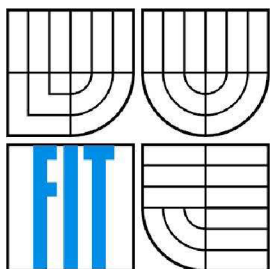


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ  
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY  
DEPARTMENT OF INFORMATION SYSTEMS

## ON-LINE PRODEJ FOTOGRAFIÍ

ON-LINE PHOTOGRAPH SALE

BAKALÁŘSKÁ PRÁCE  
BACHELOR'S THESIS

AUTOR PRÁCE  
AUTHOR

Pavel Vančuřík

VEDOUCÍ PRÁCE  
SUPERVISOR

Ing. Radek Burget, Ph.D.

BRNO 2008

## **Abstrakt**

Tato bakalářská práce se zabývá vytvářením systému pro prodej fotografií z různých sportovních a společenských akcí přes internet.

Součástí navrženého systému je webová prezentace určená pro zákazníka a administrační rozhraní pro provozovatele elektronického obchodu. To umožňuje spravovat a třídit fotografie a objednávky. V první části práce se věnuji popisu použitých technologií a počátečnímu návrhu projektu. Ve druhé části popisují programový kód této webové aplikace a její funkčnost.

## **Klíčová slova**

Web, PHP, MySQL, HTML, JavaScript, elektronický obchod, webová fotogalerie, redakční systém

## **Abstract**

This bachelor's thesis deals with developing a system for selling photographs from various sport and cultural events via internet.

The designed system contains a web presentation for customers and an administration interface for e-shop provider. The interface contains functions for editing and sorting the photographs and orders.

The first part of this thesis contains a description of technologies used for the development of the application and its concept. In the second part, there is a description of the programming code and the application functionality.

## **Keywords**

Web, PHP, MySQL, HTML, JavaScript, e-shop, web photogallery, editorial system

## **Citace**

Pavel Vančurík: On-line prodej fotografií, bakalářská práce, Brno, FIT VUT v Brně, 2008

# On-line prodej fotografií

## Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením Ing. Radka Burgeta, Ph.D.

Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....  
Pavel Vančůrk  
13.5.2008

## Poděkování

Chtěl bych tímto poděkovat svému vedoucímu Ing. Radkovi Burgetovi, Ph.D. za podnětné návrhy týkající se řešení této ročníkové práce a za to, že mi poskytl informace a odkazy na zdroje vztahující se k řešené problematice.

© Pavel Vančuřík, 2008.

*Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů..*



# Obsah

Obsah .....	7
1 Úvod.....	8
2 Použité technologie .....	9
2.1 HTML .....	11
2.2. PHP .....	12
2.3. MySQL.....	13
2.4. JavaScript .....	13
3 Třídění fotografií.....	13
3.1 Rozpoznávání obrazu .....	14
3.2 Čipové technologie.....	14
3.2 Třídění přes www prohlížeč .....	15
4 Ukládání fotografií.....	16
5 Návrh databáze.....	16
5.1 Nákupní košík .....	18
5.2 Role uživatelů.....	19
6 Administrační systém.....	20
6.1 Prostředí systému .....	21
6.2 Třídící formulář .....	22
6.3 Implementace administračního systému .....	23
7 Prezentační část.....	25
7.1 Implementace prezentační části.....	27
8 Používání systému .....	28
8.1 Nahrání a třídění fotografií.....	28
8.2 Objednání fotografií .....	29
8.3 Doručení fotografií.....	30
9 Bezpečnost .....	31
9.1 SQL Injection .....	31
10 Testování a úpravy aplikace.....	33
11 Závěr .....	35
Literatura a prameny .....	36
Seznam příloh .....	37

# 1. Úvod

Motivací pro výběr tohoto tématu bakalářské práce bylo vytvoření systému (webové aplikace) pro prodej fotografií přes internet. V České republice i v ostatních státech existují firmy, které nabízejí fotografie k prodeji z různých sportovních a společenských událostí. Zatím není běžně dostupný software typu klasického internetového obchodu, který by byl pro takovéto použití ideální a dostačující. Uživatel (firma) potřebuje velké množství fotografií rychle a efektivně zveřejnit na internetu a potom spravovat jednotlivé objednávky od zákazníků. Největší důraz při vývoji takovéto aplikace byl kladen na rychlost zpracování, správu a přehled objednávek a třídění velkého množství fotografických dat. Tříděné fotografie jsou tak pro potenciálního zákazníka daleko přístupnější a přehlednější.

Cílem tedy bylo vytvořit funkční model systému jako webovou aplikaci, u které se předpokládá, že na vstupu budou fotografie v předem daném rozlišení a datové velikosti. Požadována byla pouze funkčnost aplikace, nikoliv grafické a designové ztvárnění prezentace. Grafická a programová část vývoje webových aplikací v dnešní době již bývá oddělená. I při vývoji této aplikace jsem se tedy držel takového modelu.

## 2. Použité technologie

Pro vývoj webové aplikace byly zvoleny následující technologie. Pro psaní zdrojových textů jsem využil vlastností editoru PSPad verze 4.5.3. Tento produkt spadá do kategorie freeware. Umožňuje vyhledávání v několika souborech sdružených do jednoho projektu nebo obsažené v určitém adresáři, který lze ve filtru výběru specifikovat. Jeho další velkou výhodou je schopnost zvýrazňování syntaxe různých programovacích jazyků i jejich kombinací. Například velmi častou kombinaci PHP s HTML zvládá bez problémů a už si skoro nedokáži představit psát webový kód v editoru, který tímto nedisponuje. Navíc při tak omezeném způsobu ladění, které je při programování v PHP dostupné. Zvýrazňování syntaxe je také možné si nastavit ručně. Je tedy čistě na uživateli jaké nastavení si zvolí a na co je zvyklý. PSPad je český produkt z plzeňské dílny Jana Fialy a společnosti Prag-Soft s.r.o.

Nebylo použito takzvaných Wysiwyg editorů. Tyto editory se snaží ulehčit webovým designérům (nebo spíše začínajícím tvůrcům webu) práci a pod heslem What You See Is What You Get umožňují jednodušší vytváření a editaci HTML kódu. Bohužel moje zkušenosti s těmito programy mne vedly k jejich nepoužití při vývoji této bakalářské práce.

Kvůli rozdílnému zobrazování HTML kódu v různých prohlížečích byly pro testování použity tyto prohlížeče: Internet Explorer 6.0, Internet Explorer 7.0, Mozilla 1.5, Mozilla Firefox 2.0 a Opera 9.24. Ve výsledném produktu nebylo dosaženo přesně stejné funkce aplikace ve všech prohlížečích. Ve všech uvedených prohlížečích ale nedocházelo k vážnějším chybám v zobrazování a nebyla porušena žádná z klíčových funkcí aplikace. Byla však stanovena priorita optimalizace, a to pro nejpoužívanější prohlížeč – Microsoft Internet Explorer verze 6.0. O tom, že tento prohlížeč je nejpoužívanější mne vedlo zjištění na základě statistik, které zobrazilo počítačadlo na jednom virtuálním serveru, jehož návštěvnost byla zhruba 150 lidí denně. To není úplně mnoho, ale Internet Explorer 6.0 měl v době psaní této práce jasnou početní převahu co se týče používání uživateli nad ostatními prohlížeči. Aby bylo ale měření objektivní muselo by se určitě provádět na více serverech. Tím by se docílilo zrušení omezení jen na určitou skupinu lidí, které je jistě u každého serveru patrné. Naopak vývoj a testování probíhalo v Mozille Firefox. Tento vyspělý software poskytuje řadu výhod, které jsou webovými vývojáři velmi ceněny a které také ve značné míře usnadňují práci.

Především se jedná o doplněk aplikace (tak zvaný AddOn) Firebug. Pomocí něj může vývojář například snadno zjistit vlastnosti daného elementu v html stránce, zobrazit si jeho umístění v kódu a případně tyto vlastnosti měnit. Je schopný přehledně zobrazit strukturu DOM modelu. DOM model (Document Object Model – objektový model dokumentu) je objektově orientovaná reprezentace

dokumentu. Je to rozhraní pro přístup k modifikaci obsahu, struktury nebo částí dokumentu. Vznikl na základě myšlenky konsorcia W3C standardizovat manipulaci s HTML elementy, která je důležitá pro technologie pracující na straně klienta - například JavaScript. Specifikace DOM je platformově a jazykově nezávislá. DOM umožňuje přístup k dokumentu jako ke stromu. Firebug dále umožňuje rychlé zobrazení použitých CSS stylů. CSS (Cascading Style Sheets – kaskádové styly) je jazyk pro způsob zobrazení webových HTML, XHTML nebo XML stránek. Byl vytvořen pro oddělení vzhledu dokumentu od jeho obsahu. Toto měl umožňovat již jazyk HTML, ale z důvodu nedodržování standardů se později vyvinul jinak. Například HTML obsahuje mnoho elementů, které nepopisují obsah a strukturu dokumentu, ale i jeho zobrazení. Kvůli zpracování dokumentů a vyhledávání informací v nich ale není takový vývoj žádoucí. Další výhodou doplňku Firebug je možnost zjištění doby stahování a výpisu jednotlivých obrazových souborů, které se na dané webové stránce zobrazují. Tímto lze velice dobře detekovat obrázky, které se načítají pomalu a snižují rychlost stahování celé stránky. Dnes v době vysokorychlostních připojení k internetu by se mohlo zdát, že na takové detaily se už nemusí brát zřetel, ale opak je mnohdy pravdou. Například u webových sídel, které každý den navštíví několik miliónů uživatelů je velmi důležité, aby se ze serverů nestahovalo zbytečně mnoho dat. Na optimalizaci kódu a celých www stránek jsou dnes již specialisté, kteří se tímto problémem zabývají a činí velké webové aplikace úspěšnějšími. Firebug také v přehledu obrázků dané stránky přehledně vypíše zdroj jejich stahování. Lze tak při vývoji odchytnout ty obrázky, které nejsou například uloženy na serveru odkud se stahuje nebo generuje HTML kód. Další dobrou vlastností doplňku Firebug je možnost ladit skripty napsané v jazyce JavaScript.

Dalším dobrým pomocníkem při tvorbě webů za použití Mozolky Firefox je doplněk Web Developer Toolbar – nástrojová lišta do horní části prohlížeče určená především pro programátory webu. Umožňuje například zamezení načítání CSS stylů, obrázků JavaScriptu nebo aplikací Java. Mnoho uživatelů má například vypnutý JavaScript u svého prohlížeče. Pokud vývojář potřebuje navodit ladící prostředí, které je podobné tomu, jaké používají někteří návštěvníci internetu, tak jednoduše vypne v horní liště požadované technologie. Pomocí zmíněné lišty lze také například ohraničit tabulky, jejich buňky, oblasti `<div>` a jiné elementy. Zapnutí ohraničení `<div>` je velice přínosné a vytváří přehled o pozicování a velikosti jednotlivých elementů na stránce. Při vývoji se tak nemusí požívat staré metody vybarvování elementu jinými barvami pro jejich odlišení a rozpoznání. Pomocí nástrojové lišty lze také vypnout Cookies, což jsou data uložená v paměti prohlížeče a slouží pro uchovávání informací při přechodu mezi jednotlivými stránkami. Takto jsou často řešeny nákupní košíky na elektronických obchodech, pohyby na www stránkách zabezpečených heslem a další. Nejen pro tyto vlastnosti dává hodně uživatelů přednost Firefoxu před rychlým prohlížečem, jakým je Opera. Tvůrci Opery ale slibují uvedení dodatečných funkcí jako Firebug v brzké době společně s novou verzí Opery.

Jako emailový klient sloužící k automatizaci procesu doručení výsledné objednané fotografie byl použit Mozilla Thunderbird verze 2.0.0.14. Tento produkt je zdarma a nabízí mnoho užitečných

vlastností a umožňuje rozdílné nastavení pro jednotlivé poštovní účty. Z pohledu přínosu pro aplikaci vyvíjenou v rámci této bakalářské práce bylo důležité, že tento emailový klient podporuje tzv. drag and drop (chytň a pusť) na soubory nebo pouze odkaz na existující soubory nacházející se v lokálním systému souborů. U Mozilly Thunderbird je třeba soubor přenést myší do oblasti pro adresáty a zprávy. Ve stádiu vývoje je nyní i doplněk, který by měl umožnit při přenesení souboru přes tlačítko nové zprávy automatické vytvoření nové emailové zprávy už s přiloženým patřičným souborem. Trochu rozdílně se například chová produkt Lotus od firmy IBM, kde přiložené soubory jsou zařazeny bez nějakého výraznějšího ohraničení do samotného textu emailové zprávy. Takové „nestandardní“ chování může některým uživatelům velice vyhovovat a některým zase přijde dosti neobvyklé a neohrabané. Každopádně nad požadovanými a naopak zbytečnými vlastnostmi jednotlivých emailových klientů by se daly vést dlouhé diskuze.

Bližší pojednání o dalších jednotlivých webových technologiích přinesou následující podkapitoly.

## 2.1. HTML

HTML je zkratka pro hypertextový značkovací jazyk – Hyper Text Markup Language. Kód napsaný v tomto jazyce stále tvoří základ většiny webových stránek. Tento kód přeloží internetové prohlížeče zobrazí konečnou podobu stránek, tak jak ji známe z internetu. Tento jazyk prochází vývojem a nové verze prohlížečů podporují stále novější funkce. V tomto kontextu je důležité zmínit společnost s názvem W3C, která stanovuje validitu HTML kódu. Kód, který vyhovuje standardům W3C by měl být teoreticky zobrazitelný stejně ve všech prohlížečích podporujících standardy W3C. Skutečnost je ale jiná, a tak je prakticky nemožné dosáhnout plné kompatibility webové aplikace ve všech internetových prohlížečích. V praxi se řeší tento problém většinou tak, že se stanoví priorita pro nejpoužívanější prohlížeč skupiny návštěvníků dané webové aplikace a pro ni se vše optimalizuje. Podle zkušeností s uživateli jsem zvolil optimalizaci pro Microsoft Internet Explorer verze 6.0. Rozlišení obrazovky koncového uživatele je také věc, pro rozhodování se o optimalizaci. Na základě údajů z počítačů vím, že nejvyšší počet návštěvníků internetových stránek používá ještě rozlišení obrazovky 1024x768pixelu. Z důvodu neustálého vývoje a obnovování hardwaru jsem ale zvolil optimalizaci pro rozlišení 1280x1024pixelu, což většinou odpovídá velikosti monitoru 19". Největší internetové servery předpokládají už právě takovéto rozlišení u svých návštěvníků. Rozlišení obrazovek je také neustálým tématem v různých internetových diskuzích zabývajících se tvorbou www. Názory na věc se velmi různí. Někteří uživatelé, i když používají velké rozlišení monitoru, dávají raději přednost menší velikosti stránek, s tím že nebudou mít webový prohlížeč přes celou obrazovku. Podle mého názoru je ale většina uživatelů internetu zvyklá mít prohlížeč v maximální velikosti. Pro tuto variantu byla také webová aplikace optimalizována. Technologie HTML bylo použito kvůli požadované jednoduchosti prezentační části systému, kde se očekává další spolupráce s grafickým

pracovníkem a vylepšení vzhledu. V budoucnu bude jistě využito CSS kaskádových stylů pro určení vzhledu aplikace. Zatím ale toto nebylo nutné a ani to zadání práce nepožadovalo.

## 2.2. PHP

PHP – Hypertext Preprocessor. Tato technologie je základním stavebním kamenem pro tvorbu administračního systému. PHP je skriptovací jazyk pracující na straně serveru, kde jsou fyzicky umístěny zdrojové soubory webové aplikace. Kód napsaný v PHP zajistí, že se uživateli zobrazí na stránce právě ty údaje, které požaduje. Přitom však tvůrce webu nemusí pro každý takovýto požadavek napsat každou HTML stránku zvlášť. Generování HTML kódu provede PHP na základě požadavku uživatele. PHP je velmi oblíbený jazyk pro tvorbu aktivních webových prezentací také z důvodu nulové pořizovací ceny. Nejnovější verze jsou zatím vždy uvolňovány jako freeware. Ve spojení s řízením báze dat MySQL (popsáno níže) poskytuje PHP výkonný prostředek pro aktivní webová sídla. PHP a MySQL jsou většinou instalovány na webových serverech běžících na platformě Unix. Čím dál tím častěji se ale při výběru vhodného webhostingu můžeme setkat s možností provozovat stránku na serveru s operačním systémem Windows poskytující podporu PHP a MySQL. V této souvislosti zmíním i další prostředek pro vývoj mé aplikace. Je jím virtuální server Apache, který jsem si nainstaloval na svém počítači. Pomocí něj jsem mohl zkoušet a ladit aplikaci v režimu offline bez nutnosti použít připojení do sítě Internet. V režimu offline, ale nesmí být webový prohlížeč. V takovém případě by nebyl schopen vygenerované stránky zobrazit. S novými verzemi PHP se také mění některá nastavení překladače tohoto jazyka. Největším kamenem úrazu poslední doby je různé nastavování proměnné „register\_globals“. Někteří provozovatelé webhostingů z důvodu větší bezpečnosti webových aplikací mají nastavenou hodnotu této proměnné na off. Obecně je doporučeno takovéto nastavení od PHP verze 4.1. Přenos prezentace ze serveru s nastavenou direktivou register\_globals on na server s nastavením register\_globals off způsobí obvykle velké problémy vývojářům aplikace. On/Off nastavení této direktivy umožňuje/neumožňuje přistupovat k proměnným předávaných do skriptů metodou GET nebo POST stejně jako k lokálním proměnným používaných v rámci jednoho PHP skriptu. U aplikací typu administračního systému e-shopu, který zrovna neobsahuje žádná extra důležitá data se zdá nastavení register\_globals na off jako zbytečnost. Bohužel čím dál více hostingů s PHP má tuto direktivu takto nastavenou a při vývoji aplikace nezbývá nic jiného, než s touto skutečností počítat. Pokud totiž napíšeme kód, který bude fungovat s nastavením off, pak bude fungovat správně i s nastavením on. Naopak tomu tak není.

## 2.3. MySQL

Pro aktivní internetové aplikace se ve spojení s PHP nejčastěji používá systém řízení báze dat MySQL. Při vývoji aplikace jsem používal pro přístup do databáze webové rozhraní PHPMyAdmin verze 2.6.0. Tento „ovladač“ databáze používá pro svou jednoduchost ovládání velké množství tvůrců webu. Tento produkt je také freeware a je snadné jej převést na vyšší verzi. Zatím nová verze vždy byla doposud také freeware. Pro ladění a vývoj celého bakalářského projektu jsem využil balíčku PHPTriade, který nainstaluje do lokálního počítače PHP, MySQL a webový server Apache. Verze této oblíbené triády je již poněkud starší. Pro navození stejných podmínek jako na konečném webovém serveru jsem však jen nahrál vyšší verzi programu PHPMyAdmin a změnil implicitní nastavení `register_globals` na `off` v souboru `php.ini`. Po takovéto změně jsem nezpozoroval rozdíl mezi chováním vyvíjené aplikace na lokálním počítači a na skutečném webovém serveru.

## 2.4. JavaScript

Provádění kódu jazyka JavaScript má na rozdíl od PHP na starosti internetový prohlížeč. Stejně jako HTML kód, tak i kód napsaný v jazyce JavaScript interpretuje prohlížeč uživateli až na lokálním počítači. V poslední době se objevují skripty napsané v tomto jazyce různě zhuštěné a nepřehledné. To proto, že člověk, který takovýto skript naprogramoval si nepřeje, aby běžný uživatel internetu mohl jeho skript použít pro své účely. Zdrojový kód HTML a JavaScriptu lze totiž přes webový prohlížeč také zobrazit a uložit. K zakódování programu v JavaScriptu tak přicházejí ke slovu nejrůznější programy. Jedním z nich je i ten, který v JavaScriptovém kódu vhodně zamění názvy proměnných tak, že když jej někdo chce znovu použít a modifikovat, není to úplně jednoduché. JavaScriptu se obecně při tvorbě webu nejvíce používá na implementaci nejrůznějších menu, kontroly správného vyplnění polí ve formuláři, zrychlení práce na www stránce apod.

# 3. Třídění fotografií

Asi nejzajímavější částí projektu jsou úvahy a možné implementační řešení třídění fotografií. Vychází se z toho, že každá fotografie má vlastnost, podle které se dá fotografie nějak začlenit. Může to být startovní číslo závodníka, které je rozpoznatelné z fotografie, místo focení, EXIF informace JPEG datového souboru apod. K poslední době je mnoho sportovních akcí, kde každý její účastník (závodník) má někde viditelně připevněné startovní číslo. Toto číslo je v rámci závodu jedinečné a dá se tedy podle něj třídit.

### 3.1. Rozpoznávání obrazu

Mezi nejvíce složité, ale asi nejrychlejší způsoby se jeví použití nějakého algoritmu pro rozpoznávání obrazu. Vstupem by byly fotografie rozumného rozlišení a výstupem soubor nebo tabulka databáze přiřazující každému startovnímu číslu určitý počet fotografií, na kterých bylo toto číslo nalezeno. Velký problém ale spočívá v tom, že skoro pro každou sportovní akci se vyrábějí jiná startovní čísla. Barva, font, podklad a velikost těchto čísel je různá. Navíc fotografové nejsou schopni vyfotit každého závodníka ze stejného úhlu. Světelnost jednotlivých fotografií také bývá velice různá. Pro implementační náročnost a nezaručení stoprocentní funkčnosti takového způsobu jsem se rozhodl jej nepoužít a uvažovat o jiných možných alternativách.

### 3.2. Čipové technologie

V poslední době si pořadatelé sportovních akcí najímají na závody profesionální firmy zabývající se měřením časů jednotlivých závodníků. Časoměři obvykle využívají technologie aktivních čipů s digitálním přenosem dat. Jedná se o bezkontaktní identifikační systém. Anténka s přijímačem, komunikační a obslužný software, zajišťují zpracování velkého množství dat v reálném čase. Každý závodník má na svém těle nebo na kole připraven čip. Čip je speciálně vyvinutý miniaturní vysílač s vlastním zdrojem. Plastický vodotěsný obal mu umožňuje použití v jakýchkoliv povětrnostních podmínkách. Nabízí se možnost součinnosti s takovouto technologií. Na měřících kontrolách jsou časoměři schopni zjistit přesný čas průjezdu závodníka i jeho startovní číslo. Pokud by stál fotograf v blízké vzdálenosti měřící kontroly, může se přiřadit podle času focení k dané fotografii i startovní číslo zjištěné časoměřičskou firmou. Každý digitální fotoaparát zapisuje do formátu JPEG (nejčastější fotografický formát pro hromadné využití) datum a čas pořízení fotografie. Kromě tohoto údaje jsou v sekci EXIF u každého JPEG datového souboru i další informace o fotce. Například typ fotoaparátu, kterým byl snímek pořízen, délka expozice, nastavení objektivu, použití blesku a jiné. Všechny tyto EXIF data lze z JPEG souboru extrahovat nejrůznějšími offline programy i on-line pomocí PHP grafických funkcí. Pro velkou náročnost sehrání fotografií s časoměři a zatím neexistující sjednocený formát výstupu dat obou stran je toto řešení zatím pouze ve stádiu úvah. V budoucnu by ale takováto kooperace určitě přinesla výrazný posun v dosud nabízených službách fotografických firem.



### 3.3. Třídění přes www prohlížeč

Nejjednodušším způsobem je ruční třídění fotografií. Časově je tato varianta nejdelší. Doba třídění se ale dá jistým způsobem minimalizovat. Nabízí se několik variant řešení. S ohledem na rychlost zobrazování fotek na www jsem uvažoval o dvou variantách.

Fotografie jsou roztríděné v adresářích, jehož název nese startovní číslo závodníka, jehož fotky jsou v adresáři obsaženy. Výhodou tohoto řešení je velice rychlé zobrazování fotografií na webu. Funkce PHP pouze projde daný adresář a konkrétní fotografie pošle prohlížeči. Nevýhodou ale je nemožnost přehledného vyhledávání například podle autora, místa a data pořízení fotografie. Malé náhledy fotografií jsou také na serveru duplicitně. Jednou kvůli prohlížení všech fotografií ve velkém adresáři a potom v adresářích s názvem podle startovního čísla. Problémem také bývá časté přerušení spojení při nahrávání fotek na server pomocí protokolu FTP – File Transfer Protocol některým z prohlížečů souborového systému jakým je například Total Commander.

Tyto neduhy řeší spojení s databází MySQL. Toto řešení bylo nakonec implementováno. Každé fotce je v tabulce databáze přiřazen jeden záznam. Ten obsahuje jedinečný kód, autora, místo, datum a název akce, v rámci které byla fotografie pořízena. Díky tomu lze fotografie podle těchto parametrů rychle a efektivně vyhledávat a zobrazovat v prostředí webu. Počet řádků tabulky databáze sice při předpokládaném počtu 700 až 3000 fotografií za sportovní událost výrazně naroste, ale neznamená to omezení funkčnosti aplikace.

Samotný proces třídění byl z důvodu přenosových rychlostí na internetu přenesen do webové aplikace na lokálním počítači. Do patřičného políčka se zadá startovní číslo a to se uloží k danému záznamu v databázi. Aby bylo třídění efektivní a startovní čísla byly stoprocentně rozpoznatelné, představuje to prohlížení fotografie ve velikosti rozlišení monitoru. Při takovýchto obrazových velikostech dosahuje datová velikost jedné fotografie při vhodně zvolené kompresi zhruba 150-200kB. Na jednu sportovní událost je plánováno 700 až 3000 fotek. Z toho důvodu jsem přistoupil k řešení třídění fotografie offline nad stejnou databází jako na serveru. Potom stačí pouze tabulku databáze s fotografiemi na serveru aktualizovat prostřednictvím prostředí PHPMyAdmin podle tabulky na lokálním počítači. Rychlost načítání při offline prohlížení fotografií je dostačující a hlavně se zamezí zdlouhavému a zbytečnému nahrávání fotek na server. Také nebudou na serveru uložena zbytečně obrazová data velké velikosti, které by mohl některý hacker najít a stáhnout. Na serveru tak jsou uloženy pouze menší náhledy velikosti 180x120px a 300x451px, ze kterých není možné vyrobit ani kvalitní papírovou fotografii 10x15cm.

## 4. Ukládání fotografií

Potenciální zákazník pro své rozhodnutí koupit/nekoupit danou fotografii potřebuje vidět náhled v dostatečném rozlišení. Na druhé straně provozovatel elektronického obchodu s fotografiemi (dále už jen provozovatel) si nemůže dovolit na web vystavit fotografie v rozlišení, které by zákazníkovi stálo za to si je bezplatně stáhnout. Navíc také, jak již bylo zmíněno v minulém článku, jsou přenosové rychlosti dat na www omezené. Byla zvolena varianta dvou velikostí náhledů fotografií. Menší náhled velikosti 120x180px a větší o velikosti 300x450px. Datová velikost malého náhledu se pohybuje okolo 10kB a datová velikost většího náhledu okolo 50kB. To jsou již čísla, která jsou přijatelná pro předpokládané množství snímků. Samotné ukládání fotografií – nahrávání na webový server je možné přes www formulář. Ten ale zatím poskytuje možnost nahrát maximálně 5 souborů najednou. Výběr z umístění na lokálním počítači také zdržuje. Toto řešení s formulářem je tedy hodně neefektivní.

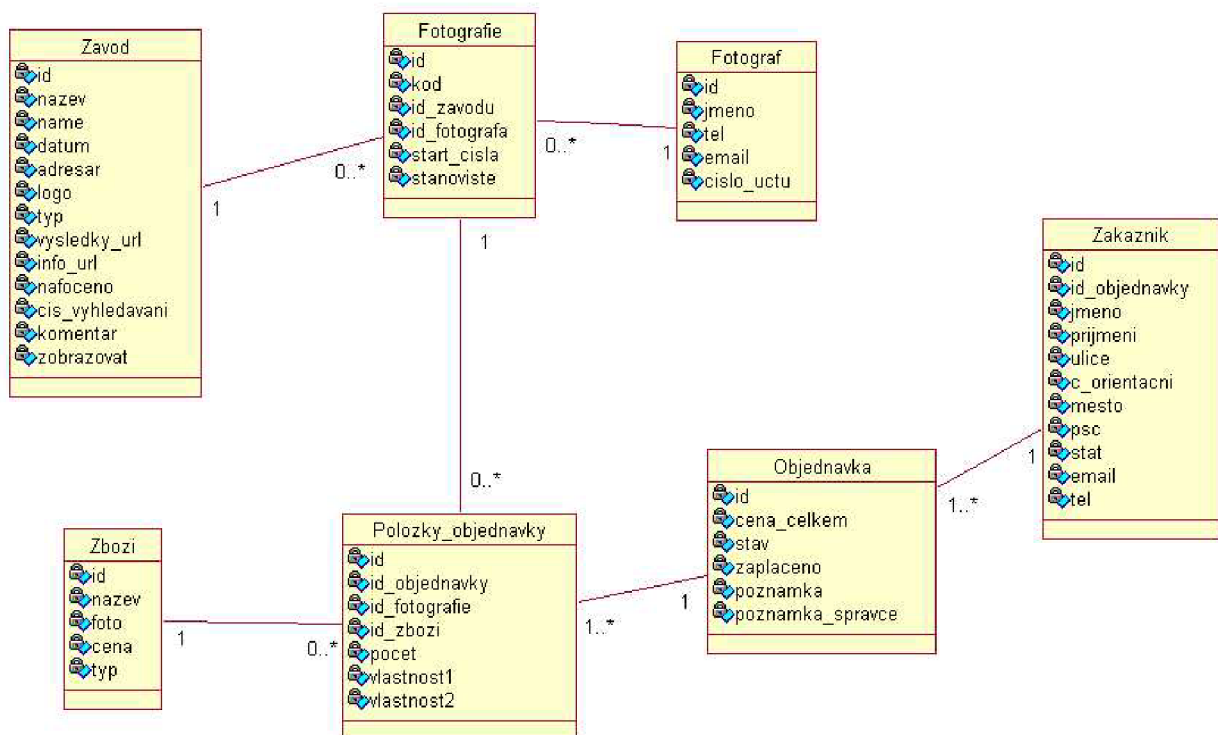
Zatím nejvhodnější a relativně rychlé je nahrávání souborů přes protokol FTP. Navíc také FTP přenos nabízí mnoho uživatelsky příjemných souborových prohlížečů. Tuto variantu jsem v projektu zvolil. Nahrávání fotografií se tak nebude dít přímo prostřednictvím webové aplikace, ale z důvodu velkého množství dat a požadované rychlosti u procesu nahrávání a třídění prostřednictvím FTP protokolu přes souborový prohlížeč.

## 5. Návrh databáze

Tato fáze projektu byla klíčová pro konečnou implementaci a celkovou funkčnost systému. Bylo potřeba si stanovit co které objekty v aplikaci znamenají a jaké operace nad nimi se budou provádět. Přesto, že implementace byla provedena neobjektově ve skriptovacím jazyce PHP, využil jsem modelovacího nástroje Rational Rose Enterprise Edition a podle notace UML si stanovil celkový návrh databáze. Objekty navržené v diagramu tříd (obr.1) korespondují s konkrétními tabulkami v databázi. Vztahy a vazby mezi jednotlivými objekty jsou v diagramu tříd také namodelovány. Například ke každé položce objednávky připadá jedna fotografie. Zde se také projevuje rozdíl mezi běžnými internetovými obchody a navrhovanou aplikací. Zde tvoří položky objednávky právě jeden artikl ze zboží a právě jedna fotografie. V praxi to znamená, že zákazník si objedná požadovanou fotografii a vybere si, jaký produkt z ní chce nechat vyrobit. Běžně se nabízejí například různé velikosti a typy vyvolaných papírových fotografií. Systém byl také navržen tak, aby se zákazník nemusel před každým nákupem přihlašovat k nějakému svému účtu nebo dokonce si takový účet při prvním nákupu vytvářet. Po vybrání požadované fotografie a zboží pouze vyplní doručovací adresu

s kontaktními údaji a tím je objednávka hotová. Vycházel jsem tak z toho, že pro mnohé návštěvníky a zákazníky internetových obchodů je unavující se pokaždé přihlašovat přihlašovacím jménem a heslem ke svému profilu v daném obchodě. Lidé nakupují už na více serverech a nechtějí si z pochopitelných důvodů dávat všude stejná hesla. Ukládat si je v paměti prohlížeče také není ideální řešení, protože například při nákupu z jiného počítače nejsou hesla k dispozici. Vyvíjený systém ani zatím nepředpokládá využití strategie slev a výhod pro stálé zákazníky a tím pádem odpadá nutnost shromažďování objednávek od jednoho zákazníka. Nelze vyloučit, že by v budoucnu něco takového mělo přinést minimálně v podobě lepšího mapování zákazníků. Na to ale z principu není aplikace zaměřena. Ke každé objednávce může být přiřazeno více položek objednávky. Tyto položky mohou mít stejné nebo různé kódy fotografií. Zákazník si tak může objednat z jednoho kódu fotografie například papírový formát 10x15cm, 20x30cm a ještě z dalšího kódu třeba velkoformátový tisk na 20x30cm.

## Diagram tříd



(obr.1)

## 5.1. Nákupní košík

Pro snadné vybírání a editaci položek v objednávce byl implementován tzv. nákupní košík, který umožní před ještě před vyplněním adresy a vlastním potvrzením objednávky editovat a měnit jednotlivé položky v objednávce. Košík je asi nejčastějším řešením jak umožnit zákazníkům pohodlný výběr zboží. Mnou implementovaný nákupní košík využívá tzv. sessions. To jsou data uložená na serveru, která shromažďují informace o uživateli právě prohlížené stránky a umožňují uchování těchto informací při přechodu z jedné stránky na druhou v rámci návštěvy jednoho virtuálního internetového serveru. Použití sessions pro implementaci nákupního košíku je lepší než použití cookies už jen z důvodu menší omezenosti objemu a počtu dat. Počet cookies je obvykle v závislosti na používaném internetovém prohlížeči omezen na 50. Sessions také umožňují ukládání informací do vícerozměrných polí, což je přínosem pro případy, kdy dopředu nevíme kolik formací budeme chtít o zákazníkovi ukládat. Práce se sessions je jednoduchá a k jejich hodnotám se přistupuje v PHP přes superglobální pole `$_SESSIONS`. Před vlastním generováním html obsahu je nutné provést inicializaci sessions příkazem `session_start()`; Dále používám pro přenášení id konkrétní session cookies. To říká příkaz `ini_set("sessions.use_only_cookies", "1");` Tyto inicializace se nacházejí v souboru `top.php`, který se připojuje shora ke každém skriptu zajišťující prezentaci dat normálním uživatelům obchodu.

Pro správný chod aplikace je tedy nutné, aby uživatel měl v prohlížeči zapnuté cookies. Identifikátor session, tzv. SID se dá ještě přenášet přes URL. Ale vzhledem k tomu, že stále více webových aplikací je závislých na použití cookies, předpokládal jsem, že jejich zapnutí je dnes již běžným standardem a uživatele nijak neobtěžuje. To se ale dnes již nedá říct o nutnosti instalace doplňku pro zobrazování Flash animací. V době narůstajícího počtu reklam si někteří uživatelé tento doplněk záměrně neinstalují, aby nebyli vystaveni přehršeli blikajících reklam a nicneříkajících grafických informací. Sessions se dají kompletně smazat příkazem `session_destroy()`; Toho použiji při smazání obsahu košíku. Mohu si to dovolit, protože data uložené `$_SESSION` obsahují pouze informace o stavu košíku. `session_destroy()`; je také dobrým ladícím příkazem při implementaci košíku. Ve spojení s funkcí `var_dump()` jsou nezbytnými ladícími nástroji při programování v PHP. Nevýhodou tohoto skriptovacího jazyka je prakticky nemožnost ladění s využitím nějakého ladícího nástroje jaký se používá například při programování v jazyce C. Programátorovi funkce `var_dump()` s parametrem nějaká proměnná poskytne informace o datovém typu a hodnotě této proměnné. Parametrem může být i několika úrovně pole včetně `$_SESSION`. Vlastní jádro poskytující operace nad nákupním košíkem se nachází v souboru `kosik_akce.php`, který je součástí přílohy. Akcemi jsou: přidání položky do košíku, smazání položky z košíku, smazání celého obsahu košíku a provedení objednávky zboží.

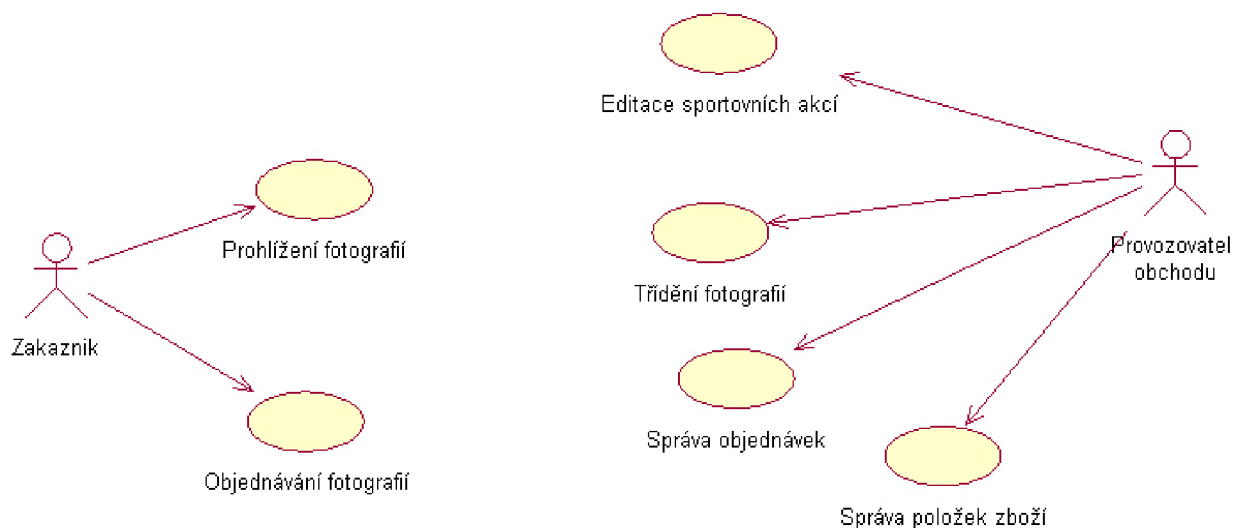
Pro implementaci kontroly polí adresy v posledním kroku objednávky byl použit JavaScript. Takováto kontrola nepatří mezi nejbezpečnější, protože kód JavaScriptu může zkušený uživatel

změnit například pomocí doplňku Firebug v prohlížeči Mozilla Firefox, ale na druhou stranu se jedná o rychlou a výstižnou kontrolu vstupních dat. Dokáže uživateli říct přesně ve kterém poli se dopustil chyby proti povoleným vstupním hodnotám. Kontroluje se například validita emailové adresy, zda PSČ obsahuje pouze čísla atp. Vlastní inicializace funkce v JavaScriptu sloužící pro kontrolu formulářových dat se provádí hned při načtení vygenerované html stránky. Funkce samotná se nachází v samostatném souboru `inc/check_form.js`.

## 5.2. Role uživatelů

Jednotlivými uživateli navrhovaného systému se myslí lidé, kteří mohou, ať již anonymním způsobem nebo po přihlášení se do systému, zasahovat a měnit nějakým způsobem obsah databáze. V první řadě je to návštěvník webových stránek, který má možnost si prohlížet náhledy fotografií a posléze si některou koupit – stává se zákazníkem a jeho údaje jsou vedeny v databázi. Po objednání změní databázi tím, že ji doplní o své doručovací údaje a požadované zboží s vybranými fotografiemi. Druhým typem uživatele je provozovatel obchodu, který prostřednictvím systému třídí fotografie, spravuje údaje o fotografech, jednotlivých fotografických událostech a zákaznících s jejich objednávkami. Jednotlivé činnosti uživatelů modeluje Use Case Diagram na obr.2. Provozovatelů obchodu může být více, ale na to zatím aplikace není nastavena. Nepředpokládá se potřeba přístupu více provozovatelů do administrační části. Pokud by ale takový požadavek nastal, může ho návrhář aplikace řešit přidáním další položky s přihlašovacím jménem a heslem do tabulky databáze. Přihlašovací jméno a heslo každého uživatele je do databáze ukládáno v hash podobě po konverzi funkcí PHP funkcí `md5()`. Message-Digest algoritmus je rozšířená rodina hashovacích funkcí, která vytváří ze vstupních dat výstup (otisk) fixní délky – v našem případě string o délce 32 znaků. Otisk je též označován jako miniatura, hash. Jeho hlavní vlastností je, že malá změna na vstupu vede k velké změně na výstupu, tj. k vytvoření zásadně odlišného otisku. Je jednostranná funkce. Není tedy možné rychle zjistit původní řetězec z výstupního otisku. Dnes sice již existují nástroje na dekodování řetězců zakódovaných pomocí funkce `md5`. Fungují ale na principu postupného zakódování známých a používaných řetězců a porovnávání jejich otisků s potřebným zakódovaným řetězcem. Pokud tedy uživatel administrátor zapomene svoje uživatelské jméno a heslo, musí mu programátor z hlediska bezpečnosti přiřadit jiné a do databáze uložit patřičné hašované otisky těchto řetězců. V případě, že se někdo nepovolaný dostane k datům v databázi. Má tak přinejmenším ztíženou práci tím, že bude muset dekodovat přihlašovací údaje administrátora.

## Use Case Diagram



(obr.2)

## 6. Administrační systém

V této kapitole a podkapitolách objasním funkce samotného administračního systému. Pro přístup do systému musí osoba, jíž byly přiděleny přihlašovací údaje, zadat jméno a heslo do přístupového formuláře. Různé platné přihlašovací údaje mohou mít více osob. Přihlašovací formulář je vyobrazen na obr.3.

The screenshot shows a login form for the website RACEFOTO.CZ. It features a logo in the top left corner, followed by the text 'RACEFOTO.CZ'. Below this, there are two input fields: 'LOGIN:' and 'HESLO:'. At the bottom of the form is a button labeled 'PŘIHLÁSIT SE'.

(obr.3)

## 6.1. Prostředí systému

Po správně zadané kombinaci přihlašovacího jména a hesla se administrátor dostane do administrační části. Pokud zadá administrátor špatné jen špatné už. Jméno nebo heslo, vypíše se hláška o tom, že byla zadána špatná kombinace těchto dvou údajů. Neposkytne tedy potenciálnímu útočníkovi informaci o tom, že se mu podařilo odhalit jeden z přihlašovacích údajů. Administrační část obsahuje funkce na editaci a přidávání závodů, fotografů, stanovišť pro focení a objednávek. Vzhled je tvořen pomocí rámců. Ve vrchním rámcí je menu s volbami k editaci příslušných kategorií, nad ním je zobrazeno jméno aktuálně přihlášeného uživatele a vpravo je tlačítko pro bezpečné odhlášení ze systému. Tím se myslí smazání informací o uživatelovi na serveru – sessions. V levém rámcí se po zvolení určité kategorie objeví bližší přehled o kategorii a případné dodatečné funkce. V hlavním prostředním rámcí je vždy výpis položek dané kategorie. Výpis obsahuje nejnütnější informace potřebné k zorientování se v obsahu databáze a odkaz na editaci příslušné položky. Pro bližší seznámení se s prostředím slouží obr.4. Je to foto obrazovky výpisu závodů.

The screenshot shows the RACEFOTO.CZ administration interface. At the top, there is a navigation menu with icons for 'Přístupy', 'Závozy', 'Fotografové', 'Zboží', and 'Objednávky'. The main content area is titled 'Závozy' and displays a table of race events. The table has columns for ID, Datum, Řazení, Název, Typ, Aktivní, Nafoceno, Logo, and Editace. There are four rows of data. To the left of the table, there are several controls: 'Počet závodů: 4', a 'Seřadit podle' dropdown menu set to 'Datum pro řazení', and buttons for 'Nový závod', 'Typy závodů', and 'Stanoviště'.

ID	Datum	Řazení	Název	Typ	Aktivní	Nafoceno	Logo	Editace
1	8.5.2007	2007-05-08	Starobno Fri...I Tour 2007	MTE	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
5	8.9.2007	2007-09-08	Blanenský Golem	MTE	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
6	15.9.2007	2007-09-15	GT Zaskar Challenge Jedovnice	MTE	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
7	8.5.2008	2008-05-08	Starobno Fri...I Tour 2008	MTE	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		

(obr.4)



Pomocí tlačítek v levém menu lze vytvořit nový závod, nový typ závodu (sportovní akce) a nové stanoviště. Každé fotce je při třídění přiřazeno jedno stanoviště, ze kterého byla pořízena. To usnadňuje zákazníkovi vyhledávání fotografií.

## 6.2. Třídící formulář



697,380

Lucie Canovová

Brod

Nerozpoznatelné číslo

uložit a přejít na další

(obr.5)



Na obr.5 je zobrazen webový formulář pomocí kterého se dané fotografie třídí. Je úmyslně volen jako nejjednodušší možný, aby osobu, která provádí třídění zbytečně nerozptylovaly grafické prvky a nezabíralo se místo v okně prohlížeče potřebné pro zobrazení fotografie v co největším přijatelném rozlišení. S ohledem na velké množství snímků je kurzor nastaven vždy do horního políčka pro startovní čísla. Pokud se na fotce vyskytnou dvě a více startovních čísel, zadají se do formuláře všechna čísla oddělená čárkou. Dalším parametrem fotky je její autor(ka). Ten(ta) se vybere z rolovací nabídky. Stanoviště focení je také již předem nadefinované a vybere se z nabídky. Stanovištěm je v databázi vyčleněna zvláštní tabulka. Je totiž možné, že na různých akcích budou fotografie pořizovány míst, která lze pojmenovat stejně. Před vlastním tříděním fotografií si administrátor vytvoří a pojmenuje jednotlivá stanoviště, která se mu pak budou při třídění zobrazovat v nabídce. Pokud se na obrazovce vyskytne závodník s nerozpoznatelným číslem, zaškrtně se tato volba. Volbu nemusí administrátor zaškrtnout myší, ale může použít i kombinaci TAB(tabulátor) pro přechod mezi formulářovými poli a SPACE(mezerník) pro změnu zaškrtnutí/nezaškrtnutí zatrhávacího políčka.

Po potvrzení formuláře se přejde na další fotku. Rolovací nabídky zobrazí původní zvolené hodnoty. Je to z důvodu rychlosti třídění fotek a typem dat, pro které je systém navržen. Očekává se, že od každého fotografa a z každého stanoviště je několik snímků po sobě. Při třídění je tedy třeba v ideálním případě(mění se jen startovní čísla závodníků na fotkách) pouze pravé části klávesnice u stolního počítače s klasickou klávesnicí. Potřeba jsou číslovky na zadávání startovních čísel, čárka jako jejich oddělovač a klávesa Enter pro potvrzení vložených dat a přechod na další fotku. Při množství přes 1000 snímků se každý zbytečný přehmat počítá a ve výsledku vytvoří zbytečné prodloužení procesu třídění.

Při takovémto způsobu třídění je možné setřídit podle čísel 2000 fotek za 2,5hodiny. Není to ještě zdaleka ideální doba, ale v porovnání s časem, jaký zabere fotografům vymazávání a úprava fotek, je to ještě přijatelné.

### 6.3. Implementace administračního systému

Administrační systém je implementován několika základními skripty, které jsou doplněny o další podle potřebných modulů. Takovými moduly se v rámci vyvíjené aplikace myslí Závody, Fotografové, Objednávky a Zboží. Každý modul je schopen editace a vytváření položek v potřebných tabulkách databáze. Administrační systém lze tedy chápat jako nástroj pro ovládání obsahu v databázi. Lze jej také označit jako CMS – Content Management Systém, neboli systém pro správu obsahu.

Kostru tvoří následující soubory v adresáři administrace

- kernel.php – jádro celého systému. Obsahuje funkce pro všechny potřebné akce v rámci administračního systému.

- `administrace.php` – rozvržení rámců na obrazovce
- `index.php` – úvodní přihlašovací stránka s chybovými hláškami pro špatné přihlášení
- `menu.php` – vrchní navigační menu sloužící pro lepší orientaci v administraci
- `top.php` – zobrazuje informace o právě přihlášeném uživateli
- `left.php` – obsahuje levá menu a zobrazuje příslušné v závislosti na požadovaném modulu
- `plocha.php` – rozvržení rámců a přesměrovávání toku řízení dle přijímaných parametrů v `$_GET` nebo `$_POST`
- `styleadm.css` – definované kaskádové styly pro administrační systém
- `modul_vypis.php` – zobrazuje výpis položek daného modulu, kde „modul“ v názvu souboru znamená například fotografove
- `req/modul_form.php` – formulář pro editaci položek daného modulu, kde „modul“ v názvu souboru znamená například fotografove
- `req/verify.php` – kontrola přihlášení a doby mezi dvěma po sobě jdoucími akcemi ze strany uživatele. Pokud je prodleva větší než stanovená doba, dojde k automatickému odhlášení ze systému. Doba je nastavena na 30minut a je definována v konfiguračním souboru `config/defaults.php`
- `req/hlavicka.php` – obsahuje doplňující JavaScriptové funkce

Mimo adresář `administrace` jsou velmi důležité konfigurační soubory, které jsou uloženy v adresáři `config`. Jedná se o soubory:

- `config.php` – obsahuje heslo a jméno databáze
- `databaze.php` – připojení k dané MySQL databázi
- `defaults.php` – obsahuje klíčové proměnné a jejich hodnoty potřebné pro celou aplikaci

Dalším typem souborů, které obsahují sdílené proměnné funkce jsou ty, které jsou uloženy v adresáři `inc`. Jsou to:

- `functions.php` – obsahuje sdílené funkce pro celou aplikaci. Například funkci `db_dotaz()`, která je velmi užitečná pro ladění při vytváření aplikace. Při chybě u SQL dotazu zobrazí přibližné místo chyby. Mimo jiné se jejím užíváním docílí větší přehlednosti v psaní kódu a zobrazitelnosti SQL dotazu v rámci PHP skriptu.
- `javascript.js` – obsahuje JavaScriptové funkce použitelné pro celou aplikaci

Pro nahrávání obrázků formátu JPEG – konkrétně u loga závodu je zapotřebí, aby adresář `img` a jeho podadresáře měly nastaveno povolení pro zápis. Toto je důležité zvláště při ostrém běhu

aplikace na webhostingovém serveru. Povolení pro zápis se provede například v prohlížeči souborového systému Total Commander změnou atributů u daného adresáře.

Celý administrační systém je vypracován tak, aby bylo možné co nejsnadněji upravovat, vyřazovat z funkce a přidávat další funkční moduly. Soubory, které jsou potřebné pro tyto akce jsou následující:

- menu.php – položky menu
- left.php – jednotlivé podskupiny akcí pro daný modul
- plocha.php – přesměrování toku řízení dle modulu
- modul\_vypis.php – stručný výpis skupiny dat ovládaných patřičným modulem
- req/modul\_form.php – zpravidla prostředí pro vlastní editaci obsahu tabulek v databázi

Ostatní soubory lze chápat jako vnější obálku, která se stará o správné a jednotné zobrazování potřebných informací.

Pokud chceme vyřadit z funkce některý modul, není třeba ve všech výše uvedených souborech mazat a editovat příslušný kód. Pouze v horním menu (menu.php) odstraníme odkaz, pře který se startuje práce s modulem. To je velmi výhodné například pro případ, kdy si nejsme jisti, zda modul nebo kód jemu velmi podobný, nebudeme v budoucnu někdy ještě potřebovat.

Při editaci funkce stávajících a přidávání nových modulů je potřeba upravit všech pět uvedených souborů. Celý systém je ale naprogramován tak, že změny jsou do jisté míry velmi podobné a programátor se může soustředit na vlastní logiku aplikace, tvorbu SQL příkazů a nezdržuje se základními funkcemi o které se starají ostatní skripty.

Z pohledu týmové práce na projektu, který by byl založen na fungování tohoto administračního systému lze říci, že programátoři v projektu zahrnutí jsou schopni se snadno a rychle seznámit s prostředím a mohou brzy vyvíjet potřebný kód pro logicky oddělené moduly.

Použití rámců v administračním systému je zde zvoleno z důvodu načítání obsahu pouze té části stránky(rámu), která se po zaslání dotazu na server změní. Například po smazání nějakého závodu nepotřebujeme, aby se načetlo horní a levé menu, ale aby se změny projevíly v prostředním hlavním rámu, kde bude nyní výpis závodů, které v databázi zůstaly.

## 7. Prezentační část

Tímto termínem se rozumí webová aplikace z pohledu normálního návštěvníka vyvíjené webové aplikace. Ten má možnost si dané fotografie prohlížet, vyhledávat podle startovních čísel, přidávat jednotlivé položky do košíku a provést objednávku. Tato část byla záměrně řešena pouze funkčně a nikoliv graficky. To z toho důvodu, aby se dala aplikace použít pro více designových návrhů prezentační části a byla tak i do jisté míry přenositelná.

Zajímavým prvkem je ochrana proti bezplatnému stahování větších náhledů fotografií ze serveru. Webové prohlížeče poskytují funkci pro stažení obrázků z internetových stránek. Pomocí kliknutí pravého tlačítka myši nad obrázkem se rozbalí nabídka, kde je většinou v anglických verzích možnost „Save Image As...“ Pokud je ale obrázek průhledný, uloží se pouze tento průhledný obrázek. Toho lze využít. Pokud zobrazíme pod průhledným obrázkem požadovaný větší náhled fotografie, uživateli se vše jeví, jako kdyby na stránce byl obrázek pouze jeden. To ale není pravda. Přes pozadí tvořené větším náhledem fotografie (v našem případě starobrno07\_0015.jpg) je několikanásobně nakopírován průhledný gif (v našem případě b.gif). Výsledek je vidět na obr.6.

Toto je řešeno následujícím vygenerovaným HTML kódem:

```
<table border="0" cellpadding="0" cellspacing="5" bgcolor="#ffffff">
  <tr>
    <td background="starobrno07_0015.jpg">
    </td>
  </tr>
</table>
```



(obr.6)

Další foto obrazovek z prezentační části je zbytečné uvádět, protože všechny její funkce už byly popsány výše. V prezentační části nebylo zvoleno použití rámců kvůli adresaci dostupné přes URL. Při použití rámců nejsme schopni zajistit, aby se po zadání jedné URL zobrazila právě požadovaná kombinace obsahu rámců, kterou si přejeme. U administrace toto nevadilo, protože zde není například žádoucí posílání odkazů emailem nebo jinou cestou jiným osobám za účelem zadání URL do prohlížeče a zobrazení požadovaného obsahu. Do administrace musí administrátor vždy vstoupit přes přihlašovací stránku a další akce se dějí na základě navigace pomocí formulářových tlačítek nebo odkazů. Které předají patřičným skriptu vždy požadované parametry v proměnných `$_POST` nebo `$_GET`.

V dalším popisu se zaměřím na proces objednání fotografií.

## 7.1. Implementace prezentační části

Prezentační část je tvořena následujícími skripty:

- `head.php` – vložení všech konfiguračních souborů a skriptů s užitečnými funkcemi. Mimo jiné obsahuje i JavaScriptovou funkci `init()`, která inicializuje povolené vstupní hodnoty formulářových polí při posledním kroku objednávky.
- `menu.php` – jednoduché textové navigační menu
- `bortím.php` – ukončovací tagy html stránek
- `index.php` – úvodní stránka
- `zavody_seznam.php` – seznam všech závodů zanesených v administraci a které mají nastaven zobrazovací příznak na aktivní a u kterých je nastaven příznak nafoceno na Y
- `kam.php` - seznam všech závodů zanesených v administraci a které mají nastaven zobrazovací příznak na aktivní a u kterých je nastaven příznak nafoceno na N
- `cenik.php` – výpis typů zboží a jejich ceny
- `závod.php` – zobrazení stanovišť (formou odkazů na fotografie) vázících se k danému závodu a vyhledávání podle startovních čísel
- `fotografie_seznam.php` – tento skript zajišťuje zobrazení malých náhledů fotografií podle určitých kritérií s možností koupě
- `fotografie.php` – zobrazení většího náhledu fotografie – viz. obr.6
- `kosik.php` – zobrazení obsahu košíku v závislosti na aktuálním stavu `$_SESSION`
- `kosik_akce.php` – vlastní jádro košíku. Jeho funkce již byla popsána výše. Výpis souboru v příloze.

- `adresa.php` – výpis(rekapitulace) položek vložených do košíku navíc s formulářem pro vepsání dat uživatele
- `odeslano.php` – skript potvrzující odeslání objednávky
- `session_destroy.php` – ladící skript pro smazání všech sessions
- `hash.php` – skript pro tvorbu hashovaných uživatelských jmen a hesel do administrační části systému. Očekává jako parametr v URL proměnnou `string2hash`.

## 8. Používání systému

Model vyvíjeného systému pro on-line prodej a třídění fotografií je určen pro zaškoleného provozovatele. Předpokládá se tedy znalost chování systému a jeho možností. V následujících dvou podkapitolách vysvětlím dvě vzorové situace pro provozovatele v administrační části a zákazníka(zaškolení se samozřejmě nepředpokládá).

### 8.1. Nahrání a třídění fotografií

Před samotným nahráním a tříděním fotografií se očekávají fotografická data ve dvou rozlišeních: 180x120px, 300x450px a zhruba 842x560px pro třídění. Takto zmenšené fotografie jsou výstupem nějakého grafického editoru. Práci s ním nezapočítávám do jednotlivých kroků procesu nahrání a třídění. Nicméně neoddelitelně patří k celému procesu. Jako nástroj pro hromadnou úpravu fotografií používám Toner Photo Studio 7. Je schopný fotografie hromadně zmenšovat dle požadované komprese, otáčet, zaostřovat, vylepšovat expozici a provádět další operace s fotografiemi.

Jednotlivé kroky provozovatele (administrátora) v procesu nahrání a třídění fotografií:

1. Provozovatel po přihlášení se do administrace vytvoří událost, ze které byly fotografie pořízeny. Specifikuje přesné jméno adresáře, kam se budou fotky nahrávat. Toto jméno musí být bez mezer a speciálních znaků. Je povoleno používat podtržítka a nezáleží na velikosti písmen.
2. Na lokálním počítači (s kopií databáze) vytvoří adresář s názvem specifikovaným v minulém kroku. Do něj nakopíruje podadresáře `nahledy_male`, `nahledy_vetsi` a source se všemi fotkami v patřičných velikostech specifikovaných výše.
3. Na svém počítači spustí přes administraci formulář třídění a vytvoří tak postupně v tabulce „fotografie“ lokální databáze nové záznamy o zatříděných fotografiích. Takovými záznamy se myslí kód fotografie (uloží se automaticky), id závodu(také se uloží automaticky), id

fotografa, id stanoviště, startovní čísla (pokud je na fotografii zachyceno více startovních čísel, oddělí se tyto čárkou) a v případě, že je na fotografii i závodník s nerozpoznatelným číslem, tak i tento záznam. Na fotografii se může nacházet účastník závodu, jehož číslo nejde přečíst, ale je například zachycen v dobré kompozici a fotografie je zdařilá, zařadí se do nerozpoznatelných čísel i přesto, že na fotce je další závodník s číslem dobře rozpoznatelným. Jinými slovy systém umožňuje začleň fotografie jak mezi startovní čísla, tak mezi nerozpoznatelná čísla zároveň.

4. Nahraje fotky přes FTP na server do požadovaného adresáře „nahledy\_male“ a „nahledy\_vetsi“. Jiné fotografie se na server nenahrávají z důvodu časové a objemové náročnosti. Při předpokladu, že by měla celá aplikace fungovat jen na serveru, musely by se na server nahrát jak fotky pro třídění, tak originály. Pokud má originální fotka datovou velikost cca 2MB a na jedné události se jich vyfotí 1500, znamená to velké množství dat, jejichž nahrávací doba by při běžném připojení k internetu dosahovala několika hodin, což je z hlediska rychlosti aplikace zbytečné zdržení. Aktualizace a udržování konzistence patřičných tabulek databáze na serveru a na lokálním počítači administrátora se tedy jeví jako efektivnější a elegantnější řešení.
5. Provede změnu tabulky „fotografie“ v databázi na serveru nahráním nové verze tabulky z lokálního počítače.
6. Nastaví v administraci příznak závodu jako Nafoceno. V databázi označeno jako Y. Těmito kroky se zajistí nahrání a zatřídění fotografií na webu.

## 8.2. Objednání fotografií

Objednat fotografie může jakýkoliv návštěvník webu bez nutnosti znát přihlašovací jména a hesla. Takovýto model objednávání byl zvolen záměrně s cílem zpřístupnit nákup fotografií bez jakéhokoliv zdržování všem uživatelům internetu.

1. Potenciální zákazník si vyhledá svou požadovanou fotografii podle startovního čísla.
2. Klikne na odkaz „koupit“. Fotografie s patřičným kódem se vloží do nákupního košíku. Nyní se zákazník může rozhodnout zda bude pokračovat dále ve výběru dalších fotografií nebo si vybere produkt (zboží), které si přeje z fotografie vyrobit.
3. Po zadání množství jednotlivých produktů k fotografiím vyplní zákazník svou adresu a kontaktní email. Zvolí také způsob doručení fotografií.
4. Zákazník je po ukončení procesu objednávání informován o správném přijetí objednávky.

## 8.3. Doručení fotografií

U objednávek digitálních fotografií emailem umožňuje systém zjednodušené odeslání fotografií na emailovou adresu zákazníka. Každá objednávka má mimo jiné parametry (představují sloupec tabulky „objednavky“ v databázi) pomocí nichž lze mezi objednávkami pro účely rychlého doručení vyhledávat. Takovými parametry jsou:

- stav zaplacení objednávky
- stav vyřízení objednávky
- zvolený způsob dopravy

Fotografie, které se mají zákazníkům odeslat emailem musí být součástí objednávky, která není ve stavu „vyřízena“ ani „stornována“, je řádně zaplacená a způsob doručení je emailem.

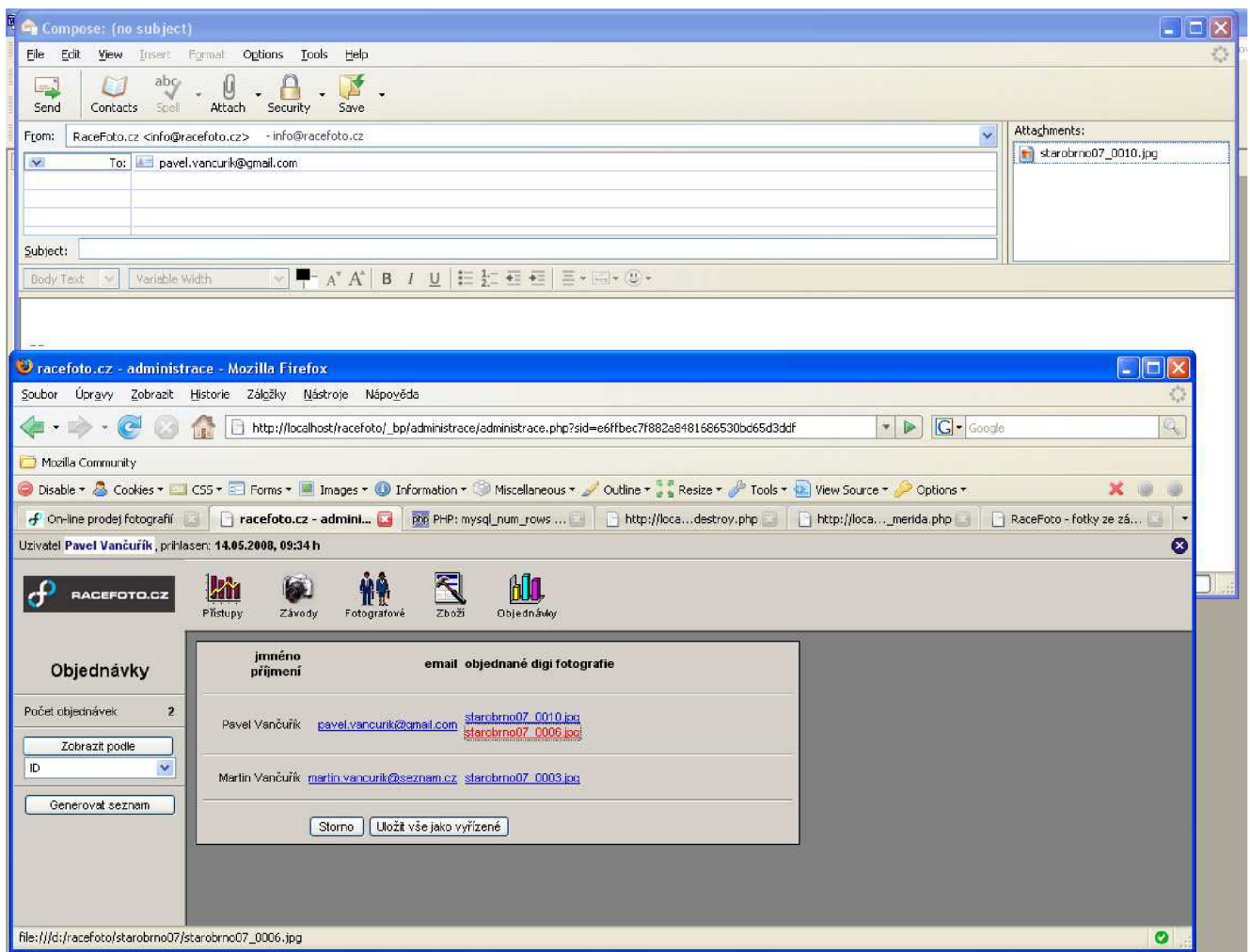
Systém na základě dotazu vyhledá tyto patřičné objednávky a vygeneruje seznam objednávek s důležitými položkami pro odeslání v prostředí administračního systému. Důležitými položkami se myslí jméno a příjmení zákazníka, jeho email a kód(y) objednaných fotografií. Tyto se zobrazují jako odkazy mířící na skutečné umístění systému souborů lokálního počítače z něhož je aktuálně přistupováno do online aplikace administračního rozhraní systému. Podmínkou správného používání tohoto zjednodušeného posílání je nastavení globální proměnné

`$config["base_url_full_img"]` v konfiguračním souboru `config/defaults.php` na adresář s úložištěm fotek. Emailový klient Mozilla Thunderbird umožňuje akci drag and drop nad soubory z lokálního počítače. V praxi to znamená myší přetáhnout odkaz na lokálně umístěný soubor v plné datové kvalitě, který se bude zákazníkovi posílat, do oblasti adres v Mozille Thunderbird. Zákazníkům email se zobrazuje jako odkaz na email. Kliknutím na něj se spustí nové okno zprávy v emailovém klientovi už s přednastavenou emailovou adresou zákazníka. Emailový klient Mozilla Thunderbird musí být nastaven jako výchozí pro práci s elektronickou poštou. Do tohoto nového okna myší přetáhneme požadované odkazy na soubory a klient sám převede tyto odkazy na skutečné soubory, viz. (obr.7). Ve spodní části obrázku jsou vidět vygenerované objednávky se jmény, adresami a soubory fotek. V horní je okno emailového klienta s novou zprávou. Tímto se velice zjednoduší a zrychlí odeslání fotografií v digitálním formátu koncovým zákazníkům emailem. Vše se děje pouze prací s myší a přehledně. Po odeslání emailů potvrdíme formulářovým odesílacím tlačítkem „Uložit vše jako vyřízené“ a tyto objednávky přejdou hromadně do stavu konečného vyřízení.

Dalším možným vylepšením systému by byla možnost napojení obchodu na bankovní účet. Dosavadní řešení předpokládá ruční kontrolu toho, zda platba za požadovanou objednávku na účet dorazila nebo ještě ne. Protože odesílat zákazníkům fotografie emailem bez předchozího zaplacení je prakticky nesmyslné, je indikace stavu zaplacení nezaplacení klíčová v celém procesu. Některé bankovní ústavy dnes nabízejí za příplatek automatické posílání výpisu z účtu ve formátu .txt nebo .csv emailem. Je možné také k těmto datům přistupovat přes speciální url. Systém by mohl být navržen tak, aby vždy za nějaké časové období nebo na popud administrátora zkontroloval dle



variabilních symbolů korespondujících s čísly objednávek došlé platby a automaticky by změnil u zaplacených objednávek stav na „Zaplaceno“. Další kroky by už zůstaly stejné.



(obr.7)

## 9. Bezpečnost

### 9.1. SQL Injection

Asi nejčastějším a nejpoužívanějším útokem hackerů je použití tzv. SQL injection. Je to technika napadení databázové vrstvy programu vsunutím, podstrčením (odtud „injection“) kódu přes neošetřený vstup a vykonání vlastního, samozřejmě pozměněného, SQL dotazu. Toto nechtěné chování vzniká při propojení aplikační vrstvy s databázovou vrstvou (téměř vždy se totiž jedná o dva různé programy). Zabránit se mu dá různě. Jedním ze způsobů je hašování vstupních dat, kdy jsou

data převedena například funkcí md5() na 32 znaků dlouhý řetězec a porovnávat a používat v SQL dotazu až tento hašovaný otisk.

Pomocí SQL injection z URL adresy můžeme přistoupit k datům z databáze. Pokud je hacker schopný vložit do URL, ve které jsou hodnoty parametrů předávány do SQL dotazu takový řetězec, kterým pozmění dotaz, může tak například zobrazit hodně citlivá data z databáze.

Pokud například SQL dotaz vypadá následovně s tím, že \$jmeno je hodnota proměnné z URL,

```
"SELECT * FROM uzivatele WHERE jmeno = '$jmeno';"
```

může hacker podstrčit do URL následující řetězec `a' or 'b'='b'`. Dotaz pak bude vypadat následovně:

```
"SELECT * FROM uzivatele WHERE jmeno = 'a' or 'b'='b';"
```

Výsledek dotazu bude tedy vždy TRUE. Pokud se útočníkovi podaří zjistit název tabulky v databázi nad kterou je dotaz prováděn, může ji dokonce v tomto případě odstranit. Předpokládejme, že útočník zadá v URL do proměnné \$jmeno řetězec `a';DROP TABLE uzivatele; --`, bude dotaz vypadat takto:

```
"SELECT * FROM uzivatele WHERE jmeno = 'a';DROP TABLE uzivatele; --';"
```

Dojde tedy k odstranění tabulky. Proto je dobré pro výsledné chování aplikace na vnějších serverech vypnout zobrazování chybových hlášek SQL.

Často se pomocí SQL injection pokoušejí hackeři dostat na stránky zabezpečené přihlašovacím jménem a heslem. Pro vyzkoušení možnosti napadení různých cvičných webů jsem se v rámci bakalářské práce seznámil s obsahem serveru [www.hackthissite.org](http://www.hackthissite.org), kde jsou pro webové programátory nachystané různé obtížnostní úrovně pro cvičné útoky na zkušební www stránky. Každý se tak má možnost seznámit s nebezpečím, který neopatrně napsaný kód internetové aplikace může způsobit.

Ve vyvíjeném systému je použito přihlašování do administrační části systému. Toto přihlašování je napojeno na databázi a může se zde tedy skrývat nebezpečí potenciálního útoku zlomyslných návštěvníků. Za účelem zvýšení bezpečnosti jsem použil dvojího stupně ochrany a to:

- hašování vstupního řetězce
- odfiltrování nealfanumerických znaků pomocí regulárního výrazu – nahrazením těchto znaků prázdným řetězcem. To udělá kód zobrazený níže.

```
//odstranim z vstupniho pole vsechny nealfanumericke znaky - nahradim je prazdnym znakem
$username = ereg_replace('[^a-zA-Z0-9]', '', $_POST[username]);
$username = md5($username);
$heslo = ereg_replace('[^a-zA-Z0-9]', '', $_POST[heslo]);
```

Funguje pouze za předpokladu, že přihlašovací jména a hesla administrátorů obsahují jen alfanumerické znaky. Tuto podmínku ale sdělí programátor budoucím uživatelům a nemělo by tedy docházet k problémům. Do systému by se tak neměl dostat nepovolený řetězec.

## 10. Testování a úpravy aplikace

Starší verze navrhované aplikace je testována v reálném provozu. Všechny podněty, které mě vedly k rozhodnutí vytvořit tento software tak, jak uvádím v této práci, vychází ze zkušeností s nasazením hodně podobné verze v reálném provozu. Z provozního hlediska je velice důležitý čas, ve kterém je administrátor schopen dodané fotografie zpracovat a vystavit na webový server. Z monitorování fungování několika firem v České republice, které se zabývají touto problematikou je jasné, že zatím se nikomu nepodařilo hromadně zpracovat velké množství fotografických dat včetně seřazení dle startovních čísel společně s nahráním na servery a zpřístupněním potenciálním zákazníkům v době do cca 2 hodin od skončení fotografické události. V rychlosti zpracování a využití dalšího software usnadňující rozpoznávání čísel z obrazu jsou tedy možnosti, které by byly jistě velice zajímavé nejen z pohledu implementace.

Nedílnou součástí systému by také v budoucnu mohla být jakási zpětná vazba pro fotografy. Potřebují vědět, které fotky si lidé nejvíce objednávají a o které mají zájem. Export v patřičném formátu přístupný pro administrátora i fotografy celou aplikaci mohl také posunout o krok dopředu. Z hlediska monitorování chování zákazníků se také naskytá otázka: Kupují si všechny fotografie se svým startovním číslem nebo jen vybrané (dle jejich názoru nejlepší)? V poslední době začíná stále více přibývat lidí, kteří si přejí fotografii v digitální podobě. Tuto si mohou sami nechat vyvolat ve fotolabu na patřičný formát papírové fotografie. Na druhou stranu ale fotografie dodaná zákazníkovi v papírové podobě přímo od fotografické firmy, na jejichž webu si zákazník fotku objednal, s sebou nese jistotu dobrého zpracování pramenící ve spolupráci fotolabu s dostatečně kvalifikovanými pracovníky a fotografické společnosti.

Odměňování fotografií na základě prodaných fotografií by jistě bylo pro některé fotografy motivační. Toto systém může v budoucnu umožňovat a je na to již nyní do značné míry připraven. Ke každé fotografii je přiřazen autor fotografie. Podle těchto kritérií lze tedy přesně určit kolik kterých ( a v jakém zboží) fotografií s prodá od daného fotografa. Jistě by bylo nutné tyto výpisy časově omezit z důvodu průběžného vyplácení mezd zaměstnancům, ale principiálně to bude asi přínosem pro obě strany. Systém by mohl umožňovat exportovaný výpis automaticky posílat emailem fotografům např. automaticky jednou za časové období nebo na podnět administrátora.

Zamezení stahování větších náhledů bylo docíleno pomocí HTML ochrany, ale jistě toto není pro mnohé uživatele internetu velká překážka si danou fotografii stáhnout. Při existenci možnosti

uchovat a následně použít aktuální obraz obrazovky (Print Screen) se ale stejně většina podobných způsobů omezení nelegálního stahování bude minout účinkem.

Věcí nekonečné diskuze by také mohly být velikosti zobrazovaných náhledů. Je potřeba zobrazit takové náhledy podle kterých si uživatel dokáže fotografii vybrat, poznat se na ní a zhruba zhodnotit její kvalitu, jestli mu stojí za to investovat do její koupě nebo ne. Na druhou stranu ale je nutné celý proces nahrávání a zobrazování fotografií co nejvíce zrychlit. Jsou tu tedy dva protipóly, jejímž střetem je datová velikost zobrazovaných náhledů fotografií.

Proces napojení na bankovní účet by dle informací jednoho bankovního ústavu měl být jednoduchý, ale zatím ho banka neposkytuje zdarma. Identifikaci plateb tedy zatím provádí správce systému dle informací z online výpisu elektronického bankovníctví poskytnutým bankou zpravidla zdarma v rámci ceny vedení účtu.

Aplikace nezobrazuje EXIF informace o vystavovaných fotografiích ze dvou důvodů. Prvním je snaha o dosažení co nejmenší datové velikosti náhledů a druhým je zamezení dozvědění se o tělech, objektivěch a režimech fotoaparátů s kterými fotografové fotí.

Celkové zhodnocení činnosti fotografických společností závislých na software podobný tomu, který byl vyvíjen v rámci bakalářské práce, se dá zhodnotit jako pozitivní a poskytuje zákazníkům možnost koupě fotek pořízených profesionálním náčiním a samotnými zkušenými fotografy. Ne každý je schopen si podobnou fotografii nechat vyfotit od kamaráda nebo rodinného příslušníka. V dnešní době, kdy se platí za čím dál tím více služeb, je tato práce logickým vyústěním dnešní doby a doplňuje nabídku dosud stávajících fotografických služeb.

Mezi rozhodné okolnosti vedoucí k vítězství při konkurenčním boji takovýchto fotografických firem bude jistě i kvalita software, skrz který se fotografie prodávají a prezentují zákazníkům na internetu. Očekávání a úspěšné zapojování nových technologií do celého procesu zpracování a třídění fotografií bude vést určitě ke zlepšení a zvýšení rychlosti poskytovaných služeb.

# 11. Závěr

Byl vytvořen funkční model systému pro on-line prodej fotografií. V systému jsou implementovány operace pro zpracování velkého množství fotografických dat. Určité akce nejsou zatím kvůli důrazu na rychlost vystavení fotografií na web plně v kompetenci administrační části systému. Nicméně tyto operace byly minimalizovány a přes administraci se provádí většina prací. Tato bakalářská práce pro mě měla zvláštní přínos v zamyšlení a pokusu realizace podobných projektů. Možné pokračování projektu je ve větším přesunutí požadovaných operací do administrační části systému při stejné rychlosti. Zároveň spolupráce například s firmami zabývajícími se měřením časů podle čipů by byla velice zajímavá.

# Literatura a prameny

- [1] J.Castagnetto, H.Rawatt, S.Schumann, C. Scollo,D.Valiath: Programujeme PHP profesionálně - Computer Press, 2002
- [2] P.Mikle – Referenční příručka DHTML: UNIS Publishing, 2001
- [3] L.Welling, L.Thomsonová.: PHP a MySQL - rozvoj webových aplikací - 2. vydání, SOFTPRESS, 2003.
- [4] Williams, H.E., Lane, D.: PHP a MySQL - Vytváříme webové databázové aplikace - Computer Press. 2002.
- [5] [www.php.net](http://www.php.net)
- [6] [www.jakpsatweb.cz](http://www.jakpsatweb.cz)
- [7] <http://www.sportis.cz> – čipové technologie
- [8] <http://www.hackthissite.org>

# Seznam příloh

Příloha 1. Zdrojový text souboru kosik\_akce.php

Příloha 2. CD s kompletními zdrojovými texty a exportem databáze