

Západočeská univerzita v Plzni

Fakulta pedagogická

Bakalářská práce

WEBOVÁ GALERIE PRO ŠKOLNÍ DÍLA

Martin Šnajdr

Vizuální kultura se zaměřením na vzdělávání

Vedoucí práce

PhDr. Vladimíra Zikmundová, Ph.D.

Plzeň 2013

Prohlašuji, že jsem práci vypracoval samostatně s použitím uvedené literatury a zdrojů informací.

V Plzni 1. července 2013

Podpis:.....

Poděkování:

Rád bych poděkoval vedoucí mé bakalářské práce PhDr. Vladimíře Zikmundové, Ph.D. za ochotu, vstřícnost a trpělivost při realizaci práce.

V Plzni 1. července 2013

Podpis:.....

Anotace

Cílem mé bakalářské práce bylo navrhnout a zrealizovat webovou galerii Katedry výtvarné kultury, která by byla vhodná pro prezentování uměleckých děl studentů katedry. Písemná část práce obsahuje základní přehled pojmů v oblasti vývoje webových aplikací, inspirační zdroje, použité technologie, samotný popis tvorby mého řešení webové galerie pro školní díla a také shrnutí poznatků a nových zkušeností, které jsem při této tvorbě získal.

Webová galerie je dostupná z adresy [červenec 2013]: **<http://galerie.martinsnajdr.com>**

Klíčová slova: aplikace, programování, galerie, web, internet, umění

Annotation

The result of my bachelor thesis was to design and create an interactive web gallery for the Department of Visual Culture, which could be an additional way to present the artworks of the students of our department. The theoretical part of my work includes some basic information about the web application development, my inspiration sources and used technologies. The practical part consists of two chapters – a brief description of the development of my web gallery application and a summary of a new experience that I have gained by creating this work.

The web gallery is available on [july 2013]: **<http://galerie.martinsnajdr.com>**

Keywords: application, programming, gallery, web, internet, art

Obsah

1 Úvod	3
2 Základní pojmy v oblasti webu	5
2.1 Historie vývoje webu a webových prohlížečů.....	5
2.2 Technologie	7
2.2.1 Client-side technologie	7
2.2.2 Server-side technologie	9
2.3 Grafické uživatelské rozhraní	10
2.4 UX.....	12
2.5 Bezpečnost.....	13
3 Příprava na realizaci webové galerie	15
3.1 Inspirace.....	15
3.1.1 Švýcarský grafický design a Bauhaus	15
3.1.2 Metro UI	16
3.1.3 Cargo Collective	17
3.2 Galerie jiných uměleckých škol.....	18
3.3 Použité technologie a nástroje	19
3.3.1 XAMPP	20
3.3.2 jQuery	20
3.3.3 LESS.....	21
3.3.4 PHP a MySQL.....	21
3.3.5 Github	22
3.3.6 Sublime Text editor	23
4 Realizace webové galerie	24
4.1 Adresářová, souborová a databázová struktura	24
4.2 MVC architektura	25
4.3 Uživatelské rozhraní webové galerie.....	26

4.3.1 Administrace.....	26
4.3.2 Frontend.....	28
5 Závěr.....	30
6 Resume	31
Seznam použité literatury	32
Elektronické zdroje.....	33
Obrazová příloha	I

1 Úvod

Téma mé bakalářské práce mi bylo přímo navrženo vedoucí naší katedry PhDr. Vladimírou Zikmundovou, Ph.D. V podstatě jsem ani chvíli neváhal, jelikož z nabízených témat se mi možnost vytvořit interaktivní webovou galerii Katedry výtvarné kultury (dále jen KVK) zamlouvala nejvíce.

Tvorbou webových stránek (dnes je spíše přesnější termín webová aplikace) se zabývám více jak deset let a tak moc dobře vím, jak velký kus cesty za tu dobu internet urazil. Internetové stránky se během chvíle přetransformovaly z jednoduchých prezentací (podobajících se spíše textovým dokumentům) na komplikované, interaktivní aplikace, které v podstatě zvládnou zastat mnoho úkolů, které se již blíží úrovni desktopových (či nativních) aplikací¹.

Od první chvíle mi bylo jasné, že se při zpracování tohoto tématu setkám s jedním základním problémem při tvorbě aplikace jako odborné práce – chybějící tištěná literatura (především česká). Důvod je prostý – vývoj webových prohlížečů jde neustále kupředu. Každý měsíc se objevují aktualizované verze webových prohlížečů, které přidávají stále nové funkce. Autoři literatury o tvorbě webových aplikací tak musejí minimálně jednou za dva roky vydávat přepracované publikace, které alespoň z části doženou vše, co se stačilo změnit. Nedá se říci, že se vývoj aplikací vždy změní od základu, ale častokrát i jedna nová funkce může změnit náhled na řešení mnoha jiných problémů. Publikace v této oblasti jsou tedy vhodné především pro začátečníky, či středně pokročilé vývojáře, kteří spíše potřebují hrubý nástin, jak funguje webová aplikace v základu. Pokud však člověk řeší konkrétní problém, prohledávání webu Googlem je volbou číslo jedna.

Pro mne jako pro vývojáře jsou nejcennějšími zdroji informací technologické blogy, které píše lidé z odborné veřejnosti. Samozřejmě ve většině případů jsou anglicky psané.

V následující kapitole své bakalářské práce se zabývám základními pojmy v oblasti webových aplikací, technologiemi, trendy, kterými procházely a celkově uživatelskými rozhraními. Třetí kapitola je věnovaná mým inspiračním zdrojům pro webovou galerii a výběrem vhodných nástrojů pro její realizaci. Čtvrtá kapitola popisuje průběh tvorby galerie, od základních postupů až po další možnosti rozvoje.

¹ Nativní aplikace – program napsaný přímo na daný operační systém.

Ve své práci se snažím klást důraz na didaktický přesah celého textu tak, aby kdokoli, kdo má základní znalosti v oblasti IT, byl schopen pochopit celý průběh tvorby podobného projektu. Některé technické části se snažím zjednodušit takovým způsobem, aby zbytečně nezasahovaly do oblastí, které nejsou pro moji práci a především pro čtenáře tolik podstatné.

2 Základní pojmy v oblasti webu

V této kapitole se zabývám základními pojmy, které je nutné znát pro bezproblémovou orientaci v kapitolách následujících.

2.1 Historie vývoje webu a webových prohlížečů

Jak jsem se již zmínil výše, termín webová stránka a webová aplikace jsou dvě odlišné věci. Zatímco webová stránka je to, co se nám zobrazí při zadání adresy v internetovém prohlížeči, webovou aplikací rozumíme vše, co běží puštěné na daném serveru, ze kterého se nám webová stránka zobrazuje. Moderní webové prezentace už dávno nejsou jen pouhými nositeli informací, ale dovolují člověku mnohem více. Od nejrůznějších animací, dynamického načítání obsahu bez nutnosti obnovení stránky až po editování dokumentů a využití vestavěného hardwaru počítače.

První webová stránka byla umístěna na web v roce 1993 evropskou organizací CERN. Letos oslavila své dvacáté narozeniny a při příležitosti tohoto výročí ji CERN obnovil – dokonce pod stejnou webovou adresou.²

O rok později, tedy v roce 1994, založil Tim Berners-Lee konsorcium W3C, které dostalo za úkol vytvořit standardizovaný webový jazyk HTML³. Konsorcium působí dodnes a je spravováno několika významnými institucemi: MIT CSAIL⁴, ERCIM⁵ a Keyo Univerzity (Japonsko). W3C udává veškeré standarty, které by měly moderní webové prohlížeče podporovat.

První velký nárůst uživatelů internetu přišel s rokem 1995, kdy Microsoft představil první verzi svého webového prohlížeče Internet Explorer (IE) určenou pro Windows 95. V průběhu let přicházel Microsoft s novými verzemi tohoto prohlížeče a dá se s nadsázkou říci, že tím získal „*monopol na celý internet*“. Bohužel to mělo za následek zpomalení vývoje dalších prohlížečů a tím i pomalou implementaci⁶ nových technologií. Tato situace se začala zlepšovat až s příchodem prohlížeče Mozilla (později Mozilla Firefox), která svými moderními standarty rozproudila dění okolo vývoje webu. Internet Explorer

² *Info.cern.ch - the world's first website*. [online]. 2013 [cit. 2013-07-01]. Dostupné z: <http://info.cern.ch/>

³ Hyper text markup language, v překladu: hypertextový značkovací jazyk

⁴ Computer Science and Artificial Intelligence Laboratory, v překladu: informatika a laboratoř umělé inteligence.

⁵ European Research Consortium for Informatics and Mathematics, v překladu: Evropské výzkumné konsorcium informatiky a matematiky

⁶ Začleňování do jádra programu.

vedle Mozilly působil jako prehistorický software. Přestože Mozilla přinesla spoustu skvělý prostor pro nové vyjádření a obohacení grafického rozhraní webu, Internet Explorer si stále statisticky držel velký náskok, a tak vývojářům nezbylo nic jiného, než optimalizovat weby pro Mozillu a Internet Explorer zvlášť. Ve výsledku to znamenalo, že většina moderních prvků musela být stejně z webu odstraněna, aby nedocházelo ke špatnému zobrazování právě v Internet Exploreru. Tím začalo asi nejhorší období pro všechny, kteří se tvorbou webových stránek zabývali.

Až v roce 2008 svitla vývojářům nová naděje na zlepšení stavu – Google Chrome. Tento prohlížeč si ihned získal přízeň nejen IT odborníků, ale i široké veřejnosti, a to především díky své jednoduchosti, rychlosti a nebývale rychlou implementací standardů. V roce 2012 předehnal Google Chrome ve statistikách podílu na trhu Internet Explorer. Google Chrome je i můj primární webový prohlížeč, který používám při práci i pro surfování na internetu. Bohužel ani v roce 2013 se nedá vyhnout nutnosti optimalizovat⁷ webové stránky pro různé prohlížeče, avšak tato optimalizace již není tak náročná, jako bývala dříve.

Ačkoliv se to na první pohled nezdá, vývoj internetu vždy závisel a dodnes závisí na vlastnostech webového prohlížeče. Tento rozvoj jde ruku v ruce s neustálým zrychlováním počítačového hardwaru a také se zlepšováním dostupnosti internetového připojení. Mobilní operátoři nabízejí spoustu možností, jak se připojit nejen doma, ale i na cestách. Avšak v České republice jsme oproti západním zemím stále o dost pozadu, co se týče rychlého a dostupného mobilního internetu. Během několika málo let se ale situace nejspíše markantně změní a trend cloudových⁸ úložišť dat a jejich streamování⁹ z internetu se brzy dostane také k nám. V USA je v nyní běžné nemít hudbu přímo na přehrávacím zařízení, nýbrž jí streamovat ze serveru poskytovatele některé z hudebních služeb. Důvody jsou prosté – úspora úložné kapacity elektronických zařízení a také mnohem větší výběr hudebních alb a interpretů. Samozřejmě je zde mnoho nevýhod cloudových služeb. Jako zásadní problém se jeví nemožnost využití těchto služeb, když je uživatel offline. V diskusích na internetu je často probírána otázka osobního vlastnictví.

⁷ Postup, při kterém se webová stránka testuje v daném prohlížeči. Je součástí zejména finálního vývoje webové stránky.

⁸ Cloud computing – poskytování služeb či programů uložených na vzdálených serverech, ke kterým mohou uživatelé přistupovat skrz webový prohlížeč nebo aplikaci napsanou přímo pro danou službu.

⁹ Streaming je technologie přenosu dat mezi serverem a uživatelem.

Majitel služby nám data vlastně jen propůjčuje, protože jsou stále uložena jen u něj na serveru, což ne všem zákazníkům vyhovuje, vzhledem k tomu, že za tato data zaplatili.

2.2 Technologie

Při vývoji moderního webu se využívá hned několika programovacích jazyků¹⁰. Ty se dají rozdělit na tzv. client-side a server-side. Client-side znamená vše, co se odehrává v prohlížeči (na klientově straně). Jde o zpracování HTML, CSS a Javascript kódu¹¹. Na druhé straně najdeme server-side jazyky, které běží na serveru a starají se o poskládání kódu, který se odešle návštěvníkovi do prohlížeče. Mezi nejpoužívanější server-side jazyky patří PHP¹², Python a Ruby.

Vzhledem k tomu, že se postupem času začaly webové aplikace rozrůstat do tisíců řádků kódu a implementace nových funkcí přímo do jádra programovacího jazyka je běh na dlouhou trať, přišla komunita vývojářů s ideou frameworků. Framework je určitou nadstavbou programovacího jazyka. Můžeme si ho představit jako knihovnu funkcí, které programátoři opakovaně používají ve svých aplikacích, ale přitom chybí v jádru daného jazyka. Frameworky pomáhají ostatním programátorům k rychlejšímu vývoji aplikací a na internetu jich najdeme nepřeborné množství. Většina z nich je licencovaná MIT¹³ nebo GNU-GPL¹⁴ licencemi a tudíž jsou volně ke stažení pro soukromé i komerční využití.

2.2.1 Client-side technologie

Základním stavebním kamenem webu je jazyk HTML, který určuje základní strukturu a obsah webu. HTML vychází z jazyka XML¹⁵, základní elementy se u obou definují tzv. tagy¹⁶. Tag reprezentuje sémantickou část obsahu, tudíž definuje význam jednotlivých částí webové stránky. Každý HTML element má také své atributy, které definují některé jeho vlastnosti. Kupříkladu atributem *id* označujeme unikátní název pro daný element. Atribut *class* definuje třídu, do které element patří a je využíván především při stylování vizuální stránky webu.

¹⁰ Programovacím jazykem rozumíme prostředek komunikace mezi programátorem a počítačem. Programátor určuje za pomoci programu, co má počítač dělat a počítač tyto příkazy poté interpretuje.

¹¹ Zápis textu počítačového programu.

¹² Rekurzivní zkratka pro: Hypertext Preprocessor – v překladu hypertextový preprocesor.

¹³ Open Source Initiative. *The MIT License* [online]. nedatováno [cit. 2013-07-01]. Dostupné z: <http://opensource.org/licenses/MIT>

¹⁴ GNU Operating System. *The GNU General Public License* [online]. nedatováno [cit. 2013-07-01]. Dostupné z: <http://www.gnu.org/licenses/gpl.html>

¹⁵ Extensible markup language, v překladu: rozšiřitelný značkovací jazyk

¹⁶ Příklad tagu definujícího odstavec textu: `<p>Obsah odstavce</p>`

Aktuální verze HTML5 je nejnovější generací tohoto jazyka. HTML5 obsahuje spoustu nových funkcí potřebnou pro moderní webové aplikace. Mezi nimi nalezneme např. video a audio tag. Také standardizuje mnoho funkcí, které používali vývojáři již pár let, ale nikdy nebyly řádně zdokumentované nebo je podporovala jen část webových prohlížečů (volně přeloženo¹⁷).

Nutnou součástí tvorby webových stránek je znalost kaskádových stylů známých pod zkratkou CSS. Přestože měly dříve webové stránky především informativní charakter s minimem grafických prvků, přišla doba, kdy jednoduchá bílá stránka s černým textem a modrými hypertextovými odkazy přestala stačit. V počátcích se vizuální stránka určovala přímo u konkrétního HTML tagu, avšak nedlouho potom se přišlo W3C s lepším řešením v podobě CSS.

CSS definuje základní styly, které mohou být přiřazeny k danému HTML elementu. K přesné determinaci elementu, kterému chci přidat daný styl, slouží výše zmíněné atributy třídy, id a další. CSS vlastností, jež můžeme elementu přiřadit, je nepřeberné množství. Od pozice na stránce, přes barvu textu a pozadí až po animace při najetí kurzoru na element. Právě CSS animace jsou asi největším posunem mezi CSS 2.0 a CSS 3.0 a umožňují vývojářům nastavit plynulé přechody mezi některými vlastnostmi elementu (např. plynulá změna barvy pozadí z modré na černou). Většina vlastností CSS 3.0 byla podporována prohlížeči o mnoho let dříve za pomoci tzv. CSS hacků. CSS hackem rozumíme jakési neoficiální řešení, kterým se dá obejít nefunkčnost některých CSS vlastností (volně přeloženo¹⁸).

Poslední z využívaných client-side technologií je skriptovací jazyk¹⁹ Javascript. Bývá často dáván do spojitosti s velice rozšířeným programovacím jazykem Java, se kterým však nesdílí více, než jen podobnou syntaxi. Javascript umožňuje ovládat interaktivní prvky uživatelského rozhraní. Dokáže přímo v prohlížeči manipulovat se CSS vlastnostmi HTML elementů, přidávat elementy nové a také vázat na tyto elementy nejrůznější události. Javascript nám umožňuje vytvořit animovanou prezentaci obrázků

¹⁷ PILGRIM, Mark. *HTML 5: Up and Running*. California, USA: O'Reilly Media, 2010. 1. ISBN 978-0596806026. s 9.

¹⁸ GASSTON, Peter. *The modern Web: multi-device Web development with HTML5, CSS3, and JavaScript*. San Francisco: No Starch Press, 2013, 264 s. ISBN 15-932-7487-4. s 29.

¹⁹ Jednotná definice skriptovacího jazyka neexistuje. Slovo skriptovací v mnoha případech spíše nahrazuje termín programovací.

nebo načítat ze serveru pouze části webové stránky a vkládat je do prohlížeče bez nutnosti obnovení celého okna. Tato technologie asynchronního načítání obsahu se nazývá AJAX²⁰.

2.2.2 Server-side technologie

Termín dynamické webové stránky je úzce spjatý se server-side technologiemi. Laicky řečeno, tyto technologie se zabývají jediným základním úkolem – poskládat HTML kód, který se odešle do webového prohlížeče, ten ho dále zpracuje a ve finále zobrazí webovou stránku. Statické webové stránky se na rozdíl od těch dynamických musí komplikovaně upravovat přímo v editoru, což je nepohodlné především pro správce stránek, kteří nemají znalosti v oblasti programování webových aplikací. S čistě statickými webovými stránkami se dnes již nemáme moc šancí setkat, především díky mnoha open-source CMS²¹ aplikacím.

Abychom mohli vůbec některou ze server-side technologií používat, musíme mít nejdříve nakonfigurovaný HTTP server, na kterém se veškeré skripty spouští. Existuje několik HTTP serverových řešení, ale v podstatě stačí znát jediné – Apache server. Apache server je zdarma a jeho instalace a konfigurace zabere jen několik minut. Jednotlivé servers-side technologie se poté přidávají do Apache v podobě modulů. Základní placené webhostingy mají Apache server předinstalovaný včetně základních modulů, což je na jednu stranu o dost snadnější řešení pro nezkušené uživatele, ale na druhou stranu je tato konfigurace většinou pevně daná a nelze jí měnit. Dražší webhostingy (s označením virtual dedicated nebo dedicated) umožňují uživatelům nakládat se serverovým prostorem podle jejich potřeb – např. instalovat další moduly pro Apache nebo dokonce přeinstalovat celý operační systém serveru.

Princip většiny dynamických stránek je identický. Data (vše, co se stránku od stránky mění) jsou uložena v databázi. Databázi se rozumí místo, kde jsou jednotlivé položky uloženy v určité jednotné struktuře – tedy alespoň to platí pro databáze typu SQL²², které se pro tvorbu webových aplikací využívají nejčastěji. Tato data jsou za pomoci některého ze skriptovacích jazyků z databáze vytažena a dále se s nimi pracuje. Skriptovací jazyky nám umožňují vyplnit těmito daty HTML šablony, které se pak odešlou jako hotové HTML dokumenty přímo klientovi do prohlížeče.

²⁰ Asynchronous Javascript and XML

²¹ Aplikace pro správu obsahu, které jsou zdarma ke stažení.

²² Structured Query Language – jazyk používaný pro práci s daty v databázích

Mezi nejpoužívanější skriptovací jazyky určené pro programování webových aplikací je PHP. Podle statistik W3Techs je 80% všech webových aplikací na internetu postaveno právě na tomto jazyce.²³ Oblíbenost PHP stojí zejména na silné podpoře komunity vývojářů, webhostingů²⁴ a na tisících praktických příkladů zdrojových kódů, které můžeme na internetu nalézt. PHP bylo také jedním z prvních jazyků používaných pro psaní dynamických webových stránek a spousta vývojářů se s ním pro jeho jednoduchost nerada loučí, přestože jsou již k dispozici mnohem modernější a kvalitnější jazyky.

Mezi tyto modernější jazyky patří např. Ruby nebo Python, které sice nebyly primárně určeny jen pro tvorbu webu, ale byly rozšířeny díky webovým frameworkům, které umožnili programátorům nebývale rychlé prototypování²⁵ aplikací. Tyto frameworky jsou však vhodné pouze pro určitý typ aplikací, a proto si stále PHP drží přední příčku v oblíbenosti.

2.3 Grafické uživatelské rozhraní

Grafické uživatelské rozhraní (GUI²⁶) vytváří pomyslný most mezi uživatelem a počítačem. Za pomoci grafických ovládacích prvků (tlačítka, ikony, okna) je uživatel schopen manipulovat s objekty na obrazovce, a tak jednoduše zadávat počítači úkoly, které má vykonávat.

„Každá dobrá aplikace musí být nejen dobře naprogramována ať už z hlediska efektivity fungování, nebo vnitřní architektury, ale musí být také dobře použitelná. Dobré uživatelské rozhraní umožňuje uživateli pracovat s aplikací účelně a pohodlně. Rozhraní, které navíc dodržuje předepsaná pravidla, viz níže, pomáhá uživateli vytvářet správné stereotypy práce se systémem. Dobré uživatelské rozhraní je také přínosem pro autory aplikací. Dobré řešení problému (aplikace) bývá zároveň jednodušší než špatné řešení. Autoři také nemusí psát tak složité a rozsáhlé manuály k aplikacím, protože uživatelé mohou uplatnit dobré návyky z ostatních aplikací.“²⁷

²³ W3Techs. *Historical trends in the usage of server-side programming languages for websites* [online]. 2013 [cit. 2013-07-01]. Dostupné z: http://w3techs.com/technologies/history_overview/programming_language

²⁴ Prostor na serveru pro umístění webových stránek.

²⁵ Proces mezi návrhem a první (částečně) funkční realizací.

²⁶ Graphic user interface.

²⁷ DOŠTÁL, Martin. *Základy tvorby uživatelského rozhraní* [online]. Olomouc, 2007 [cit. 2013-07-01]. Dostupné z: <http://phoenix.inf.upol.cz/esf/ucebni/gui-dostal.pdf>. Učební text. Univerzita Palackého, Přírodovědecká fakulta, Katedra informatiky.

Právě konzistentnost ovládacích prvků je jednou z nejdůležitějších vlastností sofistikovaného uživatelského rozhraní. Jako příklad si dovolím uvést mobilní operační systém od firmy Apple, který běží v zařízeních iPhone a iPad. iPhone je dotykový mobilní telefon s 3,5 až 4 palcovou obrazovkou. iPad je tablet s 7,9 až 9,7 palcovou obrazovkou. V nových zařízeních je vždy předinstalováno několik základních aplikací. Každá tato aplikace má sice jiný účel, ale zachovávají si podobné ovládací prvky, a tak pokud uživatel umí používat jednu z nich, není problém se naučit během chvíle pracovat s těmi ostatními. Stejně tak přechod mezi iPhonem a iPadem je pro uživatele velice snadný a plynulý, přestože se jedná o dvě vcelku rozdílná zařízení. iPad samozřejmě využívá možností větší obrazovky, ale v základu se od iPhonu příliš neliší. Pro většinu velkých platforem existují přímo psaná pravidla při tvorbě uživatelského rozhraní, kterými by se měli vývojáři aplikací třetích stran řídit. Těmto pravidlům se říká odborně Human Interface Guildlines (HIG)²⁸. Většinou obsahují jak abstraktnější části popisující zážitek, který má GUI v uživateli vyvolávat, tak i doporučené grafické prvky, barvy a fonty. Samozřejmě tato pravidla nejsou použitelná pro všechny typy aplikací – například hry se jimi příliš neřídí, protože mají častokrát svoji vlastní soustavu ovládacích prvků. Na první pohled se může zdát, že HIG se až moc snaží popřít originalitu designerů uživatelského rozhraní a natlačit je do jednotného stylu, ale z mé vlastní zkušenosti tomu tak není. Designéři spíše oceňují možnost mít v začátku z čeho vycházet, čím se inspirovat, než se znovu snažit „vynalézt kolo“.

Zaměříme-li se přímo na uživatelské rozhraní webových aplikací a stránek, můžeme se setkat se dvěma základními proudy. Tím prvním je klasický vzhled, vycházející z tištěné grafiky. Jedná se o stránky založené primárně na typografii, které jsou více či méně obohaceny grafickými prvky a slouží většinou jako webové prezentace firem, osobní stránky nebo jako elektronické deníky, blogy. Druhým proudem je vzhled inspirovaný uživatelským rozhraním desktopových aplikací. Tento proud se začal objevovat až v posledních pár letech s příchodem nových možností webových prohlížečů. Došlo k určité transformaci náhledu na účel webových stránek, které již nemusí sloužit pouze pro konzumaci hotového obsahu, ale také k tvorbě obsahu nového. Jako příklad uvedu službu Google Docs, jež funguje jako online verze programů podobných balíku MS Office. V blízké budoucnosti se zajisté bude tento trend dále rozvíjet, první vlnou

²⁸ Nemá přesný překlad. Volně přeloženo jako: pravidla uživatelského rozhraní zaměřeného na člověka / uživatele.

je operační systém ChromeOS od Googlu, který je ve své podstatě pouze a jen webový prohlížeč a veškeré aplikace na něm běžící jsou umístěny na internetu. V současné době zatím nejsme zdaleka na podobné operační systémy připraveni (hlavně kvůli špatnému pokrytí mobilním internetem mimo velká města), ale až se situace zlepší, určitě se najde místo i pro tento typ systémů.

Trendy, jimiž vzhled webových stránek prošel, vždy závisely na možnostech webových prohlížečů. První webové stránky postrádaly veškeré grafické prvky. S příchodem CSS a možností přidávat bitmapovou grafiku na web se začal formovat webdesign jako samostatný obor. Práce webdesignera představovala hodiny strávené ve Photoshopu a další hodiny strávené „řezáním“ vytvořené grafiky do jednotlivých obrázků, které poté byly znovu poskládány v CSS. Vrchol tohoto období se nazývá Web 2.0, kdy na stránkách převládaly spousty velkých sytě barevných prvků plných všemožných odlesků a stínů. Tento trend skončil okolo roku 2009, kdy se designeři zpět obrátili k vyzdvihnutí obsahu webů především poutavou a originální typografií, což vlastně trvá do současnosti. Zás a znovu v tom měly prsty webové prohlížeče, které začaly nabízet možnost používat ve webdesignu takřka jakékoli fonty bez nutnosti jejich přítomnosti na uživatelské počítači.

2.4 UX

Pojem UX čili User Experience²⁹ není zatím ještě v našich vodách úplně zabydlený, ale v zahraničních softwarových firmách se skloňuje ve všech pádech. Zatímco dříve se vývojáři zabývali především použitelností softwaru, dnes již samotná použitelnost nestačí a to hlavně díky velkému soutěžení na poli platform a produktů. UX je možné definovat jako něco, co jde ještě daleko za pouhou použitelnost softwaru, protože se jedná o design zaměřený více na samotného člověka nežli na technologie. User Experience zahrnuje sociální a kulturní hodnoty, emoční vliv – zkrátka zdali interakce člověka se zařízením zahrnuje pocit radosti z používání, estetický zážitek a celkově ji uživatele označují jako zábavnou (volně přeloženo).³⁰

Význam UX se dá popsat příslovím, jež praví, že když dva dělají totéž, není to totéž. Rozdíl v oblíbenosti dvou obdobných aplikací je v současnosti ovlivněn obzvláště

²⁹ V překladu: uživatelská zkušenost, uživatelský zážitek.

³⁰ HARTSON, Rex a Pardha PYLA. *The UX Book: process and guidelines for ensuring a quality user experience*. Boston: Elsevier, 2012, 937 s. ISBN 01-238-5241-2. s 12.

přívětivostí jejich uživatelského rozhraní a na uživatelském zážitku, která tato rozhraní přináší. Pokud jsou všechny prvky rozmístěny intuitivně a uživatel je schopen během několika kroků učinit přesně, co potřebuje, je zde velká šance, že aplikace bude mít úspěch. Na druhou stranu v grafických rozhraních, která jsou nepřehledná a přeplněná spoustou tlačítek a ikonek, se uživatel ztrácí a nakonec se raději takovému softwaru vyhne.

2.5 Bezpečnost

Otázka bezpečnosti webových stránek je stále více aktuální. Čím větší počet svých soukromých informací lidé internetu svěřují, tím větší jsou jejich obavy z možného úniku těchto informací. Bohužel podobné obavy jsou zcela namístě. Zejména bankovní weby či Facebook jsou takřka pod neustálým útokem hackerů, kteří se snaží získat jakékoli důvěrné informace o uživateli či zákaznících, jež by mohli dále využít.

Bezpečnost internetových aplikací závisí na mnoha faktorech. Jedná se o fyzickou bezpečnost serverů, kdy je nutné potencionálnímu útočníkovi zabránit v možnosti odcizení hardwaru. Dále o bezpečnost a konfiguraci aplikací a operačního systému, pod kterým daný server běží a v neposlední řadě o zabezpečení samotné webové aplikace, která vývojáře samozřejmě zajímá nejvíce.³¹

Obranou proti těmto útokům je na prvním místě šifrování přenosu dat mezi klientem a serverem a také několika stupňová přihlašovací ochrana. K šifrování přenosu se používají tzv. protokoly SSL³². SSL je certifikát, který vydávají důvěryhodné autority (firmy). Informace o SSL certifikátu nalezneme většinou nalevo od adresního řádku ve webovém prohlížeči. Pokud navštívíme stránku, která je zabezpečena důvěryhodným certifikátem, zobrazí se nám vedle webové adresy ikona malého zámku.

Autentifikace je proces, kdy uživatel zadává do přihlašovacího formuláře své uživatelské jméno (nebo email) a heslo. Při odeslání formuláře se data porovnají s databází, a jestliže se oba dva údaje shodují, systém provede přihlášení uživatele. Z bezpečnostního hlediska by mělo být heslo v databázi uloženo v zašifrované podobě, ale jsou známé také případy, kdy velké servery ukládaly hesla bez zašifrování a při úniku

³¹ MEJZR, Martin. *Bezpečnostní problémy internetových aplikací*. Praha, 2008. Dostupné z: https://dip.felk.cvut.cz/browse/pdfcache/mejzrm1_2008bach.pdf. Bakalářská práce. ČVUT, FEL, Katedra počítačů. Vedoucí práce Ing. Zdeněk Troníček Ph.D.

³² Secure Sockets Layer, v překladu: vrstva bezpečnostních socketů.

dat z databáze měly obrovský problém, což mělo za následek ztrátu důvěry jejich návštěvníků.

U menších aplikací, které nejsou pro hackery příliš zajímavé, není potřeba řešit bezpečnost do přílišné hloubky. Běžně postačí ukládat šifrovaná hesla, která se pak nedají zneužít při snaze dostat se do jiných účtů stejného uživatele a sanitizovat³³ vstup, jež může ovlivňovat komunikaci aplikace s databází.

³³ Odstranění škodlivého kódu za pomoci různých detekčních a validačních technik.

3 Příprava na realizaci webové galerie

Před samotným započítím prací na galerii bylo nutné si shrnout, jak by v základu měla galerie vypadat a fungovat, a také které nástroje při její tvorbě využiji.

3.1 Inspirace

V průběhu několika posledních let se můj pohled na grafiku a uživatelská rozhraní velice změnil. Čím více čtu o principech dobrého a esteticky příjemného UI, tím více docházím k názoru, že minimalismus je cestou k úspěchu. Tedy alespoň u aplikací, ve kterých je obsah na prvním místě.

Uživatel by neměl být rozptylován zbytečnými rozměry designéra. V tomto ohledu je dobré se řídit pravidlem - funkce převyšuje formu. Pokud nedodržujeme základy typografie hovořící o správném kontrastu písma a pozadí, ideálním řádkování atp., popíráme tak důležitost obsahu webové stránky. Pravdou zůstává, že ty nejintuitivnější grafická rozhraní ani nepotřebují rozsáhlou dokumentaci. Uživatel je schopen se v nich pohybovat bez velkých problémů a zvládne-li ovládat malou část systému, přijde mu ovládání dalších částí jasné a dobře čitelné.

3.1.1 Švýcarský grafický design a Bauhaus

Při tvorbě uživatelského rozhraní galerie mě ovlivnil především švýcarský grafický design, jenž se během posledních let stal známým pojmem. V angličtině je též nazývaný jako *International Typographic Style*.

Richard Hollis v úvodu své knihy *Swiss Graphic Design* popisuje sociální a kulturní důvody, proč se právě Švýcarsko stalo na mnoho let velmocí v oblasti grafického designu. Podle Hollise jsou Švýcaři národ uznávající lidskou tvořivost a řemeslnou zručnost. Švýcaři jsou známí svým zájmem o obrazovou kulturu, a tak i sebemenší oblast ve Švýcarsku má svoji vlajku na důkaz své identity. Smysl pro detail, inovativní práce s typografií a vysoké tiskařské standardy se promítly nejen v technických a reklamních publikacích velkých průmyslových korporací, ale také v tištěných

prezentacích menších firem (brožury cestovních kanceláří atp.). Vrchol švýcarského grafického designu nastává v padesátých a šedesátých letech 20. století.³⁴

Abych se dozvěděl něco více o švýcarské grafice a jejích představitelích, musím proniknout ještě o pár desítek let dříve – přesněji řečeno do začátku 20. století a vzniku německé umělecké školy Bauhaus. Bauhaus patří doteď k nejuznávanějším a hlavně nejvlivnějším uměleckým školám, které kdy fungovaly. Bauhaus postavil základy moderní architektury, grafiky a typografie. Sdružoval umělce různých národností a stylů, umožňoval jim tvořit, diskutovat a vystavovat. Mezi učitele, jenž na Bauhausu působili, patří taková jména jako Paul Klee, Wassily Kandinsky, či Johannes Itten. Právě poslední zmíněný autor je proslulý svými typografickými experimenty. Svoji prací bořil tehdejší grafické konvence a otevřel nový prostor pro grafické vyjádření. Na Ittena navázal neméně známý maďarský malíř László Moholy-Nagy, který přišel s konceptem *typofoto*, spojujícím text a fotografie v jednom prostoru. V té době studovalo na Bauhausu také několik švýcarských studentů. Dva z nich – Max Bill a Theo Ballmer se po studiích vrátili domů do rodné země, kde jejich vliv přetrvává dodnes. Dalším významným švýcarským grafikem byl Sigfried Giedion, který i po zániku Bauhausu v roce 1930 udržoval kontakt s významnými jeho osobnostmi (včetně zakladatele Waltera Gropia) a také spolupracoval s dalšími grafiky, jenž studovali na této škole.³⁵

Švýcarský grafický design je pro mne hlavním proudem inspirace při tvorbě jakékoli grafiky. Vycházím z jejich minimalistických postupů a snažím se udržet design čistý, jednoduchý a upřímný bez jakýchkoli nadbytečných prvků. Grafický design jako užité umění má za úkol předávat informaci v esteticky zajímavé podobě, ale z mého pohledu rozhodně ne za cenu špatného vyjádření původního sdělení. Pak veškerá grafická práce ztrácí význam.

3.1.2 Metro UI

Designový jazyk Metro je název pro UI vytvořené firmou Microsoft původně navržené pro operační systém Windows Phone. Klíčový princip Metra spočívá v soustředění se více na typografii uživatelského rozhraní než na samotnou grafiku.

³⁴ HOLLIS, Richard. *Swiss graphic design: the origins and growth of an international style, 1920-1965*. New Haven: Yale University Press, 2006, 272 s. ISBN 03-001-0676-9. s 9.

³⁵ HOLLIS, Richard. *Swiss graphic design: the origins and growth of an international style, 1920-1965*. New Haven: Yale University Press, 2006, 272 s. ISBN 03-001-0676-9. s 24.

Designový tým Microsoftu uvádí jako zdroj své inspirace značení pro cestující ve veřejné dopravě, např. to které se dá nalézt v *King County Metro*.³⁶ Metro sází na jednobarevné barevné plochy a piktogramy. Pro Metro bylo speciálně vytvořeno bezpatkové písmo Segoe, které je v rozhraní zastoupeno v mnoha různých velikostech a řezech. Působí jednoduchým, moderním dojmem, ale z mého pohledu se příliš nehodí na dlouhé odstavce textu v malé velikosti, protože jeho čtivost v řádcích není příliš dobrá. Avšak použité jako velké titulky či popisky vypadá skvěle.

Podle mého názoru je operační systém Windows Phone díky použití Metra zatím nejlépe esteticky vypadající mobilní platformou a udává směr, kterým by se mohly v budoucnu vydat i jiné mobilní operační systémy. Kontrastní typografie a velké barevné plochy napomáhají lepšímu pohybu celým systémem, který jistě ocení i lidé s horším zrakem.

Má inspirace Metrem je nejvíce znát na uživatelském rozhraní administrace galerie, kde kombinují výraznou typografii a barevné plochy.

3.1.3 Cargo Collective

Webová služba s názvem Cargo Collective³⁷ je jedním z nemnoha opravdu kvalitních projektů pro jednoduchou a rychlou tvorbu uměleckého portfolia. Zatímco jiné služby nedovolují obsahu portfolia příliš vyniknout z důvodu své těžkopádnosti a přehlcenosti, Cargo Collective vytváří iluzi skoro neviditelné webové stránky, kde je veškerá pozornost soustředěna na umělecký obsah. Tvůrci přišli s myšlenkou, jež je mi velice blízká – začít s málem a postupně přidávat jen funkce, které uživatelé opravdu ocení. Kvalita obsahu stránek vytvořených na platformě Cargo Collectivu je vysoká, jelikož registrace je možná pouze s pozvánkou od jiného uživatele. Každý nový uživatel má k dispozici 5 pozvánek, které může rozeslat komukoli, o kom si myslí, že by si zasloužil mít portfolio na CC.

Registrovaný uživatel Cargo Collectivu má v tuto chvíli možnost vybírat z 32 různých vzhledů portfolia. Tyto šablony jdou ještě dále upravovat za pomoci vlastního CSS kódu. Neplacená verze účtu na CC je omezená počtem projektů v portfoliu (max. 12).

³⁶ Společnost poskytující služby městské hromadné dopravy ve státě Washington, USA.

³⁷ Cargo Collective [online]. [cit. 2013-07-01]. Dostupné z: <http://cargocollective.com/>

Po upgradu³⁸ na placenou verzi účtu (9 dolarů měsíčně) je prostor pro prezentaci neomezený.

Cargo Collective mě zaujal zejména svými krásnými šablonami. Přestože se na první pohled jedna od druhé příliš neliší, každá obsahuje pár detailů, které ji činí originální. CC také využívá technologie AJAX pro načítání pouze nutného obsahu při přechodu mezi jednotlivými stránkami portfolia. Díky tomu funguje vše rychle, což se hodí např. při procházení většího počtu obrázků v portfoliu.

3.2 Galerie jiných uměleckých škol

V prvních dnech tvorby galerie jsem začal s prozkoumáváním internetu a hledáním systémů, které používají jiné školy k prezentaci děl svých studentů. K mému úžasu jsem přišel na jeden zajímavý fakt a to, že žádná z českých uměleckých škol nemá jednotnou webovou galerii ve smyslu místa pro prezentaci jednotlivých děl, jež studenti při svém studiu vytvořili. Buď mají galerie sloužící k prezentaci celé školy – najdeme v nich jen fotografie z pořádaných akcí, anebo nemají galerii vůbec žádnou. Nejspíš je to způsobeno tím, že kdo chce (ze studentů), může své práce na internetu vystavovat z nějaké vlastní iniciativy, pod vlastní adresou. Já s tímto přístupem nesouhlasím, umělecké školy by se měly chtít prezentovat i konkrétními uměleckými díly, které na nich vznikaly. Můžou sloužit jako ukázka toho, kam škola své studenty vede a co jim umožňuje.

Mně osobně se docela líbí zpracování webových stránek Katedry výtvarné výchovy na Masarykově univerzitě v Brně. Úvodní stránka zobrazuje aktuality a také fotografie z výstav katedry a nedávno se na jejich webu objevila nová sekce – e-galerie. Zatím v ní nalezneme pouze jedno album, ale v budoucnu bude jistě zastřešovat veškeré fotografie, které souvisí s činností této katedry.

Katedra výtvarné výchovy na Palackého univerzitě v Olomouci má také svou galerii, dokonce rozdělenou do sekcí podle oblasti tvorby. Bohužel v každé z této sekcí nalezneme maximálně deset fotografií uměleckých děl a některé z nich nejsou dokonce ani označené jménem autora.

³⁸ Přechod ze starší / nižší verze programu na novější / vyšší.

Katedra výtvarné výchovy na Karlově univerzitě v Praze využívá služby rajce.idnes.cz, ale zas a znovu jsou na fotografiích zachyceny jen momentky z pořádaných workshopů a jiných akcí.

Vysoká škola umělecko-průmyslová v Praze má webovou prezentaci, která je asi nejbližší mému pohledu na webdesign. Minimalistické pojetí celého webu VŠUP dává krásně vyniknout obsahu a vzhledem k tomu, že studentů na této škole není zase tolik, můžou si dovolit každému vytvořit samostatný profil s jeho fotkou. Ne všechny profily obsahují fotogalerii studentových prací, ale i tak možnost prohlédnout díla některých studentů na jednom konkrétním místě. Chybí však možnost plynulého procházení všech děl studentů a také možnost rychle filtrovat díla podle oborů.

Situace v zahraničí je trošku odlišná. Pro příklad uvedu web Art Center College of Design v Pasadeně, která má opravdu rozsáhlou galerii s fotografiemi tvorby studentů. Galerie je rozdělena do sekcí podle studijního programu a každé dílo v galerii je označeno jménem autora, vedoucím ateliéru a dalšími informacemi.

Většina z výše uvedených galerií je zjevně integrovaná do systému psanému přímo pro správu celého webu školy / katedry. Tyto systémy nejsou veřejně k dispozici, a tak bližší seznámení s nimi není zcela možné.

3.3 Použité technologie a nástroje

V prvních dnech vývoje aplikace jsem uvažoval nad využitím jednoho z open-source CMS systému a jeho následné úpravy pro mé potřeby. Nejvíce zkušeností mám se systémem Wordpress, který dříve sloužil jako blogovací platforma, ale díky svým možnostem rozšíření pomocí zásuvných modulů je využíván také pro správu nejrůznějších portfolií, galerií i komerčních webů. Dalším systémem pro správu obsahu je *Drupal*, ale ten jsem zavrhl v podstatě hned ze začátku. Přestože se jedná o systém velice flexibilní, jeho úpravy a konfigurace utápějí programátora v moři nepříliš dobře napsaného kódu.

Oba zmíněné systémy však obsahují spoustu nepotřebných funkcí, které by chod galerie zbytečně zpomalovaly, a tak jsem se nakonec rozhodl jít cestou kompletně vlastního systému, přestože mi bylo jasné, že půjde o práci na několik stovek hodin, což se ve výsledku ukázalo jako reálné číslo.

3.3.1 XAMPP

Zkratka XAMPP označuje balík předkonfigurovaného webového serveru obsahující Apache, MySQL, PHP a Perl. Instalace tohoto balíku probíhá jako instalace každého jiného softwaru pro Windows – stačí spustit stažený exe soubor a projít instalačním průvodcem.

Výhodou instalace webserveru přímo na lokálním počítači je možnost spouštět webové aplikace přímo ze složky na disku bez nutnosti uploadovat soubory se zdrojovým kódem přes FTP³⁹ na vzdálený server a až poté webovou aplikaci testovat.

3.3.2 jQuery

Ačkoliv existuje mnoho frameworků pro Javascript, jeden z nich se stal takřka standartem, jež využívá převážná většina moderních webových stránek. Jmenuje se jQuery a těžko budeme hledat lepší a především podporovanější řešení při psaní Javascript kódu. S jQuery mám několikaletou zkušenost a musím říci, že zkratka nemá v této oblasti konkurenci. Existují k němu tisíce různých pluginů⁴⁰, tipů a návodů. Nedávno vydaná verze jQuery 2.0 opouští podporu starých prohlížečů (jako je Internet Explorer 8 a nižší), což považuji za logický krok ve snaze o rozšíření moderních prohlížečů do celého uživatelského spektra.

Sám jsem během posledních dvou let naprogramoval několik rozšíření pro jQuery, které dále využívám při své tvorbě. Všechny použité pluginy jsou vydané pod MIT nebo GNU-GPL licencí.

Seznam jQuery pluginů používaných ve webové galerii:

- *Editorial* – jednoduchý vizuální editor textu (vlastní)
- *FileUpload* – umožňuje asynchronní nahrávání souborů na server
- *Hotkeys* – umožňuje jednoduchou integraci klávesových zkratk
- *HTML5 Shiv* – přidává podporu nových HTML5 tagů do starších prohlížečů
- *Incredible* – rozšiřuje možnosti HTML tabulek (vlastní)
- *Jupinputs* – rozšiřuje možnosti zaškrtávacích a výběrových polí formulářů (vlastní)
- *Isotope* – vytváří vertikální dynamické zarovnání HTML elementů

³⁹ File Transfer Protocol – protokol pro přenos souborů z lokálního počítače na vzdálený server nebo obráceně.

⁴⁰ Doplnkový modul rozšiřující funkčnost aplikace. V tomto konkrétním případě modul rozšiřující možnosti jQuery frameworku.

- *PJAX* – slouží k načítání částí stránek za pomoci technologie AJAX
- *Qinema* – slouží k vytváření obrázkových slideshow (vlastní)
- *Stupidtable* – umožňuje seřazování dat v tabulkách
- *Validate* – validace formulářů

3.3.3 LESS

Ve chvíli, kdy píšou dlouhý CSS kód, rád využívám tzv. CSS preprocessoru⁴¹. Koncepce jednoduchého popisu v CSS kódu s sebou totiž nese i mnoho nevýhod, které se preprocesory snaží překonat.

LESS (zkratka pro Leaner CSS) je knihovna zavádějící do jazyka CSS vlastnosti jako jsou proměnné a makra, napomáhající lepší údržbě kódu. Výhodu proměnných uvedu na následujícím příkladu: Řekněme, že máme v CSS definovanou vlastnost `color`, která nabývá stejné hodnoty u více tříd. Jestliže chceme tuto hodnotu změnit u všech tříd v CSS kódu, musíme jí ručně přepsat všude, kde se vyskytuje. Zatímco v LESS kódu využijeme možnosti vkládání proměnných a hodnotu vlastnosti `color` přiřadíme k proměnné. Proměnnou pak použijeme kdekoli, kde chceme mít ve vlastnosti `color` uvedenou hodnotu. Ve chvíli, kdy chceme hodnotu změnit, stačí ji přepsat v definici proměnné a po kompilaci LESS kódu se hodnota dosadí namísto té předchozí všude, kde jsme proměnnou použili.⁴²

Prohlížeč sám o sobě není schopen zápis v LESS formátu zpracovat, a tak je nutné ho za pomoci LESS kompilátoru⁴³ převést na běžné CSS. Zní to složitě, ale vše se dá dobře zautomatizovat, takže nemusíme po každé úpravě LESS souboru ručně spouštět kompilaci do CSS.

3.3.4 PHP a MySQL

Pro naprogramování server-side řešení webové galerie jsem si zvolil programátory oblíbenou kombinací dvou technologií – skriptovacího jazyka PHP a databáze MySQL. Ačkoliv vím, že PHP už dávno není nejmodernější jazyk, jeho stabilita a velká podpora ze strany komunity je pro mne pořád hodně důležitým aspektem při výběru vhodných technologií. PHP znám ze skriptovacích server-side jazyků nejlépe a vzhledem k tomu, že mám i spoustu hotového kódu, který se při programování hodil, byla to správná volba.

⁴¹ Program sloužící ke zpracování vstupu a následné produkci výstupu, který se používá jako vstup v jiném programu.

⁴² Zdroják.cz. In: *LESS: stejné CSS za méně peněz* [online]. 2010 [cit. 2013-07-01]. Dostupné z: <http://www.zdrojak.cz/clanky/less-stejne-css-za-mene-penez/>

⁴³ Překladač zapsaného kódu do finální podoby / formátu.

Samotné PHP by však nestačilo, věděl jsem, že potřebuji použít i nějaký jednoduchý framework pro urychlení práce. Po mnoha hodinách porovnávání nejpoužívanějších PHP frameworků jsem zvolil Slim Framework. Jedná se o jeden z těch jednodušších, ale pokrývá všechny základní funkce, které jsem potřeboval.

Díky oblíbenosti PHP není problém nalézt další samostatné open-source knihovny se spoustou užitečného kódu. Upřímně řečeno, tyto knihovny bývají častokrát na takové úrovni, že bych těžko dosáhl lepšího výsledku, kdybych se pokoušel obdobný kód napsat sám. Samozřejmě kvalitní knihovna musí být i dobře zdokumentovaná, aby se v ní vyznal kdokoli, kdo ji chce integrovat do své aplikace.

Seznam PHP knihoven používaných ve webové galerii:

- *FileUploader* – zjednodušuje práci s nahráváním obrázků
- *Idiorm* – zjednodušuje práci s databází MySQL
- *LESS* – umožňuje kompilovat LESS kód do CSS
- *Liquid* – rychlá a jednoduchá šablonovací knihovna
- *Password* – knihovna sloužící k šifrování hesel
- *SimpleImage* – umožňuje manipulovat s nahranými obrázky
- *Valitron* – slouží k validaci obsahu formulářů

3.3.5 Github

Internetová služba Github patří mezi nejlepší systémy pro údržbu, verzování a také zálohování kódu. Github je postaven na systému Git – velmi rozšířený verzovací systém používaný mnoha open-source projekty.

Systém Git nám umožňuje porovnávat a sledovat jednotlivé změny, které jsme v kódu udělali. Díky tzv. větvím můžeme pracovat na jedné vývojové linii kódu, aniž bychom nevratně zasahovali do otestovaných částí programu. Ve chvíli, kdy máme nový zdrojový kód otestovaný, můžeme ho pomocí Gitu sloučit s tím původním.

Služba Github implementuje systém Git do podoby webové aplikace a hotového řešení pro vývojáře na volné noze i softwarové firmy. Výhodou je možnost okamžité synchronizace lokálního adresáře s repositářem na serveru Githubu, která zajišťuje průběžné zálohování zdrojového kódu.

3.3.6 Sublime Text editor

Při programování webové galerie jsem používal jen a pouze textový editor Sublime Text. Pro mě je to číslo jedna mezi editory pro Windows, jelikož nabízí mnoho funkcí zabalených do pěkného a jednotného rozhraní. V základní konfiguraci si rozumí s velkým množstvím programovacích jazyků a je snadno rozšiřitelný pluginy. Jednou ze zajímavých funkcí Sublime Textu je malý náhled na celkový kód otevřeného souboru v pravé části editoru. Tento náhled pomáhá v orientaci v souboru a umožňuje rychlé přesouvání ve zdrojovém kódu nahoru a dolů bez nutnosti použití posuvníku.

Za dobu, co se zabývám programováním webových aplikací, jsem vyzkoušel nespočet textových editorů. Některé mi vyhovovaly méně, protože nenabízely příliš mnoho možností pro urychlení práce a na druhé straně několik z nich bylo přeplněných funkcemi. Upřímně řečeno, raději vždy volím minimalističtější řešení, protože stejně veškerý kód píše ručně. Samozřejmě jsem měl možnost pracovat také s Adobe Dreamweaverem, který mnozí považují za jeden z nejpropracovanějších editorů. Z mého pohledu se jedná o slušné řešení, ale rozhodně není nad to, mít nad svým zdrojovým kódem plnou kontrolu, na což teoreticky stačí i běžný poznámkový blok ve Windows. Rozhodně nechci říkat, že je Dreamweaver špatný, ale spíše zbytečně komplikovaný.

4 Realizace webové galerie

V této kapitole se snažím konkrétněji popsat samotný průběh tvorby galerie. Galerie má dvě základní rozhraní: administraci a samotnou galerii. Administrace slouží ke spravování obsahu galerie a mají k ní přístup pouze registrovaní správci.

Administrace umožňuje správci přidávat nové projekty, kolekce a stránky. Pro každou tuto činnost je v administraci jedna sekce.

Projektem se rozumí jedno autorské dílo (nebo cyklus) a každý projekt může obsahovat několik obrázků, ať už kvůli více pohledům na jedno dílo (např. objekt, instalace) nebo jako galerie fotografií jednoho autorova výtvarného cyklu. Každý projekt může správce vložit do kolekce označující techniku díla respektive typ díla (malba, kresba, fotografie atp.). Správce může tyto kolekce libovolně přidávat a odebírat. Projekty jsou v administraci nazvány jako „*Díla*“.

Galerie obsahuje i statické stránky, které popisují např. informace o galerii, či kontakt na KVK. Přestože je také možné stránky libovolně přidávat a odebírat, nepočítám s tím, že by této funkce správci příliš využívali. Obecně si myslím, že stačí jen výše jmenované dvě, ale určitě je dobré dát správcům možnost je upravovat stejně pohodlně, jako zmíněné projekty.

Poslední sekcí administrace je nastavení uživatelského účtu správce. Umožňuje změnu zobrazovaného jména profilu a hesla.

4.1 Adresářová, souborová a databázová struktura

Jako první jsem si vytvořil přibližnou adresářovou strukturu celého projektu. Je dobré si předem rozvrhnout, do jakých složek rozdělíme jednotlivé soubory – zejména kvůli přehlednosti celé práce. Mně se osvědčilo rozdělit si strukturu aplikace na několik základních složek podle typů souborů, které jsou v nich uloženy. To znamená, že odděluji funkční část aplikace (PHP soubory) od souborů s HTML šablonami, CSS styly a Javascripty. Vlastně asi neexistuje žádné zcela univerzální pravidlo, kterým je dobré se při strukturování adresářů a souborů řídit, já sám strukturu často doladuji ještě při vývoji aplikace.

Adresáře a soubory pojmenovávám anglicky, stejně tak všechny proměnné, funkce a třídy ve zdrojovém kódu. Pomáhá mi to udržet celistvost celé aplikace z hlediska formální úpravy kódu. Veškeré nativní PHP funkce mají anglické názvy, takže mi použití angličtiny v mém kódu přijde jako logický krok. Vlastně jsem na to zvyklý, takže nad tím už ani nepřemýšlím.

Po tom, co jsem skončil se strukturováním souborů a adresářů, přišla na řadu databáze. V databázi jsou definované tabulky. Tabulka slouží pro uložení mnoha záznamů jednoho typu. Typem jsou v tomto případě správcové, stránky, projekty, kolekce a nahraná média (obrázky). Každý typ je tak v databázi reprezentován jednou samostatnou tabulkou. Strukturu tabulky vytváří tzv. sloupce. Sloupec slouží k uložení jedné určité hodnoty. Například tabulka *users* (uživatelé = správcové) má sloupce: *id*, *username*, *password*, *ip_address*, *last_login* atd. Jednotlivé tabulky mají rozdílnou strukturu sloupců, poněvadž musí vyhovovat danému typu záznamu.

4.2 MVC architektura

Architektura MVC dělí aplikaci na 3 logické části tak, aby je šlo upravovat samostatně a dopad změn byl na ostatní části co nejmenší. Tyto tři části jsou Model, View a Controller. Model reprezentuje data a business logiku aplikace, View zobrazuje uživatelské rozhraní a Controller má na starosti tok událostí v aplikaci a obecně aplikační logiku. S architekturou MVC jsem se poprvé setkal při svých začátcích programování ve frameworku Ruby on Rails a od té doby je pro mě asi nejpřehlednějším řešením, jak organizovat samotný zdrojový kód aplikace.⁴⁴

V praxi vše funguje následovně:

1. Uživatel zadá internetovou adresu webové galerie do prohlížeče.
2. Aplikaci webové galerie přijde HTTP dotaz na požadovanou stránku.
3. Aplikace vyhodnotí HTTP dotaz a spustí funkci v Controlleru, která přísluší dané adrese. V mém případě chce uživatel zobrazit indexovou stránku.
4. Funkce příslušící v Controlleru indexové stránce, použije funkci v Modelu pro získání dat se všemi uloženými projekty⁴⁵ v databázi.

⁴⁴ Zdroják.cz. In: *Úvod do architektury MVC* [online]. 2009 [cit. 2013-07-01]. Dostupné z: <http://www.zdrojak.cz/clanky/uvod-do-architektury-mvc>

⁴⁵ Díla autorů.

5. Po získání dat z databáze načte Controller konkrétní soubor View obsahující šablonu pro indexovou stránku, vepíše do ní data a zobrazí ji.

Na této bázi funguje víceméně celá galerie. Právě zde se ukazuje vysoká modularita MVC architektury. V podstatě stačí naprogramovat část systému, která je schopná přidávat, odebrat a upravovat záznamy z databáze a pak tuto část aplikovat na všechny typy spravovaného obsahu – projekty, kolekce i statické stránky. Především v administraci mi MVC architektura ušetřila mnoho času.

4.3 Uživatelské rozhraní webové galerie

Jak jsem již psal v kapitole o inspiraci, při tvorbě grafiky a grafických rozhraní se snažím opírat o minimalismus, typografii a intuitivnost. Nejinak tomu bylo při návrhu uživatelského rozhraní jak samotné galerie, tak i administrace.

Mnozí by čekali, že práce na uživatelském rozhraní začínají v programu Adobe Photoshop nebo jiném grafickém editoru. Několik let jsem skutečně své návrhy webových stránek ve Photoshopu tvořil, ale dnešní realita je jiná. Abych řekl pravdu, nikdy mě nebavilo pracně navrhovat GUI nejdříve v grafickém editoru a následně vše znovu přetvářet ve funkční HTML a CSS stránku. Samozřejmě při návrh logotypu a jiných bitmapových prvků webových stránek se grafickému editoru nevyhneme, ale pro vizualizaci kompletního webu už ho zkrátka nepotřebuji. Abych si dokázal představit, jak bude aplikace vypadat a fungovat, vytvářím si tzv. wireframy. Slovo wireframe, v kontextu předběžné skici uživatelského rozhraní, se nepoužívá jen v angličtině. Běžně se s ním setkávám v hovorové mluvě českých programátorů, protože je těžké najít vhodný český ekvivalent. V příloze je možné pro lepší představu shlédnout základní (původně vektorové) wireframy zobrazující základní rozvržení administrace a veřejně viditelné části webové galerie (viz. Obrazová příloha strana I., obrázek č.1 a č.2).

4.3.1 Administrace

Vzhled administrace je postaven na bezpatkovém písmu Open Sans, které nabízí Google ve své databázi fontů. Open Sans je v dnešní době používáno mnoha webovými stránkami a osvědčilo se ve všech dostupných řezech. Chvíli mi trvalo, než jsem našel ideální velikosti jednotlivých nadpisů tak, aby podtrhovaly jejich důležitost na stránce. Základní velikost písma v administraci je 16px. Ačkoliv se to zdá přehnaně moc, myslím,

že v kontextu administrace funguje tato velikost dobře. Jednotlivé sekce jsou vyznačeny ikonami v levém navigačním menu. Navigační menu zůstává při skrolování fixované v okně prohlížeče, což zrychluje pohyb mezi jednotlivými sekcemi administrace. Použité ikony jsou součástí fontu FontAwesome. Využití ikonového fontu namísto bitmapové grafiky je z hlediska příchodu displejů s vysokým rozlišením nezbytné. Bitmapy v běžném rozlišení vypadají na displejích s vysokým rozlišením rozmazaně a nekvalitně. Ikonový font se na stránkách zobrazuje jako vektor, tudíž nedochází ke ztrátě kvality ani na těch nejjemnějších displejích.

Černobílé schéma administrace doplňuje jeden odstín středně syté modré barvy v kombinaci s bílým textem pro oživení a zvýraznění některých prvků. Kromě modré jsem vyzkoušel i jiné barvy, ale žádná z nich nefungovala v rozhraní ani z poloviny tak dobře (viz. Obrazová příloha strana II. a III., obrázek č.4 až č.6).

Administrace využívá technologie AJAX pro navigaci mezi jednotlivými stránkami, tudíž nedochází k opětovnému načítání celého okna prohlížeči při každém kliknutí na odkaz či odeslání formuláře.

Pro zobrazení tabulky se seznamem záznamů každého typu spravovaného obsahu jsem vytvořil plugin pro jQuery, díky němuž se dají za použití klávesových zkratk vybrat označit a odznačit (Ctrl+A a Ctrl+D) všechny záznamy a ty následně hromadně mazat (klávesa Del) nebo obnovovat (klávesa Ins). Tabulka také umožňuje řazení dat podle jednotlivých sloupců.

Abych zachoval konzistentnost všech ovládacích prvků v administraci, musel jsem vytvořit jQuery plugin, který nahradí stávající zaškrťovací pole a výběrová pole vlastními. Je to z toho důvodu, že zaškrťovací pole a výběrová pole nejdou stylovat za pomoci CSS. V každém prohlížeči vypadají trochu jinak a do vzhledu mé administrace zkrátka nepasovaly.

Při editaci projektu je možné nahrávat fotky přímo z disku počítače. Nahrávání též využívá technologie AJAX, a tak je velice plynulé a uživatelsky přívětivé. Malé čtvercové náhledy na fotky se zobrazují v pravé části editačního okna a můžeme měnit jejich pořadí pouhým kliknutím na ně a tažením nahoru či dolů. Obrázek, který je první v pořadí, se zobrazuje jako hlavní náhled na projekt (na úvodní stránce galerie).

Po integraci funkcí sloužící k nahrávání fotek přišlo na řadu přidání možnosti vkládat projekty do kolekcí. K tomu slouží jednoduchý seznam kolekcí zobrazený vpravo pod volbou pro nahrávání fotek. Kolekce mají svá zaškrťovací pole, která při zaškrtnutí vloží projekt do dané kolekce. Vyřešení situace, kdy jeden projekt má více kolekcí a každá kolekce má více projektů, si vyžádalo použití tzv. relační tabulky v databázi. V ní je vždy uložen záznam s id projektu a id kolekce, do které je vložen. Každý jeden projekt má tak v relační tabulce tolik záznamů, v kolika kolekcích se nachází.

4.3.2 Frontend

Grafické rozhraní samotné galerie (dále nazývané jako frontend⁴⁶ aplikace) viditelné pro každého návštěvníka má podobné rozvržení a barevné schéma jako administrace. Nalevo je umístěná fixní navigace, jež při skrolování zůstává na stejném místě v okně prohlížeče.

V prvních pár dnech tvorby jsem si pohrával s myšlenkou použít ve frontendu tmavé pozadí a světlý text. Takové schéma je totiž příjemné na oči zejména v pozdních večerních hodinách a vypadá skvěle v kombinaci s černobílými fotkami. Nakonec jsem se však rozhodl zůstat u klasického bílého pozadí a černého (respektive tmavě šedého) textu. Na bílém pozadí lépe vynikne většina typů grafického obsahu a na rozdíl od černého pozadí z něj návštěvník nemá pocit jisté stíněnosti (viz. Obrazová příloha strana IV., obrázek č.7 a č.8).

Na úvodní stránce galerie se zobrazují náhledy všech publikovaných projektů (řazeno časově od nejnovějšího). Při skrolování se automaticky načítají projekty nové, aby bylo prohlížení galerie co nejplynulejší. Galerie umožňuje buď prohlížení projektů na úvodní stránce nebo rozkliknutí detailu projektu a následný přechod na předchozí či následující projekt. V detailu projektu jsou zobrazeny všechny jeho nahrané fotografie včetně stručného popisu, autora projektu a roku dokončení.

Kromě úvodní stránky nalezneme v galerii již zmíněné statické stránky, které slouží k zobrazení základních informací o webové aplikaci a kontaktech na KVK. Všechny statické stránky jsou v době dokončování prací vyplněny pouze ilustračním obsahem. Jestliže bude galerie spuštěna oficiálně, texty se upraví do potřebné podoby.

⁴⁶ Adaptic.cz. *Frontend* [online]. 2013 [cit. 2013-07-01]. Dostupné z: <<http://www.adaptic.cz/znalosti/slovnicek/frontend>>

Webovou galerii jsem otestoval ve všech moderních prohlížečích, jako jsou Google Chrome, Firefox (od verze 4), Safari 5, IE9 a IE10. Ve starších prohlížečích (zejména IE8 a IE7) by mělo vše fungovat v rámci možností správně, ale jejich používání nedoporučuji, jelikož nepodporují moderní standardy a jsou špatně zabezpečené vůči potenciálním útokům hackerů.

5 Závěr

Naprogramovat kompletní plně funkční webovou aplikaci a navrhnout k ní vhodné uživatelské rozhraní je většinou práce pro více než jednoho člověka. Na druhou stranu to má výhodu v tom, že můžete do aplikace promítnout jen a pouze svoji vizi, která se pak odráží na všech jejích funkcích a vlastnostech. Upřímně řečeno, při programování jsem si raději veškerou práci vždy odvedl sám, jelikož v této oblasti mám jasnou představu o tom, jak by věci měly fungovat a velice nerad se přizpůsobuji rozdílným názorům.

Přestože jsem si uvědomoval, do čeho se pouštím, nakonec se samozřejmě všechno o mnoho desítek hodin protáhlo. Především z toho důvodu, že se člověk zastaví na určitém problému a jeho řešení zabere klidně i jeden celý den. Takových situací mě potkalo několik, ale i tak pro mě programování většinu času bavilo. Je skvělé vidět, jak vznikají pod rukama věci, které ještě před chvílí byly jen v mé hlavě. Při programování i menších aplikací je potřeba myslet hodně dopředu a připravit si systém tak, aby byl snadno rozšiřitelný a udržovatelný. Někdy jsem se dostal do fáze, kdy mi přišlo, že zbytečně komplikuji celý kód, kvůli funkcím, které možná ani nebudu implementovat. Nakonec se však ukázalo, že se soustředění i na ty nejmenší detaily vyplácí, protože mi v minulosti tyto detaily ušetřili mnoho času.

Chtěl bych touto webovou galerií otevřít prostor pro nový pohled studentů uměleckých oborů na programování jako takové. Díky spoustě zdrojů bezplatných internetových zdrojů se může naučit základy naprosto každý. Zním pár grafiků, kteří tvrdí, že ve chvíli, kdy byli schopni alespoň trochu porozumět práci programátorů, se jejich pohled na navrhování uživatelských rozhraní částečně změnil. Budoucnost vidím v protnutí umění a technologií, o kterém mluvil i Steve Jobs ve své autobiografii.⁴⁷

Pokaždé, když testuji funkčnost nějaké své aplikace, vidím, co by se ještě mohlo dodělat a práce pro mě vlastně nikdy nekončí. Přesně takový pocit mám také z této webové galerie. Ačkoliv v základu splňuje cíle, které jsem si na začátku stanovil, rád bych v jejím vývoji nadále pokračoval, pokud o to bude mít KVK zájem. Byl bych rád, kdyby se galerie stala v budoucnu oficiálním místem prezentace prací studentů naší katedry.

⁴⁷ ISAACSON, Walter. *Steve Jobs*. Vyd. 1. Praha: Práh, 2011, 678 s., [16] s. obr. příl. ISBN 978-80-7252-352-8.

6 Resume

I have created a basic, but fully functional web gallery for the student's artwork. The original idea was to come up with a solution of a single space, where students could present what they draw, paint and print during their studies in our department.

The web gallery application was developed by using the PHP scripting language and the MySQL database on the server-side. The client-side of this app is standard combination of HTML templates, CSS stylesheets (compiled from LESS code) and Javascript code.

I have been trying to create an invisible and simple-to-use web gallery with a focus on the photographic content. My inspiration came mainly from the International Typographic Style, which is a minimalist point of view on a graphic design established especially by Swiss graphic designers. I was also inspired by a portfolio creation application named *Cargo Collective*. I find this application very useful and good looking.

I went through all the phases of the web app development – from some simple sketches on a piece of paper to the functional app. There are many aspects of a good and stable application and it was really hard to find them all. I hope I largely succeeded and the gallery will be officially used by our department in the near future.

Seznam použité literatury

FISHEL, Catharine. *Minimal graphics: the powerful new look of graphic design*. Gloucester, Mass: Rockport, 2002, 192 s. ISBN 978-156-4968-449.

FRANZ, Laura. *Typographic web design: how to think like a typographer in HTML and CSS*. Chichester, West Sussex: Wiley, 2011, 336 s. ISBN 11-199-7687-1.

GASSTON, Peter. *The modern Web: multi-device Web development with HTML5, CSS3, and JavaScript*. San Francisco: No Starch Press, 2013, 264 s. ISBN 15-932-7487-4.

HARTSON, Rex a Pardha PYLA. *The UX Book: process and guidelines for ensuring a quality user experience*. Boston: Elsevier, 2012, 937 s. ISBN 01-238-5241-2.

HOLLIS, Richard. *Swiss graphic design: the origins and growth of an international style, 1920-1965*. New Haven: Yale University Press, 2006, 272 s. ISBN 03-001-0676-9.

HORÁČEK, Radek. *Galerijní animace a zprostředkování umění: Poslání, možnosti a podoby seznamování veřejnosti se soudobým výtvarným uměním prostřednictvím aktivizujících programů na výstavách*. 1. vyd. Brno: CERM, 1998, 139 s. ISBN 80-720-4084-7.

ISAACSON, Walter. *Steve Jobs*. Vyd. 1. Praha: Práh, 2011, 678 s., [16] s. obr. příl. ISBN 978-80-7252-352-8.

MEJZR, Martin. *Bezpečnostní problémy internetových aplikací*. Praha, 2008. Dostupné z: https://dip.felk.cvut.cz/browse/pdfcache/mejzrm1_2008bach.pdf. Bakalářská práce.

ČVUT, FEL, Katedra počítačů. Vedoucí práce Ing. Zdeněk Troníček Ph.D.

OPPLIGER, Rolf. *SSL and TLS: theory and practice*. Boston: Artech House, 2009, 257 s. Artech House information security and privacy series. ISBN 15-969-3447-6.

PILGRIM, Mark. *HTML 5: Up and Running*. California, USA: O'Reilly Media, 2010. 222 s. 1. ISBN 978-0596806026.

THOMPSON, John B. *Média a modernita: sociální teorie médií*. 1. české vyd. Praha: Karolinum, 2004, 219 s. ISBN 80-246-0652-6.

VANČÁT, Jaroslav. *Vývoj obrazivosti od objektu k interaktivitě: gnozeologické předpoklady analýzy obrazové stránky nových médií*. Vyd.1. Praha: Karolinum, 2009, 245 s. ISBN 978-802-4616-254.

Elektronické zdroje

Adaptic.cz. *Frontend* [online]. 2013 [cit. 2013-07-01]. Dostupné z: <http://www.adaptic.cz/znalosti/slovnicek/frontend>

Cargo Collective [online]. [cit. 2013-07-01]. Dostupné z: <http://cargocollective.com/>

CERN. *Info.cern.ch - the world's first website*. [online]. 2013 [cit. 2013-07-01]. Dostupné z: <http://info.cern.ch/>

DOSTÁL, Martin. *Základy tvorby uživatelského rozhraní* [online]. Olomouc, 2007 [cit. 2013-07-01]. Dostupné z: <http://phoenix.inf.upol.cz/esf/ucebni/gui-dostal.pdf>. Učební text. Univerzita Palackého, Přírodovědecká fakulta, Katedra informatiky.

GNU Operating System. *The GNU General Public License* [online]. nedatováno [cit. 2013-07-01]. Dostupné z: <http://www.gnu.org/licenses/gpl.html>

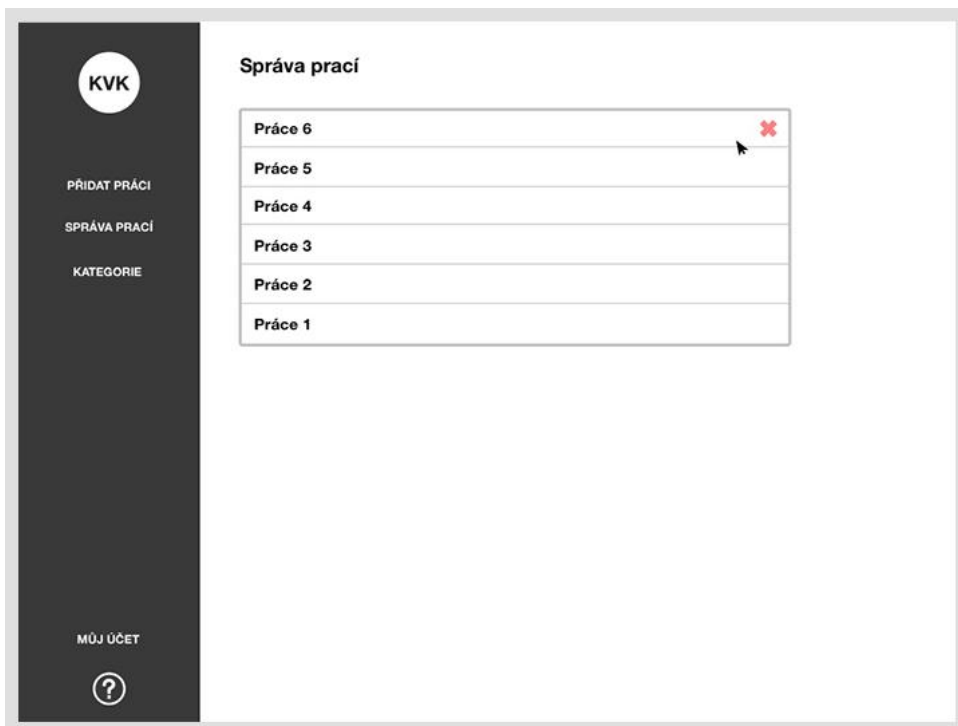
Open Source Initiative. *The MIT License* [online]. nedatováno [cit. 2013-07-01]. Dostupné z: <http://opensource.org/licenses/MIT>

W3Techs. *Historical trends in the usage of server-side programming languages for websites* [online]. 2013 [cit. 2013-07-01]. Dostupné z: http://w3techs.com/technologies/history_overview/programming_language

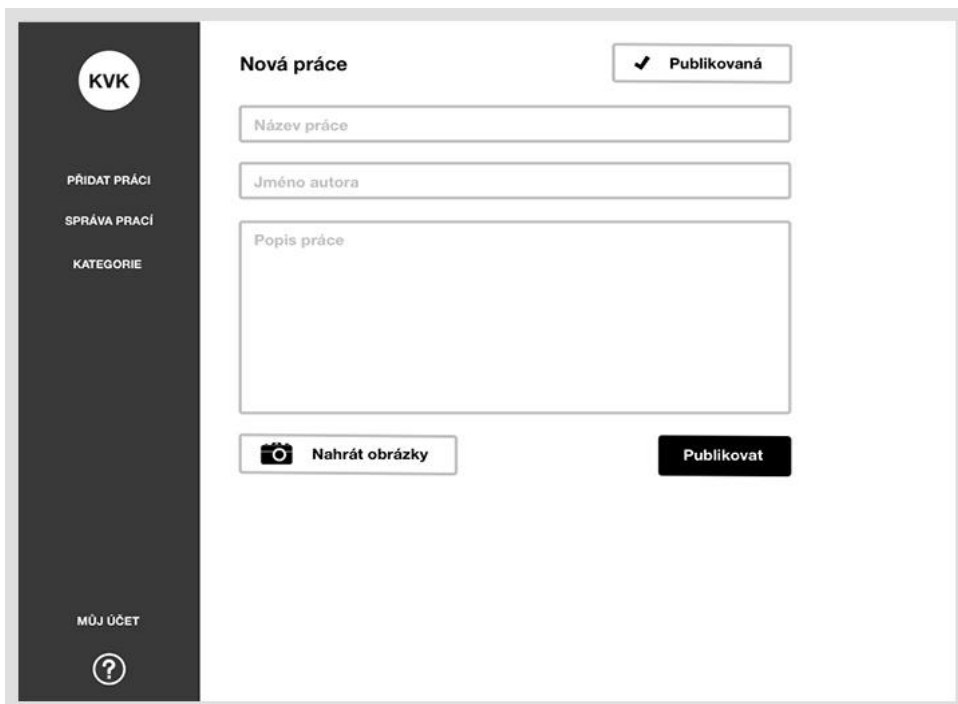
Zdroják.cz. In: *LESS: stejné CSS za méně peněz* [online]. 2010 [cit. 2013-07-01]. Dostupné z: <http://www.zdrojak.cz/clanky/less-stejne-css-za-mene-penez>

Zdroják.cz. In: *Úvod do architektury MVC* [online]. 2009 [cit. 2013-07-01]. Dostupné z: <http://www.zdrojak.cz/clanky/uvod-do-architektury-mvc>

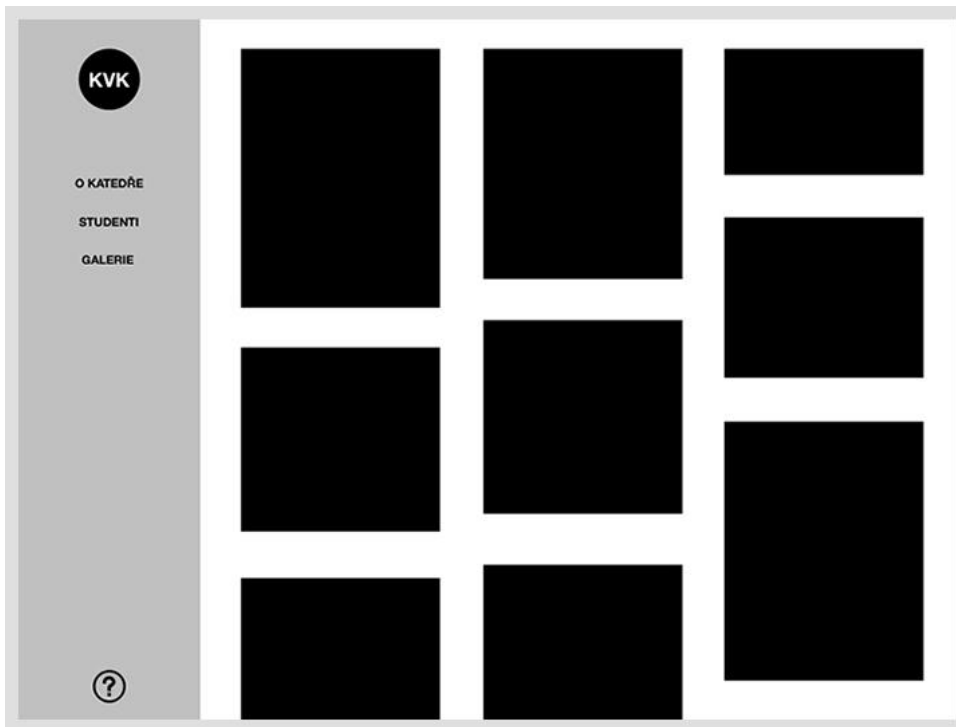
Obrazová příloha



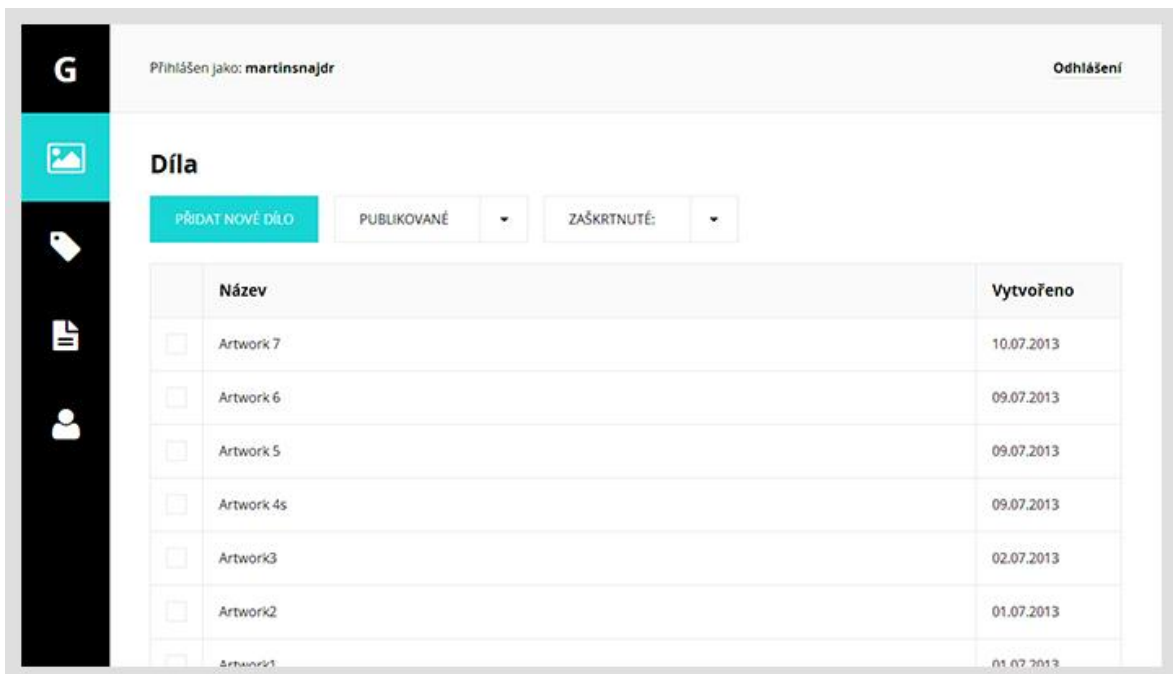
1/ Správa obsahu galerie – wireframe



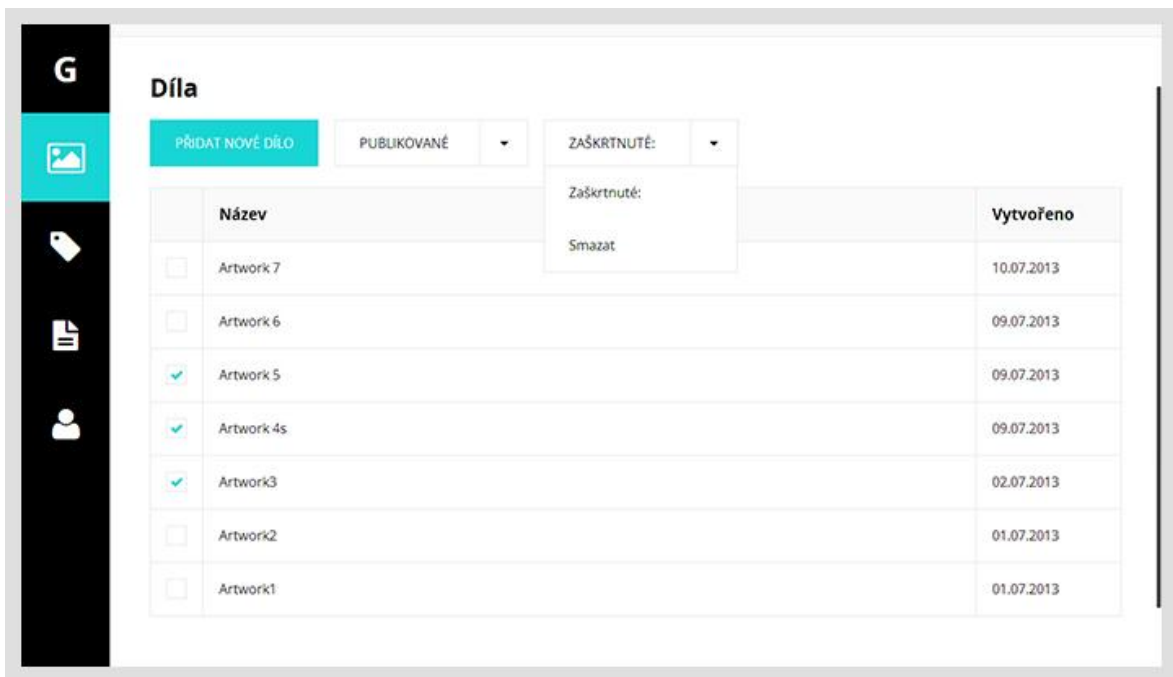
2/ Editace obsahu galerie – wireframe



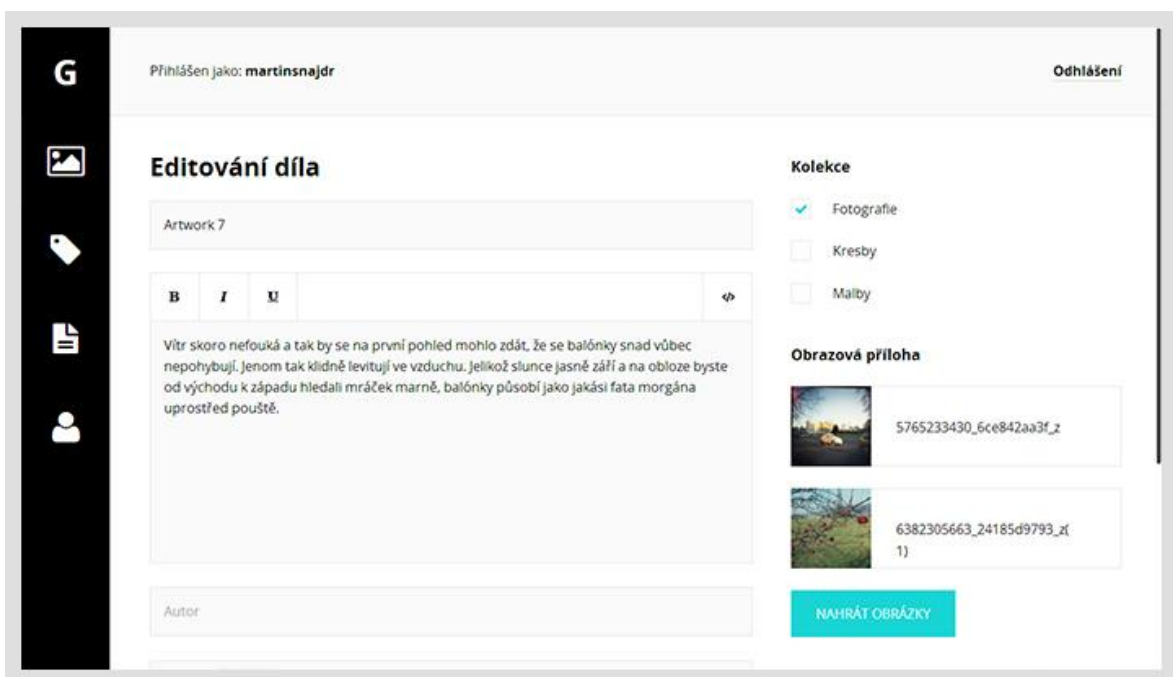
3/ Hlavní stránka galerie – wireframe



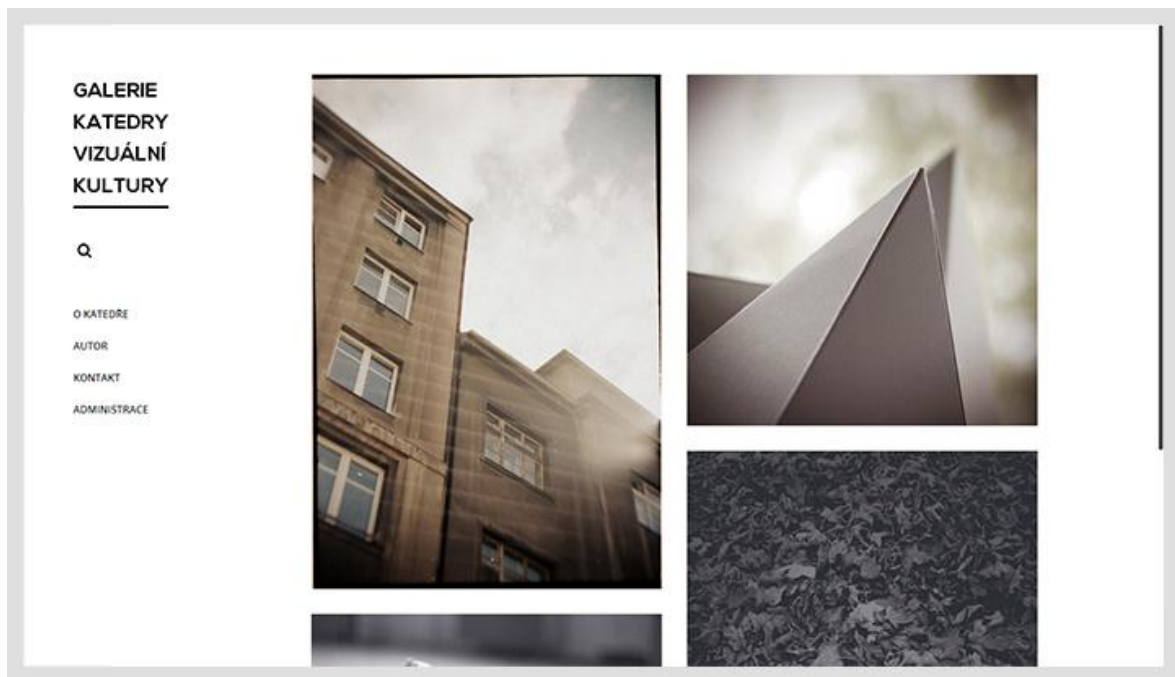
4/ Funkční podoba administrace, verze 1.0, zobrazení přidávaných děl



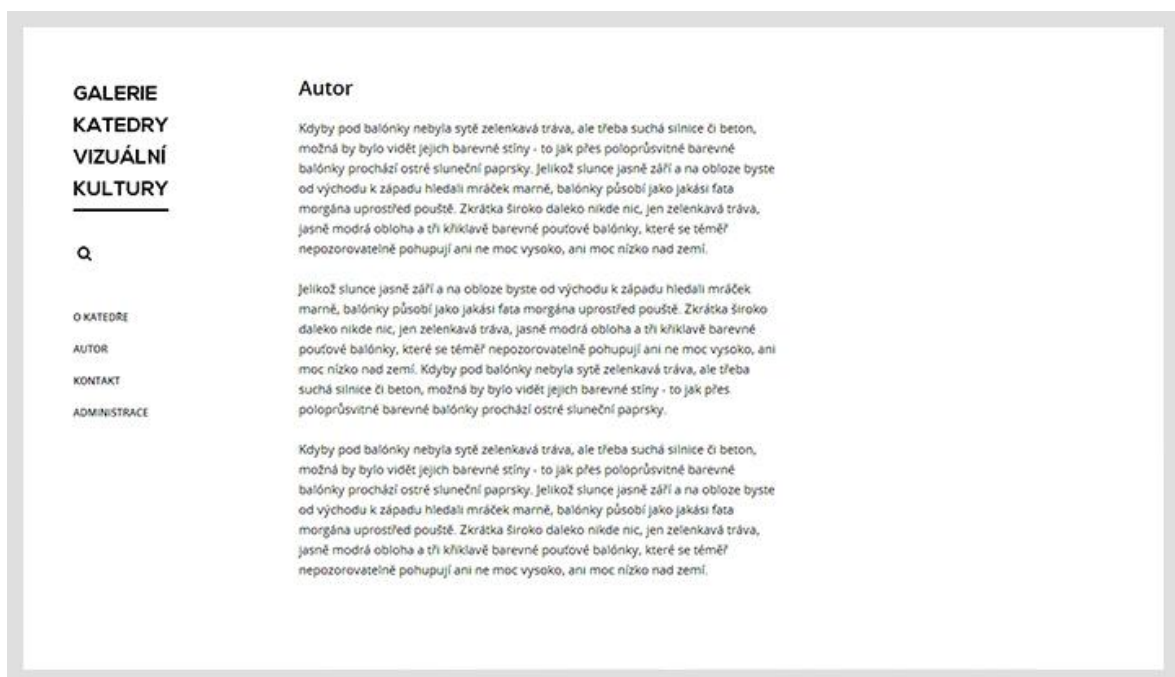
5/ Funkční podoba administrace, verze 1.0, manipulace s díly



6/ Funkční podoba administrace, verze 1.0, editování konkrétního díla

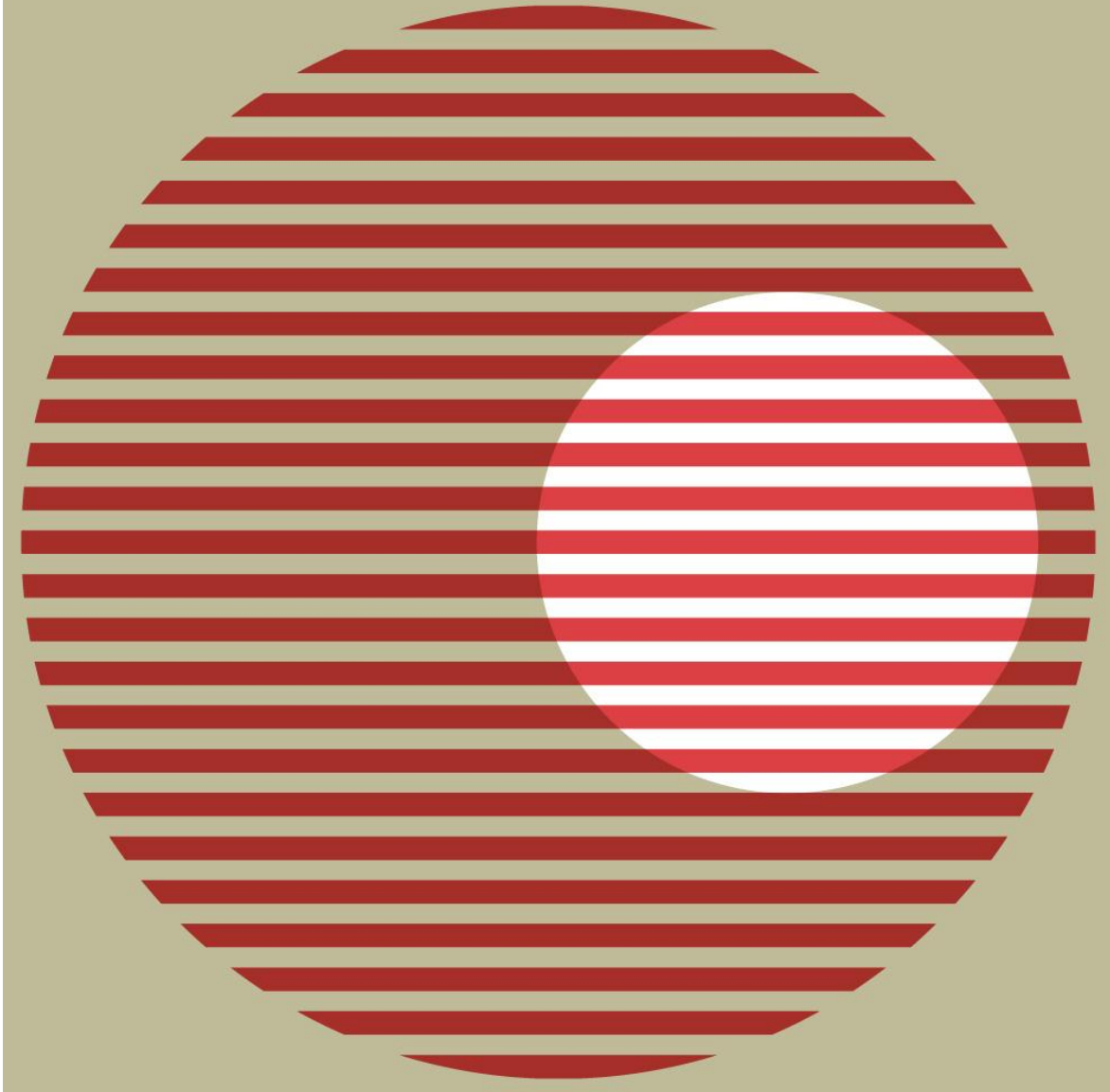


7/ Funkční podoba galerie, verze 1.0, hlavní stránka

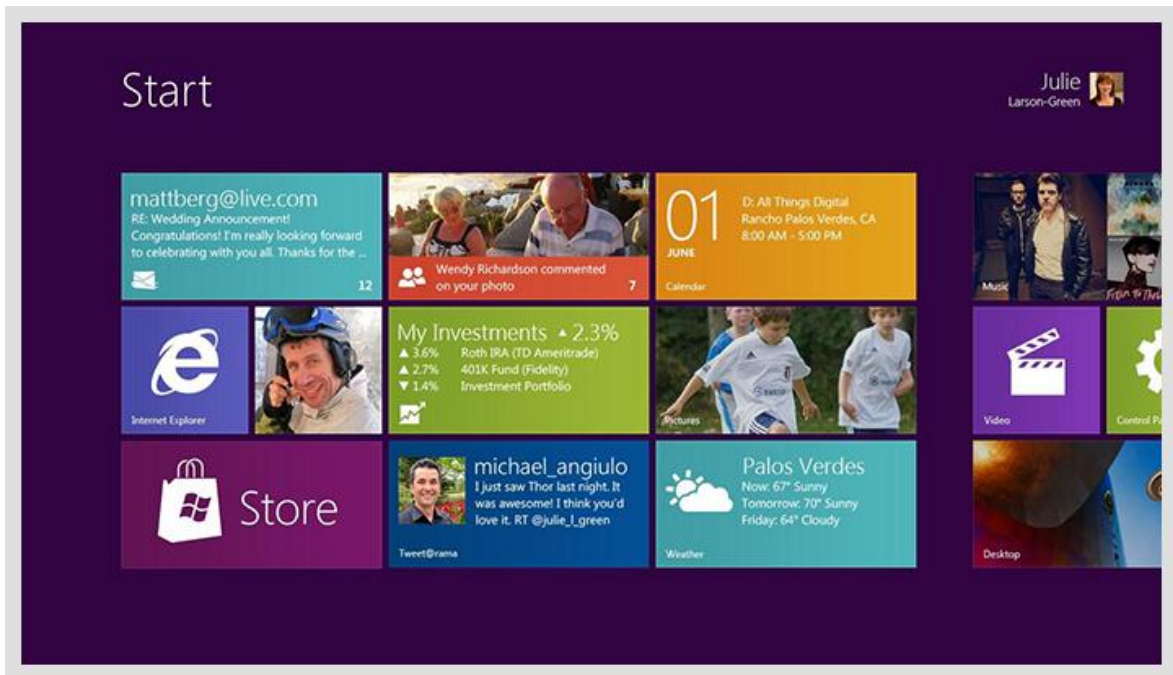


8/ Funkční podoba galerie, verze 1.0, příklad statické stránky (pouze výplňový text)

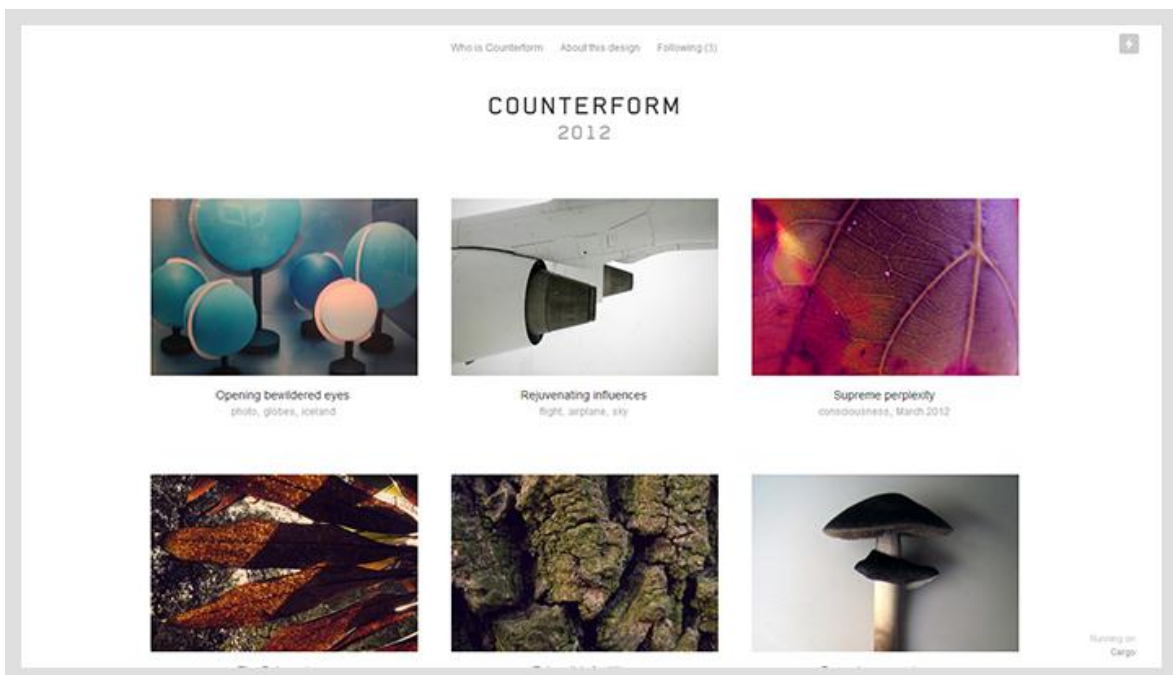
Nothing you can do to your
detergent, fabric softener or
laundry specialty is more
effective than adding a Tinopal
fluorescent brightener



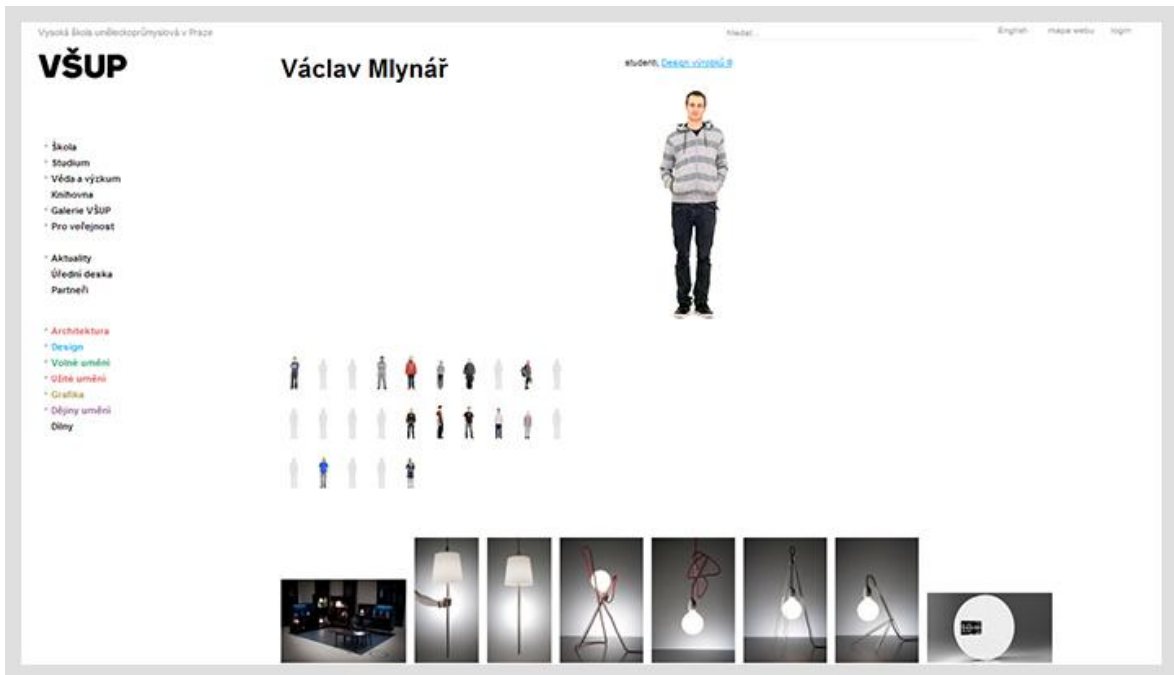
9/ Reklamní plakát pro farmaceutickou společnost Geigy - ukázka švýcarského grafického designu, zdroj: <http://thingstolookat.blogspot.cz/2009/03/geigy.html>



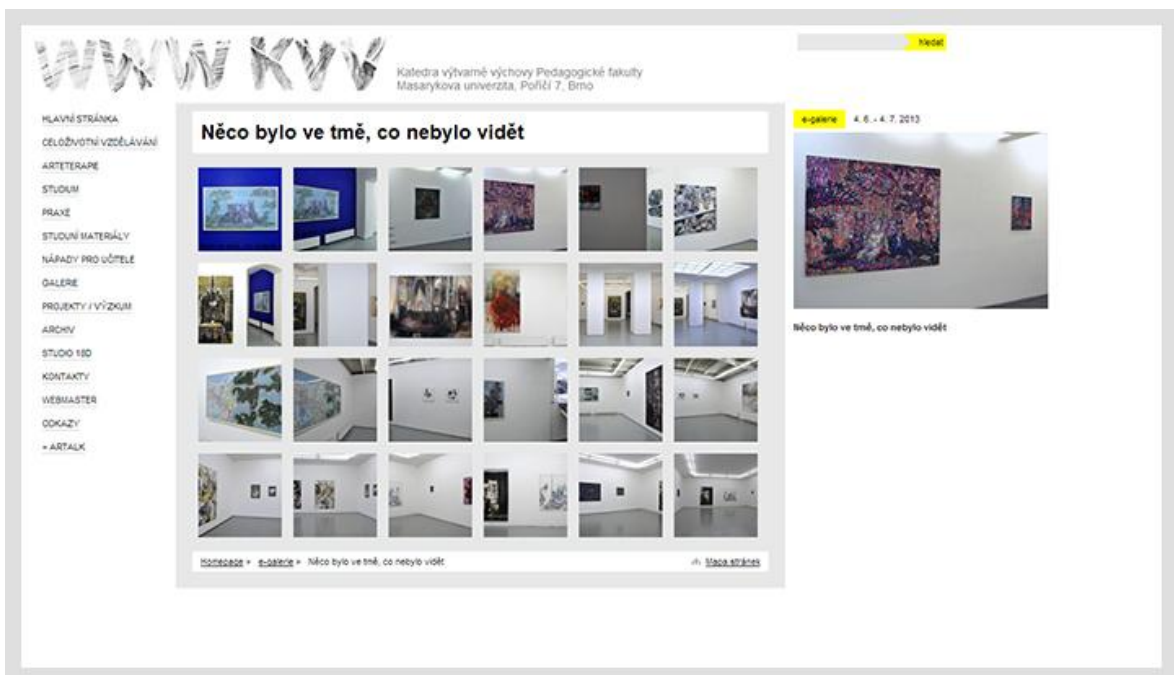
10/ Metro UI, Windows 8



11/ Cargo Collective – ukázka jedné z dostupných šablon portfolio,
<http://cargocollective.com>



12/ Profil studenta VŠUP, <http://www.vsup.cz>



13/ Galerie KVV PED MUNI, <http://www.ped.muni.cz/kvv>