příloze C.

#### 7.1.1 Ruční použití javascriptové aplikace

Pro ruční vkládání aplikace do webové stránky je nutné se držet základních podmínek pro správný chod galerie. V závorce je vždy uveden název třídy elementu (class), který je použit v souboru s CSS styly:

- Galerie musí být vložena do elementu „*div*“, nebo „*table*“ (název třídy je kGallery) a tento element musí mít přidělený identifikátor v podobě atributu „*id*“. kGenerator je optimalizován pro generování galerie v elementu „*div*“, takže element „*table*“ nebudeme dále rozebírat. Stejně tak kaskádový styl je optimalizován pro element „*div*“.
- Dále je volitelné vložit na začátek element nějakého z nadpisů „*h1*“ – „*h6*“ (kGallery-header).
- Pro vložení popisu galerie použijte element „*p*“ (kGallery-descr).
- Poté je nutné vložit element, který slouží jako obal pro obrázky. Můžete zvolit „*span*“, nebo „*div*“ (kGallery-images).
- Následně vkládáte jednotlivé obrázky, které chcete prezentovat na webu.
  - o Každý obrázek je vložen do elementu „*span*“ (kGallery-image), který obsahuje element „*a*“ (kGallery-image-url) odkazující na velký náhled obrázku. Do tohoto elementu „*a*“ je vložen element obrázku „*img*“ (kGallery-image-imelem) s atributem „*src*“ odkazujícím na miniaturu.

- Poté můžete do elementu s obrázkem vložit ještě název obrázku v podobě elementu „*span*“ (kGallery-image-name) a také popis obrázku, opět „*span*“ (kGallery-image-descr). Tento název ani popis se na webové stránce nezobrazují, název obrázku se zobrazí ve chvíli, kdy je zobrazen jeho velký náhled. Zobrazení popisu nebylo, zatím, implementováno.
- Po vložení všech obrázků už zbývá poslední věc, kterou je konfigurace galerie. Ta se provede v elementu „*script*“ s atributem „*type*“ nastaveným na hodnotu „*text/javascript*“. V tomto elementu je nutno vytvořit instanci třídy, která se stará o běh galerie, poté nastavit požadované hodnoty pomocí přístupových metod a jako poslední věc je spuštění galerie. Příklad konfigurace galerie ukazuje následující část zdrojového kódu.

```
<script type="text/javascript">
  // "kGallery" jako parametr konstruktoru je identifikátor
  elementu, ve kterém je celá galerie umístěna na webu
  var instKGallery = new kGallery("kGallery");
  instKGallery.setPreloadovatPrvkyDolniListyNavigace(true);
  instKGallery.setIntenzitaPozadi(90);
  instKGallery.setFpsZobrazovaniPozadi(100);
  instKGallery.setDobaZobrazovaniPozadi(1000);
  instKGallery.setBarvaPozadi("#000000");
  // Finální spuštění galerie. Před provedením následujícího příkazu
  nebude galerie fungovat
  instKGallery.startKGallery();
</script>
```

Konfigurace by měla být vkládána až za element, který uvozuje celou galerii („*div*“, nebo „*table*“).

### 7.1.2 Hash v URL adrese

Hash v URL adrese znamená znak dvojitého křížku (#). Téměř všechny webové galerie založené na Javascriptu tento hash nepoužívají. To má za následek, že když nějaký návštěvník chce ukázat obrázek svému kamarádovi tím způsobem, že mu pošle odkaz, který je v adresním řádku prohlížeče, tak se mu zobrazí pouze celá galerie a ne daný obrázek. To je z toho důvodu, že se při prohlížení obrázků adresa nemění, takže není daný obrázek jednoznačně identifikován URL adresou. URL adresu nelze jednoduše změnit bez kompletního obnovení stránky, podporují to pouze některé prohlížeče. Jediná část, kterou lze v adrese změnit bez obnovení stránky je právě část za znakem „#“. Část za „*hashem*“ má využití v případě, že chceme odkázat na určitý obsah stránky. To znamená, že se nám, při načtení, stránka automaticky posune na potřebné místo. kGallery ovšem této části adresy využívá tak, že jednoznačně identifikuje obrázek, který následně zobrazí.

Celé to funguje následovně. Uživatel přijde na web, kde je implementována galerie kGallery, a zobrazí si velký náhled nějakého obrázku. V tu chvíli se do adresního řádku k adrese přidá znak „#“ a za to text, který jednoznačně identifikuje daný obrázek. Při každém zobrazení jiného obrázku se tato část za znakem „#“ změní, vždy identifikuje aktuální obrázek. Když se uživatel rozhodne, že tento obrázek ukáže svému kamarádovi, zkopíruje odkaz a pošle jej (například prostřednictvím e-mailu) svému kamarádovi. Ten klikne na odkaz, který mu přišel a načte se celá stránka včetně galerie. Po dokončení

načítání, respektive po zapnutí galerie proběhne kontrola adresy, zda neobsahuje nějaký text za znakem „#“. V případě, že nějaký text obsahuje, provede se další kontrola, zda obsahuje text, který identifikuje nějaký z obrázků, které jsou dostupné na dané stránce. Pokud ne, nic se nezobrazí. Pokud obrázek existuje, tak je uživateli zobrazen.

Tato funkce je volitelná a případný webmaster má možnost hash funkci vypnout (ve výchozím stavu je zapnuta).

### 7.1.3 Další užitečné funkce

Při zobrazeném velkém náhledu se obrázek automaticky přizpůsobuje velikosti okna prohlížeče, aby uživatel vždy viděl celý obrázek. Pro zobrazení obrázku v originálním rozlišení náhledu stačí kliknout na daný náhled. Toto zobrazení už ovšem neprobíhá v rámci Javascriptu, ale je obrázek zobrazen v prohlížeči jako klasický „img“ element. S otevřeným náhledem je také zakázáno veškeré posouvání stránky vertikálně i horizontálně, aby nedocházelo k nechtěnému posunu (v internetovém prohlížeči Opera toto nefunguje na sto procent).

Navigační prvky pro prohlížení obrázků (první, předchozí, následující a poslední) jsou umístěny vždy na spodní části obrazovky a můžete pomocí nich přepínat mezi obrázky. Pro prohlížení obrázků můžete také využít šipek (vlevo a vpravo) na klávesnici. Prohlížení obrázků můžete provést kliknutím mimo obrázek, nebo kliknutím na nápis „Zavřít“ a poslední možnost je stisknutí klávesy „Esc“.

### 7.1.4 Shrnutí

Galerie nabízí základní funkce pro prohlížení obrázků, poskytuje komfort v podobě možnosti prohlížet obrázky pomocí šipek na klávesnici. Prohlížení obrázků je velice rychlé.

Na druhou stranu, galerie nedisponuje žádnými zajímavými efekty, kromě postupného zobrazování pozadí.

## 7.2 kGenerator – generátor galerie kGallery

kGenerator je software určený pro generování obrázkových galerií založených na technologiích XHTML, CSS a Javascript. Tento software byl vyvinut primárně jako bakalářská práce a byl napsán kompletně v programovacím jazyku Java. K vývoji byly použity tyto externí knihovny:

- JDOM – knihovna pro práci s XML dokumenty (<http://www.jdom.org/>),
- HtmlCleaner – knihovna pro práci s HTML dokumenty (<http://htmlcleaner.sourceforge.net/>),
- SwingX – knihovna použitá pro detekci operačního systému (<http://swinglabs.org/>).

Spouštěcí soubor, pro platformu Windows, byl vygenerován pomocí programu Launch4j (<http://launch4j.sourceforge.net/>).

### 7.2.1 Práce s programem

Program se žádným způsobem neinstaluje, pouze se spustí EXE soubor. Poté se zobrazí okno programu a je připraven ke generování galerie.

Na úvodní obrazovce programu jsou vidět základní tlačítka pro přidání a odebrání obrázků. Levé části programu dominuje seznam obrázků, které budou použity pro generování galerie. V pravé části je okno pro případné zobrazení náhledu obrázku, který je označen v seznamu obrázků. Dole pod obrázkem jsou zobrazeny informace o aktuálně zobrazeném obrázku. Program si dokáže poradit pouze s obrázky do velikosti 25 megapixelů. Vložíte-li obrázek větší, tak při zpracování obrázku program selže tak, že se přeruší vlákno pro zpracování obrázků (pozn.: nepodařilo se mi to žádným způsobem vyřešit).

Horní nabídka má tři základní položky. Položka „*Soubor*“ skrývá možnosti pro otevření nového projektu, nebo otevření uloženého projektu, dále pak uložení projektu a poslední možnost je vypnutí kGeneratoru. Další záložka „*Jazyk*“ slouží pro volbu jazykové verze programu. O této možnosti jsou informace v dalším odstavci. Poslední záložka je „*Nápověda*“, která skrývá pouze nepodstatné „*About*“, což zobrazí informace o programu.

Položka jazyk zatím obsahuje pouze dva jazyky a anglický jazyk je přeložen je z části. Nicméně jazykové verze se dají přidávat i bez kompilace programu. Tyto jazykové verze jsou uloženy v XML souboru a jakýkoli uživatel má možnost si překlad upravit, případně přidat. XML dokument má relativně intuitivní strukturu, takže se v něm vyzná i člověk, který alespoň tuší, co je XML. Tento soubor s překlady je umístěn v adresáři „*lang*“, který je ve stejném adresáři jako spouštěcí soubor kGeneratoru. Načítání jazyků probíhá tak, že při spuštění programu je načten XML soubor, kde je přečteno, jaké jazyky jsou v překladu obsaženy a poté jsou na základě toho vytvořeny tlačítka, pomocí kterých si může uživatel jazykovou mutaci změnit za běhu programu. Bohužel, program si nepamatuje volbu jazyku a po opětovném zapnutí kGeneratoru musíte verzi překladu volit znovu.

Pod horní nabídkou jsou zobrazeny 4 ikonky, které z části duplikují horní nabídku. Navíc zde přibyla ikonka Facebooku, která odkazuje na „*fanpage*“ kGeneratoru.

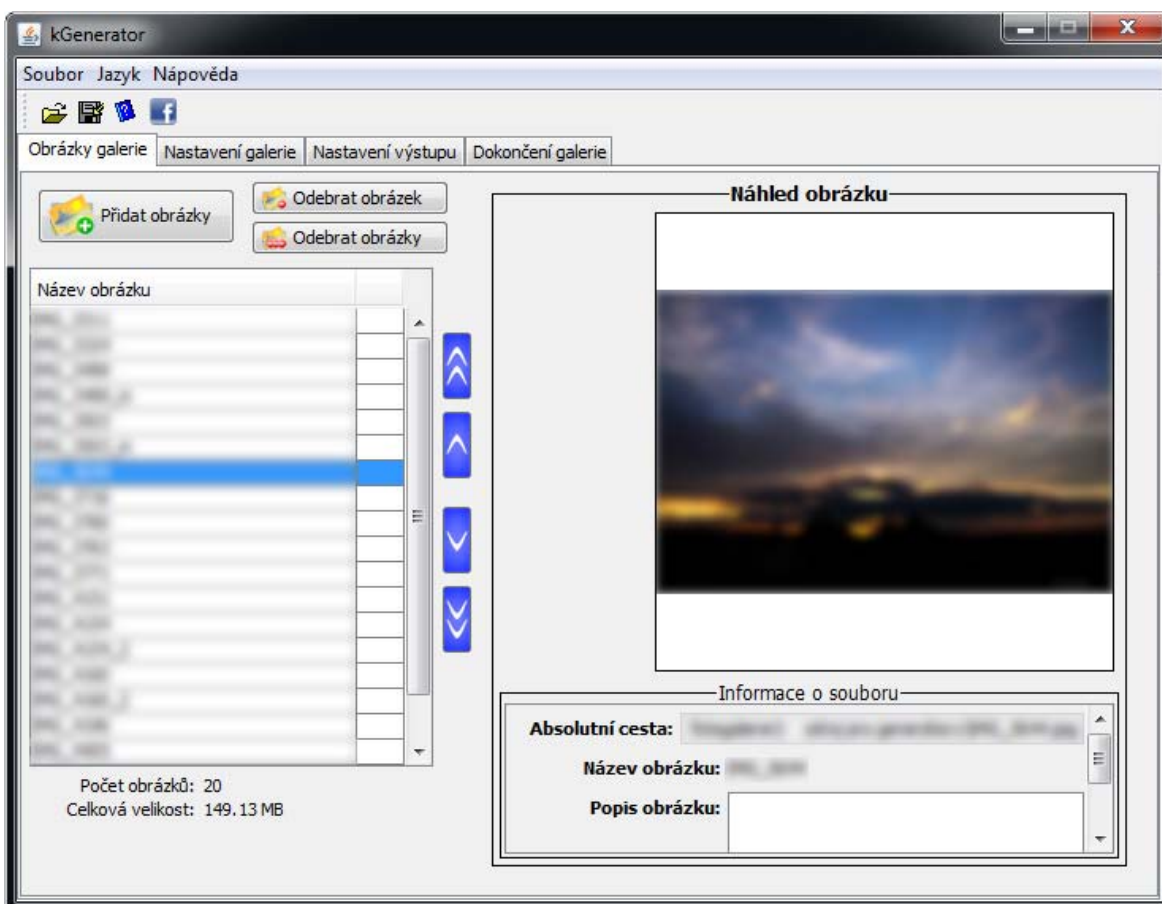
Po přepnutí na další záložku „*Nastavení galerie*“ vidíte nabídku, která je opět rozdělena na záložky. První záložkou je „*Nastavení miniatur*“, která obrazuje možnosti nastavení miniatur a vzhledu HTML stránky. Další záložka, „*Nastavení zobrazení obrázku*“, nastavuje kvalitu obrázků a vzhled galerie při zobrazení velkého náhledu. Třetí záložka nastavuje různé texty, které se v galerii zobrazují. Přepnutím na třetí záložku hlavního panelu, s názvem „*Nastavení výstupu*“, se zobrazí volby pro nastavení cílového adresáře, podadresářů, atd. poslední záložkou je „*Dokončení galerie*“, kde vidíte sumární informace výsledné galerie. Stisknutím na tlačítko „*Generuj!*“ se spustí generování galerie dle zadaných konfiguračních parametrů na předchozích záložkách.



## 7.2.2 Shrnutí

Oproti profesionálním řešením je tento software „v plenkách“ a zasloužil by si mnohem více času věnovaného vývoji. I přesto je projekt ve fázi, kdy je uživatel schopen z vložených obrázků vygenerovat obrázkovou galerii, kterou lze relativně snadno vložit do již existujícího webu, nebo ponechat jako jednoúčelovou stránku pro prezentaci obrázků.

V porovnání s programy, které byly popsány výše, kGenerator nenabízí tak jednoduché a přímočaré nastavení, ale uživatel je víceméně nucen projít celou konfiguraci a nastavit každou položku. Zpracování obrázků je o poznání pomalejší, což je zapříčiněno ne zcela ideálním přístupem k souborům a následnému zpracování obrázků. Naopak, kGenerator, jako jediný z programů uvedených v této bakalářské práci, umožňuje uživatelské nastavení, zda se mají obrázky načítat inteligentně předem, nebo zda se mají načíst všechny okamžitě při zobrazení stránky a konečně, zda se nemají načítat žádné obrázky předem. Také podporuje práci se znakem „#“ v URL adrese, což umožňuje uživateli odeslat kamarádovi odkaz přímo na obrázek a ne pouze na stránku, kde je galerie umístěna.



Obrázek 6 – Náhled uživatelského prostředí programu kGenerator

### **7.3 Možná zlepšení do budoucna**

Na vývoj této javascriptové galerie a softwarového generátoru nebylo vynaloženo zdaleka takové množství času, které by bylo potřeba a nabízí se mnoho možností k vylepšení. Mezi ně patří například přímé nahrávání galerie na FTP server, vkládání galerie do existující webové stránky, ruční úprava CSS stylu, možnost vybrat si vlastní obrázky pro navigační prvky (první, předchozí, další, poslední), přidat možnost volby efektu pro přechod mezi obrázky (zároveň s tím samozřejmě tyto efekty realizovat v Javascriptu) a mnoho dalších úprav, které vylepší stávající řešení.

## 8 Porovnání výše zmíněných generátorů galerií

Generátor STGThumb má za sebou deset let vývoje a má velkou škálu podporovaných typů obrázků, společně s VisualLightBox. Každý z těchto programů podporuje minimálně dvanáct typů obrázků. Naproti tomu Web Gallery Creator a jAlbum si poradí pouze se základními typy jako je JPG, GIF a PNG. KGenerator má zaručenou podporu pouze JPG souborů. JAlbum, jako jediný ze zmiňovaných programů, umožňuje vkládat nové moduly a plug-iny, které si můžete naprogramovat samy pomocí dostupného programátorského rozhraní. Podporu automatického vkládání vodoznaku do obrázku mají generátory STGThumb, jAlbum a v placené verzi i VisualLightBox. Poslední zajímavý parametr je doba generování galerie. Všechny zmiňované generátory byly podrobeny testu, který spočíval v tom, za jak dlouho vygenerují galerii ze stejných obrázků se shodnou konfigurací. Konkrétně se jednalo o 20 obrázků s celkovou velikostí 149 MB, parametry vygenerovaných obrázků byly 800x800 pixelů s 95% kvalitou a miniatury 180x180 pixelů. Nejrychleji byla galerie vygenerována v generátoru jAlbum, a to za 20 sekund. Naopak nejpomaleji generuje kGenerator, kterému stejný proces trval 160 sekund. Kompletní výsledky testů jsou uvedeny v tabulce níže, včetně celkové velikosti výsledné galerie.

**Tabulka 1 – Porovnání dostupných generátorů**

	STGThumb	Web Gallery Creator	jAlbum	VisualLightBox	kGenerator
Cena (\$)	19,95	39	0	49 až 69	0
Modulárnost	Ne	Ne	Ano	Ne	Ne
Podporované formáty	JPG, JP2, J2K, J2C, TIF, PNG, BMP, PCX, WMF, EMF, ICO, CUR a PSD	JPG, PNG, GIF a BMP	JPEG, GIF a PNG	BMP, DIB, RLE, JPG, JPEG, JPE, JFIF, GIF, EMF, WMF, TIF, TIFF, PNG a ICO	JPG
Podpora vodoznaku	Ano	Ne	Ano	Ano, v placené verzi	Ne
Doba generování (s)	59	38	20	26	160
Výsledná velikost galerie (MB)	2,81	3,16	2,42	3,51	2,97

Softwarový generátor, jehož vývoj začal již v roce 2001 a poslední verze (3.10) byla vydána v červnu roku 2011<sup>9</sup>. Program můžete stáhnout z Internetu zdarma pro jeho vyzkoušení a poté zakoupit licenci za 19,95 dolarů. Bez zakoupení licence generátor vkládá do obrázků vodoznak s nápisem „*STGTHUMB*“, takže je výsledná galerie víceméně nepoužitelná.

Tento generátor je navržen pro jednoduché vytvoření obrázkové galerie. Můžete si zvolit ze 4 základních typů galerie, kde pouze typ „*Lytebox*“ funguje ve spojení s Javascriptem. Ostatní možnosti galerie využívají pouze HTML, případně ještě CSS, takže se jimi nebudeme zabývat a zaměříme se na typ „*Lytebox*“.

Pomocí tohoto generátoru můžete prezentovat obrázky mnoha formátů (JPG, JP2, J2K, J2C, TIF, PNG, BMP, PCX, WMF, EMF, ICO, CUR, PSD), které budou při generování galerie převedeny na JPG. Generátor umožňuje prezentaci videí ve formátu „*avi*“ a „*mpg*“, ale to pro naše potřeby není podstatné. Dále podporuje vkládání vlastních vodoznaků do obrázků, které prezentujete. Toho se využije v případě, že mají být obrázky chráněny proti odcizení. Například, jedná-li se o prezentaci obrázků k prodeji.

---

<sup>9</sup> Informace ze dne 31. 7. 2011

## 9 Závěr

Cílem bakalářské práce bylo čtenáře seznámit se základními technologiemi používanými pro tvorbu webových galerií. Dále pak obeznámení s možnými problémy implementace Javascriptu a CSS stylů v jednotlivých webových prohlížečích. Hlavním tématem byly softwarové generátory javascriptových galerií, které byly popsány čtyři plus generátor, který je předmětem praktické části této bakalářské práce. Nejlepší generátor z výše zmíněných je jAlbum, který nabízí zdaleka nejvíce možností a také zapojuje uživatele do svého projektu v podobě možnosti programování vlastních plug-inů a šablon.

## Literatura

**Dottoro.com. 2011.** Events in JavaScript. *Dottoro Web Reference*. [Online] 2011. [Citace: 2. Srpen 2011.] <http://help.dottoro.com/larrqgck.php>.

**Grimmich, Šimon. 2008.** CSS styly. *Tvorba webu*. [Online] 2008. [Citace: 30. Červenec 2011.] <http://www.tvorba-webu.cz/css/>.

—, 2008. HTML. *Tvorba webu*. [Online] 2008. [Citace: 28. Červenec 2011.] <http://www.tvorba-webu.cz/html/>.

—, 2008. JavaScript. *Tvorba webu*. [Online] 2008. [Citace: 31. Červenec 2011.] <http://www.tvorba-webu.cz/javascript/>.

—, 2008. JavaScript. *Tvorba webu*. [Online] 2008. [Citace: 31. Červenec 2011.] <http://www.tvorba-webu.cz/javascript/>.

—, 2008. XHTML: XHTML. *Tvorba webu*. [Online] 2008. [Citace: 28. Červenec 2011.] <http://www.tvorba-webu.cz/xhtml/>.

—, 2008. XML: XHTML. *Tvorba webu*. [Online] 2008. [Citace: 28. Červenec 2011.] <http://www.tvorba-webu.cz/xml/xhtml.php>.

**Hønsi, Torstein, a další. 2011.** Highslide JS. *Highslide JS*. [Online] Highsoft, 26. Leden 2011. [Citace: 27. Srpen 2011.] <http://www.highslide.com/>.

**Milde, Daniel. 2011.** jQuery - JavaScript framework - DunLog. *DunLog*. [Online] 2011. [Citace: 3. Srpen 2011.] <http://blog.milde.cz/post/146-jquery-javascript-framework/>.

**Pipošiar, Jiří. 2003.** *Programování v C/C++*. [Online] 2003. [Citace: 5. Srpen 2011.] <http://programovat.kvalitne.cz/>.

**Refsnes Data. 2011.** HTML DOM Event Object. *W3Schools Online Web Tutorials*. [Online] Refsnes Data, 2011. [Citace: 2. Srpen 2011.] [http://www.w3schools.com/jsref/dom\\_obj\\_event.asp](http://www.w3schools.com/jsref/dom_obj_event.asp).

**Studenti Západočeské univerzity.** Dioné: Programovací jazyk Java. *Dioné | Nezávislý studentský informační server Západočeské univerzity v Plzni*. [Online] [Citace: 5. Srpen 2011.] <http://v1.dione.zcu.cz/java/uvod.html>.

**TAUFER, Ivan, KOTYK, Josef a JAVŮREK, Milan. 2009.** *Jak psát a obhajovat závěrečnou práci bakalářskou, diplomovou, rigorózní, disertační, habilitační*. Pardubice : Univerzita Pardubice, 2009. ISBN 978-80-7395-157-3.

**THAU, Dave. 2009.** *Velký průvodce JavaScriptem*. Praha : Grada Publishing a.s., 2009. ISBN 978-80-247-2211-5.

**Wikipedie. 2004.** C++. *Wikipedie, otevřená encyklopedie*. [Online] Wikipedie, 20. Červen 2004. [Citace: 5. Srpen 2011.] <http://cs.wikipedia.org/wiki/C%2B%2B>.

— **2004.** Extensible HyperText Markup Language. *Wikipedie, otevřená encyklopedie*. [Online] 31. Říjen 2004. [Citace: 28. Červenec 2011.] <http://cs.wikipedia.org/wiki/XHTML>.

— **2004.** HyperText Markup Language. *Wikipedie, otevřená encyklopedie*. [Online] 16. Červenec 2004. [Citace: 28. Červenec 2011.] <http://cs.wikipedia.org/wiki/Html>.

— **2004.** Java (programovací jazyk). *Wikipedie, otevřená encyklopedie*. [Online] 22. Červen 2004. [Citace: 5. Srpen 2011.] Java (programovací jazyk).

— **2004.** Kaskádové styly. *Wikipedie, otevřená encyklopedie*. [Online] 28. Říjen 2004. [Citace: 30. Červenec 2011.] <http://cs.wikipedia.org/wiki/Css>.

**Žďárek, Roman. 2011.** Události v JavaScriptu | JavaScript. *Tvorba webu, Tvorba WWW stránek - Návod zdarma*. [Online] 2011. [Citace: 31. Červenec 2011.] <http://www.jakdelatweby.cz/javascript/udalosti.php>.

— **2011.** Úvod do CSS | CSS. *Tvorba webu, Tvorba WWW stránek - Návod zdarma*. [Online] 2011. [Citace: 30. Červenec 2011.] <http://www.jakdelatweby.cz/css/uvod.php>.

— **2011.** Úvod do JavaScriptu | JavaScript. *Tvorba webu, Tvorba WWW stránek - Návod zdarma*. [Online] 2011. [Citace: 31. Červenec 2011.] <http://www.jakdelatweby.cz/javascript/uvod-do-javascriptu.php>.

— **2011.** Úvod do jazyka HTML. *Tvorba webu, Tvorba WWW stránek - Návod zdarma*. [Online] 2011. [Citace: 28. Červenec 2011.] <http://www.jakdelatweby.cz/html/uvod.php>.

## Příloha A – Ukázka validního HTML dokumentu

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
  <head>
    <title>Titulek html stránky</title>
  </head>
  <body>
    <!-- komentář v html stránce -->
    Obsah html stránky.
  </body>
</html>
```



## Příloha B – Javascript a přehled událostí

Událost	HTML tagy	Vyvolání události	Podporované prohlížeče
onabort	<image>	Přerušení načítání obrázku	IE, Firefox, Opera, Chrome, Safari
onerror	<image>, <style>, <object>	Chyba načítání obrázku nebo dokumentu	IE, Firefox, Opera, Chrome, Safari
onload	<image>, <body>	Obrázek, nebo stránka, byl načten	IE, Firefox, Opera, Chrome, Safari
onblur	Většina tagů	Element ztratil fokus	IE, Firefox, Opera, Chrome, Safari
onchange	<input>, <select>, <textarea>	Změna obsahu textového pole	IE, Firefox, Opera, Chrome, Safari
onclick	Většina tagů	Kliknutí na element	IE, Firefox, Opera, Chrome, Safari
ondblclick	Většina tagů	Dvojklik na element	IE, Firefox, Opera, Chrome, Safari
onfocus	Většina tagů	Element dostal fokus	IE, Firefox, Opera, Chrome, Safari
onkeydown	Většina tagů	Stisknutí klávesy na klávesnici	IE, Firefox, Opera, Chrome, Safari
onkeypress	Většina tagů	Stisknutí, nebo držení, klávesy na klávesnici	IE, Firefox, Opera, Chrome, Safari
onkeyup	Většina tagů	Zvednutí stisknuté klávesy na klávesnici	IE, Firefox, Opera, Chrome, Safari
onmousedown	Většina tagů	Stisknutí tlačítka myši	IE, Firefox, Opera, Chrome, Safari
onmousemove	Většina tagů	Posunutí ukazatele myši po elementu	IE, Firefox, Opera, Chrome, Safari
onmouseout	Většina tagů	Posunutí ukazatele myši mimo element	IE, Firefox, Opera, Chrome, Safari
onmouseover	Většina tagů	Posunutí ukazatele myši na element	IE, Firefox, Opera, Chrome, Safari
onmouseup	Většina tagů	Zvednutí stisknutého tlačítka myši	IE, Firefox, Opera, Chrome, Safari
onresize	Většina tagů	Změna velikosti okna prohlížeče	IE, Firefox, Opera, Chrome, Safari
onselect	<input>, <textarea>	Označení nějakého textu z textového pole	IE, Firefox, Opera, Chrome, Safari
onunload	<body>, <frameset>	Uživatel opouští stránku	IE, Firefox, Opera, Chrome, Safari
onsubmit	<form>	Odeslání formuláře	IE, Firefox, Opera, Chrome, Safari
onreset	<form>	Stisknutí tlačítka reset ve formuláři	IE, Firefox, Opera, Chrome, Safari
onmove	Většina tagů	Změna pozice levého horního rohu elementu	IE
onmoveend	Většina tagů	Dokončení přesouvání elementu	IE
onmovestart	Většina tagů	Začátek přesouvání elementu	IE

onoffline	<body>	Nastavení prohlížeče pro práci v offline režimu	IE, Firefox
ononline	<body>	Nastavení prohlížeče pro práci v online režimu	IE, Firefox
oncut	Většina tágů	Před vystřížením označeného textu elementu	IE, Firefox, Chrome, Safari
onbeforecut	Většina tágů	Před vystřížením označeného textu elementu, umožňuje akci přerušit	IE, Firefox, Chrome, Safari
oncopy	Většina tágů	Před zkopírováním do schránky	IE, Firefox, Chrome, Safari
onbeforecopy	Většina tágů	Před zkopírováním do schránky, umožňuje akci přerušit	IE, Chrome, Safari
onpaste	Většina tágů	Před vložením textu do elementu	IE, Firefox, Chrome, Safari
onbeforepaste	Většina tágů	Před vložením textu do elementu, umožňuje akci zrušit	IE, Chrome, Safari
onresize	Většina tágů	Změna velikosti elementu	IE, Firefox, Opera, Chrome, Safari
onresizestart	Většina tágů	Začátek změny velikosti elementu	IE
onresizeend	Většina tágů	Konec změny velikosti elementu	IE
onscroll	<body> a další tágy	Posunutí textu nahoru, dolů, doleva, doprava	IE, Firefox, Opera, Chrome, Safari
onmousewheel	Většina tágů	Otočení kolečkem na myši	IE, Opera, Chrome, Safari
onsearch	<input type="search">	Stisknutí „ENTER“ pro hledání	Chrome, Safari
onselectionchange	document (Javascript)	Změna výběru textu na stránce	IE
onselectstart	Většina tágů	Začátek výběru textu elementu	IE, Chrome, Safari
onstart	<marquee>	Začátek animace textu elementu Marquee	IE, Firefox
onfinish	<marquee>	Konec animace textu elementu Marquee	IE, Firefox
onbounce	<marquee>	Při dotyku textu okraje elementu Marquee	IE, Firefox
onstop	document (Javascript)	Uživatel přeruší načítání stránky	IE
onafterupdate	<input>, <div>, <span>, <textarea>, atd	Po úpravě a odeslání dat do uložení (XML, atd)	IE
onbeforeupdate	<input>, <div>, <span>, <textarea>, atd	Po úpravě a před odesláním dat do uložení (XML, atd)	IE

onerrorupdate	<input>, <div>, <span>, <textarea>, atd	Při chybném pokusu o úpravu dat v uložišti (XML, atd)	IE
onactive	Většina tágů	Element se stává aktivní	IE
onbeforeactive	Většina tágů	Před tím, než se element stane aktivním	IE
ondeactive	Většina tágů	Element přestává být aktivní	IE
onbeforedeactive	Většina tágů	Před tím, než se element stane neaktivní	IE
onbeforeunload	<body>, <frameset>	Před vypnutím stránky (možno vložit potvrzovací dialog)	IE, Firefox, Chrome, Safari

Použité zdroje pro informace o událostech (Dottoro.com, 2011) a (Refsnes Data, 2011).

## Příloha C – Uživatelské rozhraní kGallery

V této příloze je kompletní seznam uživatelsky nastavitelných položek javascriptové galerie kGallery. Všechny položky jsou konfigurovatelné pomocí rozhraní tvořeného metodami třídy s názvem „kGallery“. To znamená, že pro provedení konfigurace nejdříve musíte vytvořit instanci třídy „kGallery“, poté nastavit patřičné položky a nakonec galerii spustit metodou „startKGallery()“. Některé možnosti konfigurace zde nejsou uvedeny, protože nebyly dotaženy do konce, takže je nelze snadno použít, nebo jejich užití nemá pro běh galerie smysl.

```
/**
 Pomocí této metody nastavujete, zda se mají obrázky začít načítat
 okamžitě při zobrazení stránky, či nikoli.
 Výchozí hodnota - false.
 Hodnota true znamená okamžité načítání.
 Hodnota false znamená, že se obrázky načítat nebudou.
 */
setPreloadovatVse (boolean);

/**
 Tato metoda nastavuje, zda se mají obrázky inteligentně načítat ještě
 před zobrazením, či nikoli. Funguje to tak, že při zobrazeném obrázku
 se přednačtou obrázky, na které můžete přejít okamžitě z aktuálního
 obrázku (první, předchozí, následující, poslední).
 Výchozí hodnota - true.
 Hodnota true znamená, že se obrázky budou načítat předběžně.
 Hodnota false znamená, že se obrázky načítat předem nebudou.
 */
setPreloadovatPrvkyDolniListyNavigace (boolean);

/*
 Umožňuje nastavení, zda má galeri pro svůj chod využívat hash v URL
 adrese, či nikoli. Více o této funkci zjistíte v kapitole Hash v URL
 adrese.
 Výchozí hodnota - true.
 Hodnota true polovuje využití hash.
 Hodnota false použití hash zakazuje.
 */
setPouzivatHash (boolean);

/*
 Nastavuje barvu pozadí webové stránky při otevřeném náhledu velkého
 obrázku. Toto pozadí skryje veškerý obsah webové stránky, aby nerušil
 při prohlížení obrázků.
 Výchozí hodnota - #000000.
 Očekává barvu v hexadecimálním tvaru s dvojitým křížkem na začátku,
 nebo jednu ze základních barev (black, green, blue, yellow, atd).
 */
setBarvaPozadi (string);
```

```

/*
Nastavuje intenzitu barvy pozadí při otevřeném náhledu velkého
obrázku.
Výchozí hodnota - 95.
Očekává hodnotu od 0 do 100, kde 0 je neviditelné pozadí a 100 je
hodnota, která zajistí naprosté překrytí veškerého obsahu stránky.
*/
setIntenzitaPozadi(int);

/*
Tato hodnota udává, kolik milisekund bude trvat, než intenzita pozadí
naběhne z hodnoty 0 na požadovanou hodnotu.
Výchozí hodnota - 100.
Očekává hodnotu od 0 a horní hranice není omezena, ale je doporučeno
volit hodnoty do 1000, aby se pozadí nezobrazovalo příliš dlouho a
nerušilo uživatele při prohlížení obrázků.
*/
setDobaZobrazovaniPozadi(int);

/*
Tato hodnota udává plynulost změny intenzity pozadí. Čím vyšší
hodnota, tím kvalitnější přechod z hodnoty intenzity 0 do požadované
hodnoty.
Výchozí hodnota - 80.
Očekává hodnotu od 1 a horní hranice není omezena, ale je doporučeno
volit hodnoty do 100, protože čím vyšší hodnotu zvolíte, tím vyšší
jsou nároky na výpočetní výkon uživatelského počítače.
*/
setFpsZobrazovaniPozadi(int);

/*
Tato hodnota je udávána v pixelech a určuje tloušťku rámečku kolem
obrázku při zobrazeném velkém náhledu. Horní část rámečku bude vždy
minimálně 25 pixelů tlustá, aby se tam vešla informace o pořadí
aktuálního obrázku a nápis „Zavřít“.
Výchozí hodnota - 5.
Očekává hodnotu od 0 a horní hranice není omezena, ale je doporučeno
volit hodnoty do 30, aby rámeček nezabíral příliš mnoho prostoru,
který může být využit pro zobrazení obrázku.
*/
setSirkaRameckuObrazku(int);

/*
Tato hodnota je udávána v pixelech a určuje vzdálenost, kterou
obrázek, případně rámeček, bude držet od kraje okna prohlížeče.
Výchozí hodnota - 2.
Očekává hodnotu od 0 a horní hranice není omezena, ale je doporučeno
volit hodnoty do 30, aby tento volný prostor zbytečně neubíral místo
pro zobrazení obrázku.
*/
setVzdalenostOdKrajeMonitoru(int);

```

```

/*
Nastavuje barvu rámečku kolem obrázku při zobrazeném velkém náhledu.
Zároveň je tato barva použita na horní lištu, na které je nápis
„Zavřít“ a informace o pořadí aktuálního obrázku.
Výchozí hodnota - white.
Očekává barvu v hexadecimálním tvaru s dvojitým křížkem na začátku,
nebo jednu ze základních barev (black, green, blue, yellow, atd).
*/
setBarvaRameckuObrazku(string);

/*
Nastavuje barvu textu pro horní část rámečku, kde je nápis „Zavřít“ a
informace o pořadí aktuálního obrázku. Barva by měla být volena
dostatečně kontrastní od barvy rámečku obrázku.
Výchozí hodnota - black.
Očekává barvu v hexadecimálním tvaru s dvojitým křížkem na začátku,
nebo jednu ze základních barev (black, green, blue, yellow, atd).
*/
setBarvaTextuRameckuNaObrazku(string);

/*
Nastavuje barvu pozadí navigační lišty umístěné v dolní části okna
prohlížeče. Na této liště jsou tlačítka sloužící k procházení obrázků
(první, předchozí, následující, poslední).
Výchozí hodnota - white.
Očekává barvu v hexadecimálním tvaru s dvojitým křížkem na začátku,
nebo jednu ze základních barev (black, green, blue, yellow, atd).
*/
setBarvaPozadiNavigacniListy(string);

/*
Nastavuje intenzitu barvy pozadí navigační lišty.
Výchozí hodnota - 100.
Očekává hodnotu od 0 do 100, kde 0 je neviditelné pozadí a 100 je
hodnota, která zajistí naprosté překrytí prvků umístěných ve vrstvách
pod lištou.
*/
setIntenzitaPozadiNavigacniListy(int);

/*
Nastavuje intenzitu prvků, které v daný moment nelze využít. Například
při prohlížení obrázků dojdete na poslední a už nelze jít dále. V tu
chvíli se nastaví intenzita prvku „další“ a „poslední“ na hodnotu
nastavenou touto metodou.
Výchozí hodnota - 20;
Očekává hodnotu od 0 do 100, kde 0 naprosto skryje daný prvek a
hodnota 100 intenzitu oproti aktivním prvkům nezmění.
*/
setIntenzitaNeaktivnihoPrvkuListyNavigace(int);

/*
Nastavuje umístění obrázků navigačních prvků pro dolní lištu. Umístění
se udává relativní z pohledu souboru, ve kterém je navigace umístěna,
nebo absolutní.
Výchozí hodnota - kGallery/js/img/.
Očekává relativní, nebo absolutní cestu k adresáři, kde jsou umístěny
obrázky pro prvky navigace.
*/
setSlozkaPrvkuDolniListyNavigace(string);

```

## **Příloha D – Obsah CD přiloženého k bakalářské práci**

CD obsahuje zdrojové kódy programu kGenerator v podobě projektu v programu jDeveloper. Projekt je včetně dat potřebných k vygenerování galerie. Dále CD obsahuje adresář se spustitelným programem kGenerator na operačním systému Windows. Nechybí ani adresář se soubory, které mohou posloužit jako zkušební zdroj generování webové galerie. Poslední důležitou položkou obsahu CD jsou dokumentace, a to jak uživatelská, tak programátorská.