

Sem vložte zadání Vaší práce.



ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE  
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ  
KATEDRA SOFTWAREVÉHO INŽENÝRSTVÍ



Diplomová práce

# **Automatizace interakcí na sociální síti Facebook**

***Bc. Jakub Škvára***

Vedoucí práce: Ing. Jaroslav Kuchař

5. května 2013



---

## Poděkování

Chtěl bych poděkovat svému vedoucímu Ing. Jaroslavu Kuchařovi, za ochotu a pomoc při psaní této diplomové práce.



---

## Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o etické přípravě vysokoškolských závěrečných prací.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů, zejména skutečnost, že České vysoké učení technické v Praze má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona.

V Praze dne 5. května 2013

.....

České vysoké učení technické v Praze

Fakulta informačních technologií

© 2013 Jakub Škvára. Všechna práva vyhrazena.

*Tato práce vznikla jako školní dílo na Českém vysokém učení technickém v Praze, Fakultě informačních technologií. Práce je chráněna právními předpisy a mezinárodními úmluvami o právu autorském a právech souvisejících s právem autorským. K jejímu užití, s výjimkou bezúplatných zákonných licencí, je nezbytný souhlas autora.*

## **Odkaz na tuto práci**

Škvára, Jakub. *Automatizace interakcí na sociální síti Facebook*. Diplomová práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2013.



---

# Abstract

Online Social Networks (OSNs) have become an integral part of today's Web. Politicians, celebrities, revolutionists, and others use OSNs as a podium to deliver their message to millions of active web users. Unfortunately, in the wrong hands, OSNs can be used to run astroturf campaigns to spread misinformation and propaganda. Such campaigns usually start off by infiltrating a targeted OSN on a large scale. In this paper, we evaluate how vulnerable OSNs are to a large-scale infiltration by socialbots: computer programs that control OSN accounts and mimic real users. We adopt a traditional web-based botnet design and built a Socialbot Network (SbN): a group of adaptive socialbots that are orchestrated in a command-and-control fashion. We operated such an SbN on Facebook—a 750 million user OSN—for about 8 weeks. We collected data related to users' behavior in response to a large-scale infiltration where socialbots were used to connect to a large number of Facebook users. Our results show that (1) OSNs, such as Facebook, can be infiltrated with a success rate of up to 80%, (2) depending on users' privacy settings, a successful infiltration can result in privacy breaches where even more users' data are exposed when compared to a purely public access, and (3) in practice, OSN security defenses, such as the Facebook Immune System, are not effective enough in detecting or stopping a large-scale infiltration as it occurs.

**Keywords** Nahradte seznamem klíčových slov v angličtině oddělených čárkou.

---

# Abstrakt

V několika větách shrňte obsah a přínos této práce v češtině. Po přečtení abstraktu by se čtenář měl mít čtenář dost informací pro rozhodnutí, zda chce Vaši práci číst.

**Klíčová slova** Nahradte seznamem klíčových slov v češtině oddělených čárkou.

---

# Obsah

<b>1</b>	<b>Úvod</b>	<b>1</b>
1.1	Stručný popis facebooku . . . . .	1
1.2	Motivace . . . . .	2
1.3	Stručný popis experimentu . . . . .	2
1.4	Přínos . . . . .	3
1.5	Etické hledisko experimentu . . . . .	3
<b>2</b>	<b>Analýza</b>	<b>5</b>
2.1	Možností interakcí s facebookem . . . . .	5
2.2	Možnosti ovládání webového rozhraní facebooku . . . . .	8
2.3	Detailní popis programu Selenium . . . . .	11
2.4	Výhody a nevýhody použití programu Selenium WebDriver . . . . .	15
2.5	Využití aplikace . . . . .	17
<b>3</b>	<b>Implementace</b>	<b>19</b>
3.1	Požadavky na aplikaci . . . . .	19
3.2	Popis uživatelského rozhraní výsledné aplikace . . . . .	20
3.3	Volba technologií pro aplikaci . . . . .	23
3.4	Architektura celé aplikace . . . . .	26
3.5	Rozdělení aplikace . . . . .	27
3.6	Stahování informací z facebookových profilů . . . . .	27
3.7	Problémy při implementaci . . . . .	32
3.8	Nahrávání dat na facebookové profily . . . . .	36
3.9	Části aplikace sloužící pro získání dat pro statusy . . . . .	39
3.10	Běh aplikace . . . . .	40
<b>4</b>	<b>Průběh experimentu</b>	<b>43</b>

4.1	První generace facebookových účtů . . . . .	43
4.2	Příprava experimentu . . . . .	43
4.3	Problémy se zakládáním facebookových účtů . . . . .	47
4.4	Skupiny facebookových účtů pro experiment . . . . .	47
<b>5</b>	<b>Závěr</b>	<b>63</b>
5.1	Možné způsoby rozšíření a pokračování v experimentu . . . .	63
5.2	Vyhodnocení . . . . .	64
5.3	Možnosti pokračování experimentu . . . . .	64
	<b>Literatura</b>	<b>67</b>
	<b>A Slovník použitých pojmů a zkratk</b>	<b>69</b>
	<b>B Obsah přiloženého CD</b>	<b>71</b>

---

## Seznam obrázků

2.1	Schéma přihlašování k facebooku pomocí OAuth . . . . .	7
3.1	Uživatelské rozhraní výsledné aplikace . . . . .	20
3.2	Přiřazování uložených uživatelů k jednotlivým skupinám . . . . .	21
3.3	Formulář pro přidávání nových úkolů v aplikaci . . . . .	22
3.4	Přehled připravených úkolů v uživatelském rozhraní aplikace . . . . .	23
3.5	Přehled vztahů mezi třídami presenteru a modelu . . . . .	27
3.6	Časový rozdíl u atributů title a data-utime u elementů zobrazujících čas na facebooku . . . . .	34
3.7	Uživatelské rozhraní nahrávání úkolů na facebookové profily . . . . .	38
3.8	Přehled ukládaných dat do jednotlivých kolekcí v databázi MongoDB . . . . .	39
3.9	Přehled ukládaných dat do jednotlivých tabulek v databázi MySQL . . . . .	40
4.1	Zadávání telefonního čísla pro ověření na facebooku . . . . .	44
4.2	Potvrzení zda se znají osoby v reálu při zasílání žádosti o přátelství na facebooku . . . . .	53
4.3	Zrušení žádosti o přátelství na facebooku . . . . .	53
4.4	Formulář na facebooku nabádající k vyplnění vystudované střední školy . . . . .	56
4.5	Formulář na facebooku pro zrušení nepřijatých a nepotvrzených žádostí o přátelství . . . . .	57
4.6	Informace o dosažení maximálního počtu nevyřízených žádostí o přátelství na facebooku . . . . .	58
4.7	Formulář pro ověření totožnosti zadáním data narození na facebooku . . . . .	59
4.8	Facebookové upozornění kvůli příliš rychlé interakci . . . . .	60



## Úvod

Facebook je v dnešní době největší sociální síť na světě. Podle posledních oficiálních výsledků [5] za první čtvrtletí roku 2013 se každý den na facebook přihlásí 665 milionu uživatelů a každý měsíc dokonce 1,11 miliard uživatelů. Celkový počet obyvatel země je nyní přibližně 7,1 miliardy, pokud by facebook byl jeden ze států, jednalo by se o třetí nejlidnatější zemi ihned po Číně a Indii [14].

### 1.1 Stručný popis facebooku

Sociální síť facebook slouží k udržování kontaktu s vašimi kamarády pomocí internetu. Nejdůležitější funkce facebooku jsou: zasílání statusů (krátkých zpráv) na svoji virtuální zeď, přidávání fotek s možností označovat své kamarády a zasílání soukromých zpráv.

Facebook byl spuštěn v únoru roku 2004 na univerzitě Harvard studentem Markem Zuckerbergem. Od té doby se tato sociální síť stala celosvětovým fenoménem. S oblibou této sociální sítě se ale pojí i problémy s jejím používáním. Největší kritiku facebook sklízí za problémy se soukromím uživatelů. Pokud si totiž na facebooku přidáte do přátel jakýkoliv účet, má přístup ke všem statusům a fotkám, které jste na tuto sociální síť nahráli. Vše jde změnit v nastavení, ale z uživatelského hlediska to není zcela jednoduché a většina uživatelů si nechává na svém profilu výchozí nastavení.

### 1.2 Motivace

Kvůli těmto bezpečnostním rizikům na facebooku vzniknul na Univerzitě Britské Kolumbie zajímavý výzkum s názvem: The Socialbot Network: When Bots Socialize for Fame and Money [14]. Jedná se o experiment, ve kterém pomocí takzvaných **socialbotů** ovládají facebookové účty aby se chovaly jako reální uživatelé. Cílem bylo zjistit, kolik procent účtů si přidá do svých přátel na facebooku účty ovládané socialboty. Dalším cílem práce, bylo zjistit, kolik osobních informací jsou o sobě uživatelé pomocí facebooku sdělit.

Tento experiment mne velmi zaujal a rozhodl jsem se, že by mohlo být velmi zajímavé provést podobný experiment se zaměřením na facebookové účty z České republiky. Záměrem nebylo experiment zcela zkopírovat, ale vyzkoušet podobný test s rozdílným přístupem.

### 1.3 Stručný popis experimentu

Jako součást diplomové práce jsem tedy navrhnul a naprogramoval aplikaci pro ovládání facebookových účtů, podobně jako v experimentu zmíněném výše. Aplikace umožňuje především přidávat statusy i nové přátele na facebooku, stahovat data ze zdi a provádět další akce důležité pro zachování aktivity profilu. Při implementaci jsem narazil na některé problémy se stahováním dat z facebooku, ale nakonec se mi vše podařilo vyřešit.

Poté, co byla aplikace hotová jsem založil na facebooku 25 experimentálních účtů, které byly ovládány touto aplikací. Při každém přihlášení účty simulovaly reálné chování facebookových uživatelů. Během experimentu byly na profily experimentálních účtů nahrávány statusy přidávání přátel a byla stahována dostupná data z cizích profilů. Tento experiment běžel spuštěný přibližně měsíc a za tu dobu bylo nasbíráno poměrně velké množství facebookových přátel a staženo velké množství dat. Výsledky celého experimentu jsou popsány dále.

Podle statistik je na facebooku 3,8 milionu účtů z České republiky [1]. Přišlo mi tedy zajímavé zaměřit se na tuto skupinu a celý experiment více lokalizovat. Mezi hlavní důvody tohoto rozhodnutí patřila především znalost českého facebookového prostředí (aktivní stránky a aktuální trendy), zamezení jazykovým bariérám mezi účty a větší důvěryhodnost profilů. Všechna komunikace na facebookových profilech proto probíhala v českém jazyce.



Existuje velké množství internetových sociálních sítí, kromě facebooku například twitter, google+, linkedIn a další. Facebook jsem pro experiment vybral ze dvou důležitých důvodů. Facebook má nejvíce uživatelů ze všech existujících sociálních sítí a tím pádem je i celosvětově nejznámější. Druhým důvodem je to, že na facebooku je mnoho funkcí pro komunikaci a interakci mezi jednotlivými uživateli, jedná se například o posílání statusů, nahrávání fotek, zakládání událostí, vytváření skupin a dále.

Je potřeba zmínit, že celý popis návrhu a provedení experimentu je platný pro první polovinu roku 2013. Facebook velmi často mění uživatelský vzhled, nastavení a funkčnost, takže je možné, že některé z popisovaných funkcí anebo postupů se mohou změnit nebo zastarat. Při vývoji aplikace jsem zkoumal funkčnost a navrhoval vše s ohledem na aktuální možnosti. Aktuální chování jednotlivých funkcí je vždy podrobně popsáno, pokud by se v budoucnu měnilo. V průběhu experimentu se mi stalo, že facebook změnil pro vybrané účty vzhled některých stránek.

## 1.4 Přínos

Hlavním cílem práce bylo zjistit, v jaké míře jsou uživatelé facebooku ochotni přijímat žádosti o přátelství od zcela cizích profilů a kolik přátel na facebooku lze pomocí automaticky ovládaných účtů získat. [todo nasbirana data]

Výsledky experimentu také odhalily, jaké jsou nejlepší strategie pro přidávání nových přátel na facebooku a na které faktory je dobré se zaměřit. Naproti tomu jsem narazil i na pár zádrhelů se kterými je potřeba počítat a vyvarovat se jich.

## 1.5 Etické hledisko experimentu

Celá aplikace byla při experimentu využívána k ovládnutí nereálných facebookových účtů. Potřeboval jsem totiž získat odpovídající množství profilů a založení bylo nejjednodušším způsobem získání. Aplikace by ale šla použít i pro ovládnutí facebookových účtů jakýchkoli osob. Do aplikace stačí zadat pouze přihlašovací údaje a id profilu na facebooku.

Při návrhu celého experimentu jsem dbal na co nejnížší možné riziko všechna stažená data byla šifrována a anonymizována, aby nomhla být

## 1. ÚVOD

---

zneužita. Po ukončení a vyhodnocení experimentu byla všechna získaná data smazána.

Tento výzkum nemá v žádném případě nabádat k zakládání a ovládání účtů na facebooku. Naopak díky tomuto experimentu půjde jednodušeji odhalit účty s podezřelým chováním na internetu a bude jednodušší je eliminovat.

---

## Analýza

Před začátkem celého experimentu jsem prošel různé možnosti, kterými by šel experiment nejlépe realizovat. Zjišťoval jsem možnosti a omezení dostupných nástrojů a sledoval jsem výhody a nevýhody možných řešení.

### 2.1 Možností interakcí s facebookem

#### 2.1.1 Rozhraní internetové stránky facebook

Pro interakci s facebookem používá většina uživatelů internetové rozhraní, které je na webové adrese `www.facebook.com`. Toto rozhraní je upraveno pomocí techniky takzvaného responzivního designu<sup>1</sup> pro různé typy zařízení, takže se zobrazuje jedno rozhraní stránky pro monitory osobních počítačů s velkým rozlišením a odlišný design pro chytré telefony a tablety s malými displeji a s nízkým rozlišením. Většina uživatelů je na toto rozhraní zvyklá a to i přesto, že jej facebook často drasticky mění bez možnosti návratu ke starší verzi designu.

Dříve se kvůli změnám rozhraní nejednou zvedla velká vlna nevole (například změna osobních profilů v roce 2009 [10], změna zobrazení facebookové zdi v roce 2010 [11], zavedení timeline v roce 2011 [13] anebo přepracování facebookových zpráv taktéž v roce 2011 [12]). Nyní se kvůli těmto problémům facebook snaží oznamovat předem veřejnosti výhody a nové možnosti úprav designu a zapíná nové funkce postupně s možností dočasně používat starou verzi.

---

<sup>1</sup>[http://en.wikipedia.org/wiki/Responsive\\_web\\_design](http://en.wikipedia.org/wiki/Responsive_web_design)

### 2.1.2 Facebook Graph API

Druhou možností, jak používat facebook jsou aplikace. Většinou se jedná o aplikace pro chytré telefony, tablety a další zařízení. Tyto aplikace komunikují s facebookem pomocí takzvaného **Graph API** [2] Jedná se o RESTové webové rozhraní.

Přihlašování ke Graph API se provádí následujícím způsobem. Každá aplikace má přiřazen svůj vlastní veřejný klíč (API Key), sloužící jako jednoznačný identifikátor (ID) aplikace a dále tajný neveřejný klíč (takzvaný APP Secret), pomocí kterého se k aplikaci přihlašuje. Každý uživatel, který chce aplikaci používat, musí danou aplikaci schválit na svém facebookovém profilu a povolit oprávnění, která data může aplikace číst a měnit.

Celý proces přihlašování k facebooku používá OAuth autorizaci<sup>2,3</sup>, při které je uživatel aplikace přesměrován na stránky facebooku, zde se musí uživatel přihlásit a při prvním použití aplikace povolit přístup dané aplikaci k uživatelskému profilu a datům na facebooku. Poté je uživatel přesměrován na naše stránky a aplikace dostane token, pomocí kterého se identifikuje při každé další komunikaci s REST rozhraním (jak je vidět na obrázku 2.1). Podrobnější popis přihlašování se všemi detaily je popsán na stránkách facebooku v dokumentaci (<https://developers.facebook.com/docs/concepts/login/login-architecture/>).

Existují různé knihovny pro práci s facebook Graph API. Tyto knihovny se liší hlavně použitou platformou (Google Android, Apple iOS, ...) a programovacím jazykem (serverové: PHP, Java, apod., klientské: JavaScript). Některé z těchto knihoven mohou mít přihlašovací proces kvůli omezením mírně odlišný (například JavaScript), ale hlavní princip zůstává vždy stejný.

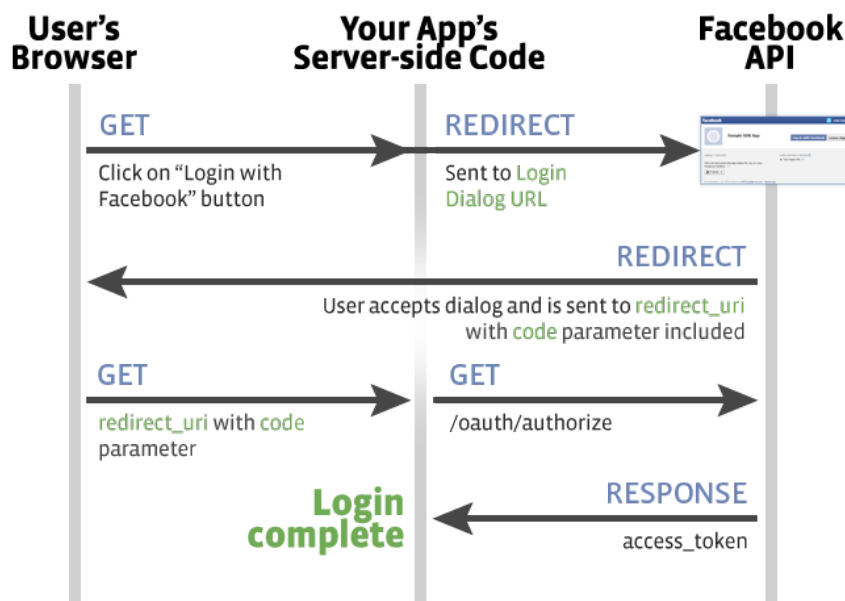
Nemusí se ale jednat pouze o aplikace pro interakci s facebookem, velké množství programů (například hry, kvízy a další) data z facebooku pouze čtou a získávají informace o přátelích uživatele a podobně. Takovéto aplikace jsou pak zobrazeny v okně facebooku (jsou vloženy do stránky pomocí HTML tagu iframe) a mohou používat facebookové prvky (tlačítka, vyskakovací okna, apod.<sup>4</sup>), takže méně zkušený uživatelé si ani nemusí uvědomit, že používají aplikaci, která není vyvíjena facebookem.

---

<sup>2</sup><http://oauth.net/>

<sup>3</sup><https://developers.facebook.com/docs/howtos/login/server-side-login/>

<sup>4</sup><https://developers.facebook.com/docs/plugins/>



Obrázek 2.1: Schéma přihlašování k facebooku pomocí OAuth

(<https://developers.facebook.com/docs/concepts/login/login-architecture/>)

Facebook nepodporuje žádnou jinou možnost, jak aplikaci ovládat kromě Graph API. Kvůli častému zneužívání tohoto API byl ale facebook postupem času nucen některé funkce tohoto rozhraní omezit nebo úplně zakázat. Jedná se hlavně o možnost dát skupině nebo stránce na facebooku like anebo přidat si přátele. Tato funkce API byla ještě spolu s takzvaným likejackingem<sup>5</sup> často zneužívána k podvodnému navyšování fanoušků facebookových skupin.

### 2.1.3 Omezení facebook Graph API

Facebook proto zrušil možnost dávat stránkám like skrze Graph API. S dalším zneužíváním funkcí facebook API se může stát, že budou dále zakázány i

<sup>5</sup>Variace clickjacingu, pomocí HTML, CSS anebo JavaScriptu je na stránce skryto facebookové tlačítko like a je překryto jiným tlačítkem, uživatel pak po kliknutí na viditelné tlačítko nevědomky aktivuje i skryté facebookové like tlačítko <http://en.wikipedia.org/wiki/Clickjacking#Likejacking>

## 2. ANALÝZA

---

některé další funkce. Využití facebook API je tedy poněkud omezené a s nejistou budoucností. Hlavní nedostatky facebook Graph API pro využití pro experimentální aplikaci jsou:

- nemožnost dát like skupině
- nemožnost přidat si nové přátele
- omezený výpis informací o osobě oproti webovému rozhraní
- časem mohou být zakázány nebo omezeny některé důležité funkce

Tyto nedostatky Graph API mi nedovolovaly jej použít pro implementaci aplikace. Kvůli těmto omezením jsem si tedy vybral raději možnost obsluhovat facebook pomocí webového rozhraní za použití specializovaných programů a nástrojů k tomu určených.

## 2.2 Možnosti ovládání webového rozhraní facebooku

Po zamítnutí použití Graph API pro aplikaci jsem se zabýval dalšími možnostmi ovládání facebooku. Podrobněji jsem se zaměřil na jedno nízkoúrovňové a jedno pokročilejší řešení. První možností je využití protokolu HTTP. Druhou technologií na kterou jsem se zaměřil byly programy pro simulování a ovládání webových prohlížečů.

### 2.2.1 Výhody a nevýhody využití HTTP protokolu pro komunikaci s facebookem

Jako první možnost ovládání webového rozhraní facebooku se nabízelo simulovat webový prohlížeč pomocí HTTP požadavků. Toto řešení je poměrně přímočaré a webový protokol HTTP je velmi dobře zdokumentovaný a jednoduše použitelný, později jsem ale tuto možnost zavrhnul, jelikož bych musel znovu implementovat již mnohokrát hotové řešení, kde bych musel řešit nízkoúrovňové problémy HTTP komunikace.

Použití HTTP protokolu by se nejvíce vyplatilo u statických webových stránek. Největším problémem tohoto přístupu je ale skutečnost, že face-

book používá velké množství AJAXových<sup>6</sup> požadavků a dynamických JavaScriptových oken, které by mohly způsobit komplikace.

Ztížilo by to především získávání informací z uživatelských profilů a podobně. Například pro zobrazení příspěvků na zdi uživatele za určité období, je potřeba dosáhnout konce stránky, kde se musí kliknout na tlačítko načíst další příspěvky a tento postup se opakuje, do té doby, než dosáhneme požadovaného množství nebo stáří příspěvků, stejný způsob zobrazování dalších dat se na facebooku používá i na dalších stránkách.

Mezi další nevýhody používání HTTP komunikace patří to, že facebook pro některé části webových stránek generuje odlišný HTML kód (mění se například atributy `id` u HTML značek a podobně). Některé formulářové prvky pak mají jiné názvy a musel by se při každém požadavku zpracovávat HTML kód s formulářovými elementy.

Nejvíce jsem se s těmito změnami setkal na hlavní stránce facebooku, kde pravděpodobně probíhá takzvané A/B<sup>7</sup> uživatelské testování stránek. To znamená, že se uživatelům zobrazují různé grafické verze stránky a následně se měří rozdíly úspěšnosti konverzního poměru měřených veličin (například: počet zobrazení stránky / počet nákupů) pro jednotlivé verze.

V poslední době se začal rozšiřovat nový internetový protokol nazvaný SPDY<sup>8</sup>, který slouží jako nadstavba nad tradičním HTTP protokolem pro rychlejší načítání internetových stránek. Facebook tento protokol již na svých serverech používá a mohly by tak vzniknout problémy při HTTP komunikaci i když by měl být protokol SPDY teoreticky zpětně kompatibilní, nemuselo by vše být úplně bez problémů.

Kvůli těmto problémům spojeným s používáním HTTP protokolu jsem se tedy rozhodnul využít raději některé z hotových řešení a zabývat se komunikací s webovými stránkami facebooku na vyšší úrovni.

---

<sup>6</sup>AJAX - asynchronní načítání dynamických částí webových stránek pomocí JavaScriptu (viz [http://en.wikipedia.org/wiki/Ajax\\_\(programming\)](http://en.wikipedia.org/wiki/Ajax_(programming)))

<sup>7</sup>[http://en.wikipedia.org/wiki/A/B\\_testing](http://en.wikipedia.org/wiki/A/B_testing)

<sup>8</sup><http://en.wikipedia.org/wiki/SPDY>

### 2.2.2 Programy simulující chování webového prohlížeče

Jako nejjednodušší z možností ovládání webového rozhraní facebookové stránky se naskytlo využití programů, které simulují internetový prohlížeč. Existuje celá škála takovýchto programů. Téměř pro každý programovací jazyk existuje některý program, který dokáže zpracovávat webové stránky. Rozdíl mezi těmito programy je především v počtu a kvalitě implementovaných webových specifikací a funkcionalit.

Programy pro zobrazování webových stránek se liší především implementovaným vykreslovacím jádrem webového prohlížeče. V následujícím seznamu je zobrazen přehled nejpoužívanějších jader pro vykreslování webových stránek.

- WebKit<sup>9</sup> - Apple Safari
- Blink<sup>10</sup> - Google Chrome
- Gecko<sup>11</sup> - Mozilla Firefox
- Trident<sup>12</sup> - Microsoft Internet Explorer

Kromě těchto nejpoužívanějších vykreslovacích jader, které vyžívají nejznámější internetové prohlížeče, existuje i velké množství méně známých jader a v podstatě každý si může implementovat vlastní vykreslovací jádro pro webové stránky.

Vykreslovací jádro ovlivňuje to, jak bude webová stránka vypadat a jak bude interagovat. Každé webové jádro má své specifické chování a způsob vykreslování internetových stránek. Mohou se lišit způsoby vykreslení stejné webové stránky mezi jednotlivými verzemi stejného jádra. Existují i speciálně upravená jádra pro vykreslování internetových stránek na mobilních přístrojích, které často naschvál neimplementují některé pokročilejší

---

<sup>9</sup>WebKit - <http://www.webkit.org/>

<sup>10</sup>Společnost Google starající se o prohlížeč Chrome v nedávné době oznámila, že již nadále nebude používat opensource jádro WebKit, ale vytvoří vlastní fork tohoto jádra s názvem Blink - [http://en.wikipedia.org/wiki/Blink\\_\(layout\\_engine\)](http://en.wikipedia.org/wiki/Blink_(layout_engine))

<sup>11</sup>Gecko - <https://developer.mozilla.org/en-US/docs/Mozilla/Gecko>

<sup>12</sup>Trident - [http://en.wikipedia.org/wiki/Trident\\_\(layout\\_engine\)](http://en.wikipedia.org/wiki/Trident_(layout_engine))



funkce kvůli hardwarovým nárokům mobilních zařízení a rychlosti vykreslování celé stránky.

Druhým důležitým parametrem programů pro práci s webovým rozhraním internetových stránek je to, zda se webové stránky vykreslují jen v paměti počítače, nebo zda se výsledek vykresluje i na obrazovku. Existují programy, určené především pro testování webových stránek, které pouze kontrolují výskyt a pozici jednotlivých elementů, ale nedokáží stránku vykreslit, mají vše uloženo pouze v paměti. Díky tomu jsou ale i rychlejší a jsou i implementačně jednodušší.

Různé programy pro zpracování webových stránek jsou zobrazeny v tabulce 2.1. Jedním z nejpoužívanějších programů je pravděpodobně Selenium. Používá se především pro automatické testování webových stránek a zdrojový kód je napsán v programovacím jazyce Java. Dalším používaným programem je PhantomJs. Jak už název napovídá, je tento program napsán v jazyce JavaScript. Hojně se používá pro vykreslování dynamických částí webového rozhraní internetových stránek přímo na serveru. Je to užitečné především pokud chceme poslat webové stránky pro vyhledávací roboty, kteří mají problémy se zpracováním JavaScriptu. Další programy uvedené v tabulce slouží především pro testování výstupu webových stránek a liší se především programovacím jazykem pro který je program určený a rozhraním pro práci s programem.

Tabulka 2.1: Příklady programů pro zpracování webových stránek

Název	Jádro	Programovací jazyk/API	Zpracování
Selenium	Gecko	Java, C#, Ruby, Python, PHP, Perl, Haskell, REST API	vykreslení
PhantomJs	WebKit	JavaScript	paměť
WebInject	vlastní	XML API	paměť
HtmlUnit	vlastní	Java	paměť

## 2.3 Detailní popis programu Selenium

Programů zpracovávajících webové stránky je velké množství. Jeden z nejpoužívanějších nástrojů se nazývá Selenium. Původně tento program vznikl jako nadstavba webového prohlížeče Mozilla Firefox. Pomocí nainstalovaného

## 2. ANALÝZA

---

pluginu v tomto prohlížeči jej lze ovládat přes programové API, které poskytuje program Selenium.

Jedná se o Javový `.jar` soubor, který se stará o spouštění a ovládání webového prohlížeče. API programu Selenium je nezávislé na programovacím jazyku a tak mohly vzniknout knihovny pro velké množství těchto jazyků. Selenium používá webové RESTové API, takže se dají na internetu najít i neoficiální knihovny pro různé další jazyky, které nejsou podporovány přímo od výrobce tohoto programu. Dnes již existují k programu Selenium i pluginy, které umožňují místo prohlížeče Mozilla Firefox používat prohlížeče Google Chrome, Opera, PhantomJS a další.

Hlavní výhoda programu Selenium spočívá v tom, že jsou webové stránky renderovány ve skutečném internetovém prohlížeči a operace, které se v prohlížeči odehrávají vidíme v reálném čase. V případě potřeby je možné převzít ruční kontrolu nad stránkou a pokračovat v interakci s webovou stránkou manuálně.

S použitím reálného prohlížeče se pojí i výhoda optimalizace webových stránek facebooku. Stránky jsou pro reálný prohlížeč připravovány a zcela jistě jsou v něm facebookem důkladně testovány pro správné zobrazování a fungování všech částí. V případě nefunkčnosti by facebook všechny nedostatky co nejdříve napravil a určitě by velmi rychle reagoval na nahlášené problémy. (Facebook o některých nahlášených problémech ví, ale kvůli tomu, že pro něj nejsou natolik důležité se s nimi nezabývá, například nefunkčnost URL pro přidání podmínek pro používání facebookové aplikace nahlášena v prosinci 2012 - <https://developers.facebook.com/bugs/393455200735878> není v dubnu 2013 opravena.)

Selenium je produkt, pod kterým se skrývá soubor několika programů, které slouží primárně k testování internetových stránek přímo v prohlížeči. Původně se jednalo jen o rozšíření (takzvaný plugin) do prohlížeče Mozilla Firefox. Pomocí tohoto pluginu je možné nahrát konkrétní uživatelské chování na webové stránce (například: zadat konkrétní text do uživatelského pole, kliknout na tlačítko, otevřít novou stránku a podobně). Všechny akce se nahrávají a ukládají do souboru (s příponou `.html`) a lze je jednoduše znovu opakovaně spouštět. Celý postup je totožný s používáním funkce makro pro hromadné operace v programech jako jsou Microsoft Word anebo Adobe Photoshop. Tento program se nyní nazývá Selenium IDE<sup>13</sup>.

---

<sup>13</sup><http://docs.seleniumhq.org/projects/ide/>

Díky oblibě tohoto pluginu se postupem času utvořil kolem této technologie celý ekosystém pro automatizované testování pomocí webového prohlížeče. Do tohoto ekosystému patří následující programy:

- **Selenium IDE** - plugin popsáný výše
- **Selenium Remote Control** - program pro ovládání prohlížeče lokálně nebo na jiných počítačích z různých programovacích jazyků
- **Selenium WebDriver** - program pro spouštění prohlížeče lokálně nebo na jiných počítačích
- **Selenium Grid** - za pomoci programu Selenium Remote Control dokáže spouštět webové prohlížeče na více serverech zároveň - hodí se především na testování stránek v odlišných prohlížečích a na různých operačních systémech

Více informací o jednotlivých produktech lze nalézt na oficiálních stránkách jednotlivých programů: <http://docs.seleniumhq.org/projects/>.

### 2.3.1 Popis fungování programu Selenium WebDriver

Pro naši aplikaci se nejlépe hodí program Selenium WebDriver. Jedná se o programátorské rozhraní k ovládání celého prohlížeče z různých jazyků. Můžeme tak pomocí běžného programovacího jazyka napsat program, který spustí prohlížeč, provede v něm určené úkoly a poté prohlížeč zavře. To vše bez potřeby jakékoli lidské interakce, takže tyto úkony mohou být opakovatelné a prováděné na různých počítačích.

Celý mechanismus funguje na jednoduchém principu. Selenium WebDriver je v podstatě jen jeden `.jar` soubor<sup>14</sup>. Jedná se o lokální webový server, který naslouchá na určeném portu našeho počítače a vystavuje RESTové rozhraní k ovládání prohlížeče. Z našeho programu se pak můžeme připojit k webovému serveru Selenium WebDriver a posíláme na něj požadované akce, které se mají v prohlížeči vykonat.

---

<sup>14</sup>spustitelný program napsaný v programovacím jazyce Java

## 2. ANALÝZA

---

Jelikož je webové rozhraní programu nezávislé na programovacím jazyku, existují knihovny pro mnoho jazyků a je jednoduché dopsat si i vlastní knihovnu pro jakýkoli jazyk. Program Selenium WebDriver můžeme ovládat i pomocí nástrojů pro práci s HTTP protokolem (například cURL, HTTPie, apod.). Celé rozhraní je detailně popsáno na stránkách projektu<sup>15</sup>.

Program Selenium WebDriver používá k ovládání prohlížeče takzvané drivery. V tomto případě se jedná o programy, které dokáží ovládat samotný prohlížeč - otevřít jej, otevřít konkrétní URL (viz ukázka: 2.1) a pohybovat se po stránce. Pozor, neplést s ovladači u hardware, nazývají se také drivery, ale jedná se o něco zcela jiného.

Listing 2.1: Příklad otevření URL v programu Selenium WebDriver pomocí cURL

```
# Pomoci tohoto prikazu vytvorime novou session
# (otevreme prohlizec firefox)
curl -X POST http://localhost:4444/wd/hub/session \
-d "{desiredCapabilities: {browserName: 'firefox'}}"

# Z predchoziho prikazu jsme ziskali sessionId
# sessionId pouzijeme v prikazu pro otevreni stranky
# :sessionId je potreba nahradit vysledkem predchoziho
# dotazu
curl -X POST \
http://localhost:4444/wd/hub/session/:sessionId/url \
-d "{url: 'http://www.facebook.com'}"
```

Původně dokázal Selenium WebDriver ovládat pouze prohlížeč Mozilla Firefox, ale postupem času vznikly drivery i pro ostatní prohlížeče. V této době již můžeme použít Selenium WebDriver k ovládání následujících prohlížečů: Mozilla Firefox, Google Chrome, Microsoft Internet Explorer, Apple Safari, Android browser (defaultní prohlížeč na mobilních telefonech s operačním systémem Google Android), iPhone browser (defaultní prohlížeč na mobilních telefonech Apple iPhone), PhantomJS (prohlížeč implementovaný v jazyce JavaScript, běžící jen v paměti počítače), HtmlUnit (prohlížeč implementovaný v jazyce Java, běžící jen v paměti počítače).

Mimo driverů pro webové prohlížeče existují pro program Selenium WebDriver i dva speciální drivery: EventFiringWebDriver (slouží k odchytávání

---

<sup>15</sup><https://code.google.com/p/selenium/wiki/JsonWireProtocol>

JavaScriptových událostí, které v prohlížeči vzniknou, můžeme na ně pak v našem programu reagovat) a RemoteWebDriver (slouží ke vzdálenému ovládání prohlížečů na jiných počítačích). S těmito drivery máme možnost ovládat všechny hlavní internetové prohlížeče a to jak desktopové, tak i mobilní.

Vývojáři prohlížeče Opera vydali 12. února 2013 článek [8] o tom, že plánují pro svůj prohlížeč do budoucna používat jádro WebKit, které používají prohlížeče Google Chrome a Apple Safari. Pár měsíců po tomto oznámení vydali vývojáři prohlížeče Google Chrome 3. dubna 2013 na svém webu článek [9] o tom, že vytvoří vlastní vykreslovací jádro Blink<sup>16</sup>. Bude se jednat o fork doposud používaného opensource jádra WebKit. Do budoucna bude určitě zajímavé sledovat, jak se celá situace nakonec vyvine, nicméně funkčnost programu Selenium WebDriver by to nemělo omezit.

## 2.4 Výhody a nevýhody použití programu Selenium WebDriver

Selenium WebDriver je program sloužící k ovládání internetového prohlížeče pomocí velkého množství programovacích jazyků. Největší výhodou používání programu Selenium WebDriver pro zpracování webových stránek je, že se stará o všechnu práci spojenou s komunikací skrze HTTP protokol, vykreslováním i ovládáním webových stránek a zpracováním všech dynamických skriptů na stránce.

Selenium WebDriver nám umožňuje přistupovat k výsledným vyrenderovaným elementům prohlížeče a to jak textově, tak i k HTML zdrojům jednotlivých elementů na stránce. Problém nastává při následném zpracování, protože data ze stránky nejsou vždy jednotná jako v případě použití Graph API. Musíme tedy získaná data následně zpracovávat a upravovat.

Máme tedy k dispozici naprosto stejná data, jako uživatel internetového prohlížeče. Nemusíme se zabývat nízkourovňovými záležitostmi spojenými s fungováním internetu. Dále můžeme použít různé vykreslovací programy a webové prohlížeče a to včetně programů určených pro mobilní zařízení. Tento nástroj se nejvíce používá k automatickému testování webových stránek v různých prohlížečích a v této oblasti je velmi rozšířen. Můžeme k němu tedy nalézt velké množství návodů, knihoven, doplňků a rad na internetu.

---

<sup>16</sup><http://www.chromium.org/blink>

### Výhody využití programu Selenium WebDriver

- hojně používaný program
- možnost změny vykreslovacího programu
- velké množství vykreslovacích programů k použití
- možnost ručního ovládání browseru
- velké množství návodů, zdrojů a rad na internetu
- RESTové rozhraní pro ovládání browseru
- velké množství programátorských knihoven pro práci s tímto programem
- možnost přistupovat přímo k HTML zdrojům jednotlivých elementů na stránce

### Nevýhody využití programu Selenium WebDriver

- neexistuje oficiální knihovna pro jazyk PHP od tvůrců programu Selenium
- nutnost spouštět celý browser - paměťově i časově náročnější oproti použití samostatné knihovny
- nutnost dále zpracovávat získaná data z textu nebo z HTML zdrojů elementů stránky

Pro náš experiment se program Selenium WebDriver hodí, jelikož výhody převyšují jeho nevýhody. Některé silné stránky, jako použití reálného prohlížeče, jsou vykoupeny pomalejším zpracováním. Pro experiment ale bylo důležitější použití reálného browseru, jako v případě procházení facebooku běžnými uživateli.

Neexistující oficiální knihovna pro jazyk PHP nakonec takovou nevýhodou nebylo, jelikož se mi podařilo najít jinou kvalitní knihovnu pro tento jazyk, viz dále.

Nutnost zpracovávat získaná data by byla nutná u všech postupů mimo použití Graph API. Tato nevýhoda ale byla nahrazena výhodou, a to tím že můžeme používat všechny funkce facebooku bez omezení, které Graph API přináší.

## 2.5 Využití aplikace

Facebook koncem března 2013 oznámil, že má více než miliardu aktivních měsíčních uživatelů (to znamená, že se přihlásí minimálně jednou za měsíc) [5]. Takováto masa uživatelů je obrovská a díky tomu se z facebooku stal velmi zajímavým marketingovým a komunikačním nástrojem. Díky obrovské oblibě facebooku tedy vznikly i internetové stránky zabývající se nákupem a prodejem aktivity na facebooku.

### 2.5.1 Nákup aktivity na facebooku

Na internetu se dnes běžně objevují nabídky na nákup fanoušků do facebookových skupin (příklady viz tabulka 2.2). Ve většině případů se jedná o falešné nebo ukradené facebookové účty. Některé stránky jdou ale ještě dále a nabízejí na internetu možnost koupit si na facebooku falešnou přítelkyni (<http://www.girlfriendhire.com/>, <http://namorofake.com.br/> a další).

Tabulka 2.2: Příklady webových stránek sloužících pro nákup Like do skupin na facebooku

Stránka	Cena
<a href="http://get-likes.com/facebook-likes-store/">http://get-likes.com/facebook-likes-store/</a>	1000 fanoušků za 30\$
<a href="http://fblikesmart.com/">http://fblikesmart.com/</a>	1000 fanoušků za 50\$
<a href="http://officialfacebooklikes.com/">http://officialfacebooklikes.com/</a>	1000 fanoušků za 80\$

Možnosti jsou různé od nákupu statusů od imaginární přítelkyně na vaši zeď nebo s vaším ozančením až po přidání do vztahu na facebooku. Nabízí se otázka, kolik účtů, které služby nabízí, je pravých a jaká je záruka, že po zaplacení požadované částky účty nezmizí. Každopádně se jedná o zajímavý typ výdělku pomocí facebooku.

Exparimentální aplikace by šla jednoduše využít k podobným účelům výdělku. Stačilo by na facebooku založit několik účtů, které by se pomocí této aplikace chovaly jako ostatní běžné účty. Následně by se v aplikaci nastavilo, které zaplacené statusy se mají kdy publikovat. Pravděpodobně by se muselo v aplikaci zpracovat na případech, pokud někdo uvedenou

## 2. ANALÝZA

---

osobu kontaktuje. Upozornění na všechny tyto události se ale posílají na e-mailovou adresu zadanou při registraci, takže se o nich dozvíme.

Aplikace by šla jednoduše využít i na ovládání osobních facebookových účtů. Facebook nyní neumožňuje publikovat příspěvky v určitý čas, ale publikuje se vždy ihned po odeslání formuláře. Existují aplikace pro správu facebookových stránek sloužící především pro zjednodušení komunikace se zákazníky, koordinaci publikování příspěvků na více facebookových stránkách a přidávají statistiky aktivity. Většinou jsou v takovýchto aplikacích zahrnuty i další sociální sítě jako jsou Twitter a podobně.

Všechny aplikace na správu facebookových stránek a účtů podle dostupných informací používají Graph API. Omezují se tedy pouze na dostupné akce, které toto API dovoluje používat. S využitím programu Selenium WebDriver můžeme ale využívat veškeré možnosti, které facebook nabízí.

Jedalo by se určitě o zajímavou možnost ovládání facebooku, například možnost naplánovat publikování příspěvku na určitý den a čas nebo automatické odpovědi na facebookové zprávy, stejně jako u e-mailů. Existují případy, kdy byli lidé vykradeni, kvůli tomu, že byli délejší dobu mimo domov a zloději si to odvodili z toho, že dotyční na facebook nepřidali od té doby žádný příspěvek. Další možností by bylo automatické zasílání přání k narozeninám přátelům na facebooku.



---

## Implementace

### 3.1 Požadavky na aplikaci

Před samotným vývojem aplikace jsem se zaměřil především na zjištění, které hlavní funkce bude program obsahovat a na které části se mám zaměřit. Důležité funkce aplikace pro ovládání účtů na facebooku byly následující:

- přidávání přátel
- přidávání statusů na vlastní zeď
- přidávání účtu ke skupinám
- účast na facebookových událostech
- pozvání přátel na události
- stahování dat ze zdi profilu
- stahování dat z profilů přátel

Všechny tyto funkce byly do aplikace úspěšně implementovány a sloužily k automatizaci ovládání facebookových účtů.

## 3.2 Popis uživatelského rozhraní výsledné aplikace

Aplikaci pro ovládání účtů jsem nazval **Facebook client**. Uživatelské rozhraní výsledné aplikace je vidět na obrázku výpisu uložených facebookových účtů 3.1.

ID	First name	Last name	Email	Password	Username	Facebook id	Action
2							<a href="#">Login</a> <a href="#">Download</a> <a href="#">Down Fr</a> <a href="#">Add Fr</a> <a href="#">Tasks</a> <a href="#">Edit</a> <a href="#">Delete</a>
3							<a href="#">Login</a> <a href="#">Download</a> <a href="#">Down Fr</a> <a href="#">Add Fr</a> <a href="#">Tasks</a> <a href="#">Edit</a> <a href="#">Delete</a>

Obrázek 3.1: Uživatelské rozhraní výsledné aplikace

V levé části aplikace jsou sekce: Users (uživatelé), Users groups (skupiny uživatel), Tasks (úkoly), Instagram a Quotes (citáty). Každá z těchto sekcí má v aplikaci speciální část. Postupně stručně všechny tyto části popíšu.

Sekce users slouží pro uložení facebookových účtů které bude možné skrze aplikaci ovládat. U každého účtu máme uloženo jméno a příjmení, což slouží pouze pro jednodušší orientaci mezi účty, například při přidávání nových úkolů, ale stačilo by používat pouze e-mailovou adresu. Další uloženou informací jsou přihlašovací údaje, tedy e-mail a heslo k facebookovému účtu. Poslední dva údaje jsou facebookové uživatelské jméno, zobrazené v URL adrese profilu, a facebookové číselné id profilu. Tyto údaje slouží pouze k identifikaci profilu při ukládání stahovaných informací do databáze. Tato akce také zobrazí kolik je u účtu nových přátel, zpráv a upozornění (notifikací) od posledního přihlášení.

Pro každého uloženého uživatele v aplikaci můžeme provést následující

akce. Akce Login slouží pouze k přihlášení daného uživatele, tato akce je důležitá, protože se uloží aktuální stav webového brosweru a další operace tak mohou pokračovat ve stejném okně. Download složí ke stažení dat z uživatelské zdi, udělení like statusům na této zdi a stažení všech přátel daného účtu.

Akce Download Fr (friends) se používá ke stažení informací na profilu, dat na zdi a přátel všech přátel daného účtu. Tato akce trvala delší dobu, takže nebyla spouštěna při každém přihlášení. Další možností, kterou lze pro každý účet pomocí aplikace dělat, je přidávání nových přátel.

Nejdůležitější akcí pro každý profil je Tasks neboli úkoly, na této stránce lze vybírat uložené akce a provádět je na facebookových účtech. Přidávání úkolů bude popsáno dále- Poslední dvě akce slouží k úpravě jakýchkoli informací zobrazených na výpisu uživatelů (například pro změnu helso), tato akce se jmenuje Edit a poslední akce Delete slouží k odstranění účtu.

Další část aplikace Users groups slouží k zařazování uložených profilů do skupin, například muži nebo náctiletí. Můžeme vytvořit neomezené množství takovýchto skupin a jednotlivé uložené účty lze k jednotlivým skupinám přiřazovat, jak je vidět na obázku.

#### Facebook client

- [Users](#)
- [Users groups](#)
- [Tasks](#)
- [Instagram](#)
- [Quote](#)

#### Assign users to group: male



Obrázek 3.2: Přiřazování uložených uživatelů k jednotlivým skupinám

Důležitou částí aplikace byly úkoly, neboli Tasks. Zde se ukládaly veškeré aktivity, které bylo možné na uložených účtech vykonat. Celý proces funguje tak, že uživatel nejdříve uloží všechny akce. Například úkol přidat status, zadá text, volitelně i pro který účet se má text zobrazit a vše uloží. Formulář pro přidávání nových úkolů je pro lepší pochopení celého procesu zobrazen na obrázku 3.3.

Všechny uložené úkoly, které zatím nebyly použity lze vypsat. Na výpisu

## Facebook client

- [Users](#)
- [Users groups](#)
- [Tasks](#)
- [Instagram](#)
- [Quote](#)

### New Task

Task type:

Parameter 1:

Parameter 2:

Users:

Obrázek 3.3: Formulář pro přidávání nových úkolů v aplikaci

je u každého úkolu vidět identifikátor úkolu z databáze, typ úkolu, parametry, preferovaný uživatel (pokud byl nastavený) a čas uložení. Jakýkoliv úkol lze i smazat. Uživatelské rozhraní výpisu všech připravených úkolů je na obrázku 3.4. Po vykonání úkolu (sekce Users a akce taks) a provedení všech akcí na facebookovém účtu se úkolu nastaví příznak, že je již vykonaný a ve výpisu se již dále nezobrazuje.


Předposlední sekce se nazývá Instagram. Jedná se pouze o doplňkovou funkcionalitu aplikace. Instagram (<https://instagram.com/>) je sociální síť, kam můžete nahrávat fotky z mobilního telefonu upravené pomocí různých filtrů. Jedná se tedy o obrovskou sbírku amatérských fotografií. Tuto sociální síť jsem vybral pro získávání obrázků nahrávaných jako statusy uložených účtů.

## Facebook client

- [Users](#)
- [Users groups](#)
- [Tasks](#)
- [Instagram](#)
- [Quote](#)

### Tasks

[New](#)

Id	Task	Params	User	Time	Actions
517d540d59aa88b1a000000	postStatus		26	2013-04-28 18:53:33	<a href="#">Delete</a>

Obrázek 3.4: Přehled připravených úkolů v uživatelském rozhraní aplikace

Sekce Instagram tedy sloužila k získávání náhodných obrázků. Vše funguje tak, že stačí zadat jakékoliv slovo a stránka vrátí obrázek včetně URL adresy, která lze použít při nahrávání na facebooku. Pokud se nám zobrazený obrázek nelíbí pomocí tlačítka Refresh na stránce lze zobrazit fotku jinou.

Poslední sekce se velmi podobá té předchozí. Jak již název Quote = citáty napovídá, funkcí této části aplikace je získávání citátů pro nahrávání na facebookové profily. Jako zdroje vybíraných citátů jsem použil webové stránky <http://citaty.net/>. Můžeme zde najít citáty slavných osobností, lidová pořekadla, přísloví a citáty, které nahrávají na stránky samotní uživatelé. Vybírají se vždy náhodné citáty a aktualizací stránky se zobrazí citát další, jako v případě instagramu.

## 3.3 Volba technologií pro aplikaci

### 3.3.1 Volba programovacího jazyka

Pro implementaci celé aplikace jsem si vybral programovací jazyk **PHP**<sup>17</sup>. Jedná se o dynamicky typovaný programovací jazyk s veřejným zdrojovým kódem, který se používá především k vytváření webových stránek a internetových aplikací. Tento jazyk jsem si vybral především proto, že s ním mám osobně největší zkušenosti a přijde mi na menší aplikace ideální.

Díky dynamickému typování lze v tomto jazyce vytvořit aplikace velmi rychle. Protože aplikace neobsahuje složitý model a hlavní funkcionalitou je převážně volání RESTových služeb a ukládání nebo vybírání dat z databáze,

<sup>17</sup><http://www.php.net/>

### 3. IMPLEMENTACE

---

přišel mi tento jazyk jako nejlepší možnost pro implementaci programu tohoto typu.

Celá aplikace facebooku je napsána v jazyce PHP (i když používají speciální kompilátor pro tento jazyk - HipHop for PHP<sup>18</sup>). Kromě jiných programů facebook uvolnil k volnému používání na internetu knihovnu pro používání programu Selenium WebDriver<sup>19</sup> pro jazyk PHP. Tuto knihovnu v aplikaci také používám pro komunikaci se Selenium WebDriverem.

Při rozhodování, který programovací jazyk pro aplikaci zvolím, jsem uvažoval i o jazyku Java, který je na rozdíl od PHP staticky typovaný. Jeho hlavní výhodou oproti jazyku PHP pro tento projekt byla knihovna přímo od tvůrců programu Selenium. Naopak nevýhodou byla nutnost výsledný program před spuštěním vždy překompilovat a tedy i pomalejší vývoj celé aplikace. Výhody jazyka Java jsou vidět až u větších a rozsáhlejších aplikací s rozsáhlým datovým modelem a pro takto malou aplikaci by se tyto výhody vykoupené pomalejším vývojem pravděpodobně nevyplatily.

Mezi další nevýhody jazyka Java patří to, že pokud chceme nasadit aplikaci na server, je složitější sehnat hosting a je náročnější nastavit celou aplikaci. Programy v jazyce PHP je možné nasadit i na sdílený hosting s dalšími běžícími aplikacemi a konfigurace je jen minimální. Selenium WebDriver může běžet i na jiném serveru, protože veškerá komunikace porohíhá pomocí HTTP protokolu.

#### 3.3.2 Volba databáze

Jako databázi pro ukládání dat jsem pro začátek zvolil **MySQL**. Jedná se o open source relační databázi, kterou aktuálně vlastní společnost Oracle (viz <http://www.mysql.com/>). Tato databáze je nejčastěji používána ve spojení právě s programovacím jazykem PHP. Do této databáze jsou ukládány informace o založených účtech, se kterými pracuje stahovací program. Jedná se o přihlašovací údaje (e-mail a heslo) a další údaje určené k identifikaci účtu (jméno na facebooku, uživatelské jméno na facebooku a id na facebooku).

Původně jsem chtěl pro ukládání stažených dat použít také MySQL databázi, ale rozhodl jsem se jinak. Pro ukládání dat stažených z uživa-

---

<sup>18</sup>[http://en.wikipedia.org/wiki/HipHop\\_for\\_PHP](http://en.wikipedia.org/wiki/HipHop_for_PHP)

<sup>19</sup><https://github.com/facebook/php-webdriver>

telských profilů se používá NoSQL (nerelační) databáze **MongoDB**. MongoDB je dnes jedna z nejrozšířenějších nerelačních databází. Jedná se o open source dokumentově orientovanou NoSQL databázi o kterou se stará společnost 10gen (<http://www.mongodb.org/>). Jedna z největších předností této databáze je to, že do ní lze ukládat jakákoli nestrukturovaná data ve formátu JSON (JavaScript Object Notation - <http://www.json.org/>) a pak lze s těmito daty nadále manipulovat a provádět nad nimi agregační dotazy.

Databáze MongoDB byla vybrána hlavně proto, že umí pracovat s jakýmikoli nestrukturovanými daty, to znamená, že cokoliv, co získáme z facebooku můžeme přímo bez úpravy uložit. Takovéto chování se hodí především k ukládání příspěvků ze zdi facebookových uživatelů a informací z profilových stránek osob. Nemusíme se starat o to, že někteří uživatelé mají omezenou viditelnost svých přátel nebo skupin. Kvůli těmto důvodům jsem si tedy pro ukládání stahovaných dat vybral databázi MongoDB vhodnou pro tuto úlohu.

Facebook v roce 2010 vyvinul vlastní NoSQL databázi nazvanou Cassandra ([http://en.wikipedia.org/wiki/Apache\\_Cassandra](http://en.wikipedia.org/wiki/Apache_Cassandra)), která slouží k vyhledávání nad velkým objemem dat. Facebook tuto databázi později nahradil další nerelační databází s otevřeným zdrojovým kódem nazvanou HBase (<http://hbase.apache.org/>). Tyto databáze jsou ale určeny především k vyhledávání a jejich výhoda se ukáže až při ukládání mnohem většího množství dat, než bylo pro tento experiment potřeba.

#### 3.3.3 Volba webového frameworku

Jako programovací jazyk bylo pro celou aplikaci zvoleno PHP, jak bylo zmíněno již výše. Pro tento programovací jazyk existuje kvalitní webový framework. Jmenuje se **Nette Framework** (<http://nette.org/>) a hodí se především na menší aplikace. Jedná se o MVP (Model - View - Presenter) framework z dílny českého autora. Tento framework se stará především o zpracování webových formulářů, obsluhu požadavků webového prohlížeče, vykreslování HTML šablon a celkovou architekturu celé aplikace. Více informací o tomto frameworku je k dispozici v oficiální dokumentaci (<http://doc.nette.org/>).

Pro webové aplikace, které jsou rozsahem podobně velké jako je naše experimentální aplikace je tento framework ideální. Na internetu se dá najít velké množství návodů k tomuto frameworku v českém jazyce a existuje i

početná česká komunita, která se tímto frameworkem zabývá a pomáhá s vývojem.

## 3.4 Architektura celé aplikace

Díky použití frameworku nette byla celá aplikace rozdělena podle paradigmatu MVP, tedy Model - View - Presenter. V aplikaci teď jsou samostatně nahraditelné části pro práci s daty = model, šablony pro zobrazování dat = view a část, která se stará o zpracování uživatelských požadavků a propojení Modelu a View = presenter.

V aplikaci existuje pro každou sekci v navigaci samostatný presenter. Každý presenter se stará o vykreslování stránek a zpracování formulářů pro dané sekce. Dále aplikace obsahuje několik presenterů pro správný chod aplikace, například presenter pro zpracování chyb a podobně.

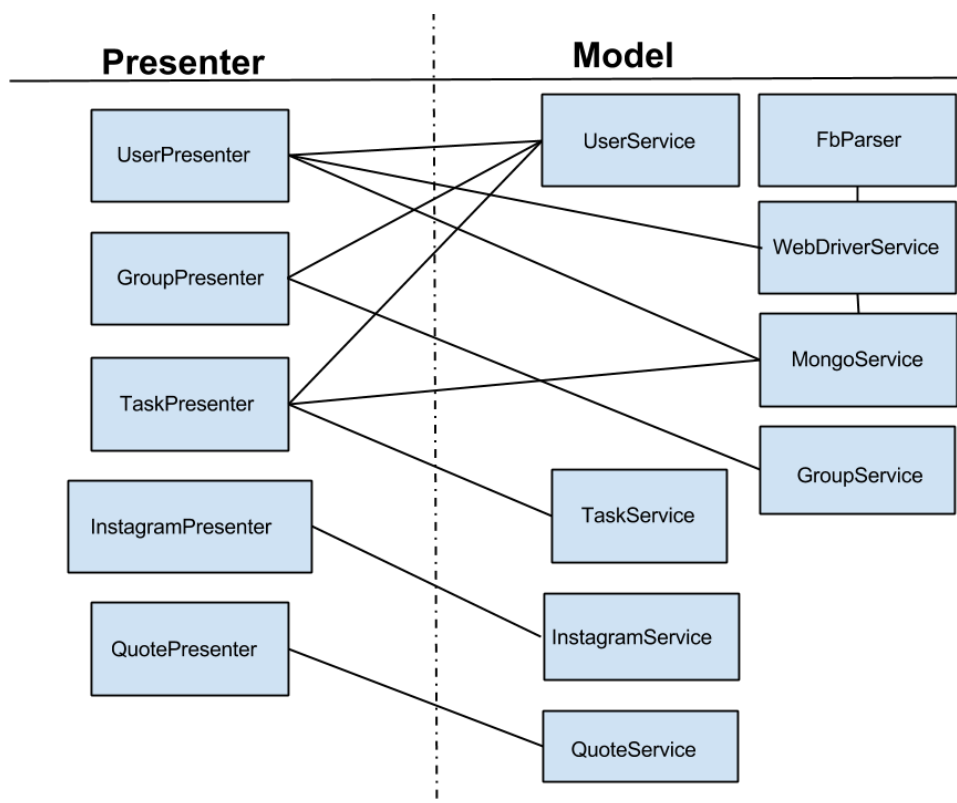
Model naší aplikace sestává ze samostatných servis (Service). Jedná se o oddělené části aplikace starající se o konkrétní části modelu. Důležité jsou servisy pro uživatele (UserService), pro skupin (GroupService) a pro úkoly (TaskService). Tyto se starají o zpracování odeslaných formulářů a ukládání dat do databáze. Poté jsou v aplikaci dvě servisy sloužící pouze k získávání dat ze ostatních stránek. Jedná se o QuoteService a InstagramService.

Nakonec existují ještě dvě speciální servisy, jde o MongoService, která se stará o ukládání dat do databáze MongoDB. Poslední a nejrozsáhlejší servisa se nazývá WebDriverService, stará se o ovládání programu WebDriver pomocí další knihovny a o stahování a zpracování dat z facebooku. Pro zpracování dat z facebooku slouží třída FbParser.

Vztah mezi třídami modelu a presenteru jsou vidět na obrázku 3.5.

View jsou vyřešeny tak, že pro každý presenter existuje samostatná složka se šablonami. Každá samostatná stránka má svůj soubor se šablonou se jménem odpovídajícím názvu stránky v URL. Pro celkové rozložení stránek existuje speciální šablona s názvem `@layout`. Tato struktura je dána frameworkem Nette a pro zápis šablon se používá nástroj latte, který je součástí tohoto frameworku.





Obrázek 3.5: Přehled vztahů mezi třídami presenteru a modelu

### 3.5 Rozdělení aplikace

Aplikace sloužící pro otestování celého experimentu má dvě základní funkce. První z nich je sbírání dat o uživatelích facebooku z jejich profilů a informací na jejich účtech. Druhou funkcí je ovládní experimentálních účtů, to znamená, nahrávání dat na facebook a interakce s dalšími uživateli. Data uživatelů z facebooku slouží k analýze a k dalšímu porovnání.

### 3.6 Stahování informací z facebookových profilů

Důležitou součástí aplikace je získávání informací o uživatelích z jejich facebookových profilů. Jak již bylo zmíněno v předchozí kapitole, vybral jsem pro stahování program Selenium WebDriver, který je spouštěn pomocí jazyka PHP. Sbíraná data jsou ukládána do databáze MongoDB nad

### 3. IMPLEMENTACE

---

kterou lze jednoduše vykonávat dotazy pro získání jakýchkoli statistických dat.

O veškeré zpracování stránek facebooku se stará program Selenium WebDriver, pomocí kterého se v prohlížeči Mozilla Firefox načte konkrétní stránka (například zeď anebo přátelé určitého uživatele). Z této stránky se poté přečtou jednotlivé fragmenty (například příspěvek na zdi anebo jméno přítele na facebooku) a tato data jsou pak pomocí PHP zpracována, upravena a uložena do databáze MongoDB. O tuto část aplikace se stará WebdriverService.

Pro každý náš facebookový účet uložený v aplikaci je spuštěna následující sekvence operací:

- stažení příspěvků na vlastní zdi
- stažení všech přátel, kteří si náš účet přidali
- stažení posledních příspěvků každého uživatele, kterého má účet v přátelích

#### 3.6.1 Zpracování stahovaných informací

Všechny statusy (příspěvky na zdi uživatele), které jsou stahovány ze zdi jsou pročištěny od nedůležitých dat a upravovány do stejné podoby. Data, která jsou získávána z facebooku, jsou někdy zobrazena rozdílně. Například zobrazení like u statusů. Pro analýzu statusů potřebujeme ukládat pouze počet like, které status obdržel. Facebook bohužel ale někdy místo počtu like zobrazuje i jména lidí, kteří like udělili. Stává se tak v případě, že like udělil, někdo z lidí, které má účet v přátelích anebo pokud je u statusu pouze jeden like. O veškeré zpracování a úpravu stahovaných informací se stará třída FbParser.

Pro účely experimentu se ukládá ke statusu pouze počet like, což je pro nás nejdůležitější údaj ohledně like. V programu se o získání počtu like stará funkce, která sečte počet jmen a počet like u statusu a vrátí výsledný součet like.

Podobná situace nastává u komentářů u facebookových statusů. Pokud status obsahuje dva až čtyři nebo méně komentářů, tak se komentáře zobrazí i s textem. (Počet plně zobrazených komentářů se různí a není dán pevně

pro každý status.) Pokud status obsahuje více komentářů, tak se zobrazí text pouze dvou (až čtyř) posledních příspěvků v komentářích u statusu a všechny předchozí komentáře jsou skryté a zobrazí se až po kliknutí na odkaz pro zobrazení starších komentářů. Naštěstí facebook v textu tohoto odkazu zobrazuje počet předchozích komentářů. Takže pro získání počtu komentářů nemusíme odkaz rozklikávat.

Text dlouhých komentářů se navíc na facebooku zkracuje, pokud je příliš dlouhý. Většinou se zobrazí pouze prvních pět řádků komentáře. Pro zobrazení celého textu je potřeba rozkliknout odkaz na konci zkráceného textu a zobrazí se pokračování textu komentáře.

Pro komentáře u skupin na facebooku se používá jiný typ zobrazení než u statusů a osobních profilů. U statusů lidí na facebooku se zobrazují komentáře chronologicky podle data vložení, jen v jedné úrovni bez zanoření a pokud chce někdo reagovat na předchozí komentář, musí označit autora předchozího komentáře pomocí znaku @ a jména jeho profilu na facebooku.

U facebookových skupin se komentáře zobrazují ve dvou úrovních s jedním zanořením, to znamená, že pokud chce někdo reagovat na určitý komentář, tak se nový příspěvek odsadí, ale pouze u první reakce a je vidět jistá hierarchie. Nadále se komentáře u příspěvků skupin neřadí podle data přidání, ale podle oblíbenosti a relevance (tento algoritmus není veřejný, ale většinou rozhoduje počet like u komentářů). Stejný způsob komentářů facebook zobrazuje i u svého doplňku Facebook Comments<sup>20</sup>, který nabízí pro ostatní webové stránky na vložení jako plugin.

#### 3.6.2 Stahovaná data z facebooku

Pro účely experimentu byla u statusů ukládána následující data získaná z facebooku za použití programu Selenium WebDriver:

- myWall - zda byl status stažen ze zdi našeho experimentálního účtu nebo ze zdi někoho z přátel účtu
- user - jméno uživatele, který status přidal
- fromUser - ze kterého účtu byl status stažen
- status - text statusu

---

<sup>20</sup><https://developers.facebook.com/docs/reference/plugins/comments/>

### 3. IMPLEMENTACE

---

- commentsCount - počet komentářů
- likesCount - počet like
- timestamp - čas vložení statusu

Dále byli ukládáni do databáze všichni přátelé profilu. Struktura těchto dat byla následující:

- name - uživatelské jméno přítele na profilu
- url - URL adresa profilu
- friendsCount - počet všech přátel stahovaného účtu, pokud to bylo možné z profilu zjistit
- mutual - počet společných přátel stahovaného účtu, pokud to bylo možné z profilu zjistit
- fromUser - kterého uživatele je stahovaný účet přítelem
- downloaded - čas stažení informace

Poslední stahovanou kolekcí do databáze MongoDB byly profily jednotlivých přátel našich profilů. Jedná se o informace o osobě, které o sobě každý na facebooku může vyplnit. Ukládaná data do této databáze jsou následující:

- user - uživatelské jméno profilu
- friend - zda se jedná o přímého přítele nebo ne
- friendsCount - počet všech přátel stahovaného účtu, pokud to bylo možné z profilu zjistit
- photosCount - počet všech fotek stahovaného účtu, pokud to bylo možné z profilu zjistit
- likesCount - počet všech like skupin stahovaného účtu, pokud to bylo možné z profilu zjistit
- eduwork - pole informací o vystudované škole a zaměstnání

- family - pole jmen rodinných příslušníků a rodinných vztahů k dané osobě
- relationship - informace o partnerském vztahu účtu
- yearly - důležité mezníky v životě uživatele seřazené podle data (narození apod.)
- Bio - popis na profilu
- Birthday - datum narození
- Sex - pohlaví
- Gender - pohlaví
- Interested In - o které pohlaví se daný účet zajímá
- Hometown - rodné město
- Current City - město, ve kterém daný účet právě žije
- Languages - jazyky, kterými účet umí mluvit
- Relationship Status - zda je daný účet v partnerském vztahu
- Religious Views - náboženské vyznání
- Political Views - politické vyznání
- Anniversary - výročí
- Mobile Phones - telefonní čísla
- Address - adresa bydliště
- Neighborhood - čtvrť
- Email - e-mailová adresa
- Facebook - adresa facebookového profilu
- Quote - oblíbené citace
- Screen Name - kontakty na různé komunikační kanály (Skype, ICQ, Jabber a další)
- Website - odkaz na webové stránky

### 3. IMPLEMENTACE

---

- downloaded - čas stažení profilu

Všechny údaje, které začínají velkým písmenkem jsou parsovány přímo z facebookové stránky. Jména klíče se tedy shodují s názvy oddílů na facebookové stránce. Začínají tedy proto velkými písmeny. Pro databázi MongoDB není v rozdílné velikosti písmen u klíčů žádný problém.

Některé z těchto informací nemusí být na profilech dostupné. Záleží vždy na konkrétním uživateli facebooku, zda je ochoten o sobě tyto informace prozradit a zveřejnit.

Jediná povinná informace, která se zadává již při registraci na facebooku je pohlaví, takže pohlaví se na stránce s osobními informacemi o uživateli zobrazí vždy. Počet zobrazených informací na této stránce se navíc může lišit pro uživatele, které mají profil v přátelích a pro uživatele, kteří jsou pro profil cizí. Každá sekce má své vlastní nastavení viditelnosti. Možnosti viditelnosti jsou: veřejné (viditelné pro všechny), přátelé (viditelné jen pro přátele), jen já (není viditelné pro nikoho kromě uživatele samotného) a vlastní nastavení (lze nastavit kteří konkrétní uživatelé nebo skupiny uživatelů mohou údaj vidět a pro které uživatele nebo skupiny je údaj naopak neviditelný).

Nová verze facebookového vzhledu zavedla na stránce s osobními informacemi nové sekce jako jsou: Hudba, Filmy, Seriály, Knihy, Místa, Like, Skupiny, Fitness a Poznámky. Protože byla aplikace nastavena pro starší verzi facebookové timeline, nepřidával jsem do stahování tyto informace, jelikož některé účty měly ještě starou verzi timeline a získávané informace by tudíž byly nekonzistentní.

## 3.7 Problémy při implementaci

Při práci na aplikaci jsem postupně narazil na různé problémy a překážky. Facebook je velmi rozsáhlá aplikace a využívá obrovské množství serverů po celém světě, aby mohl obsloužit všechny požadavky od uživatelů. Při správě takto rozsáhlé aplikace je potřeba řešit problémy se synchronizací serverů a postupným zaváděním změn do uživatelského rozhraní.

Facebook je celosvětovou sociální sítí a je tedy přeložen do velkého množství jazyků (včetně pirátské angličtiny a angličtiny s písmenky obrácenými

vzhůru nohy). Facebook je primárně vyvíjen v anglickém jazyce a do ostatních jazyků je celá aplikace přeložena. Překlad do všech jazyků probíhá pomocí speciální facebookové aplikace s názvem Translations <sup>21</sup>, kam mohou své překlady zasílat všichni uživatelé facebooku. (Dříve bylo například poměně dlouhou dobu v českém překladu špatně přeloženo slovo neděle jako slunce - anglicky Sun = slunce a zkratka pro neděli.) Jazyk procházených stránek pomocí programu Selenium WebDriver byl tedy vždy nastaven na anglický, aby nedocházelo k problémům s měnícími se překlady.

### 3.7.1 Problémy se stahovaným časem

Další problém při stahování dat z facebooku nastal při získávání času statusů a komentářů. Pro uživatele zobrazuje facebook na svých stránkách v různých formátech. První možností je relativní časový úsek od vložení, například: about a minute ago (přibližně před minutou), X minutes ago (před X minutami), X hours ago (před X hodinami) a podobně. Takovéto údaje se zobrazují u komentářů vložených před méně než 24 hodinami.

Dalším zobrazovaným časovým údajem bývá: Yesterday (včera), Monday (pondělí) a podobně. Jedná se o statusy a komentáře vložené před více než 24 hodinami a méně než týdnem. Poslední možností zobrazení časového údaje je přesné datum, které se zobrazuje ve formátu: 15 April (15 duben). Navíc se liší zobrazený čas u statusů a komentářů pod nimi. U starších komentářů se zobrazuje i čas vložení, například: 2 February at 16:45 (2. únor v 16.45), kdežto u statusů se zobrazuje čas jen v ojedinělých případech.

Nevýhodou tohoto zobrazení časových údajů je, že nejsou vůbec přesné a časový rozptyl je až 24 hodin. Zkoumal jsem proto zdrojový kód stránky, zda jsou ve stránce dostupné i další údaje týkající se času vložení statusů a komentářů. Zjistil jsem, že element obsahující čas vložení má atribut `title` s textovou časovou reprezentací a atribut `data-utime` s takzvaným unixovým časem (počet sekund od 1. ledna 1970), příklad elementu je vidět ve výpisu 3.1.

Listing 3.1: Příklad elementu obsahující čas

```
<abbr
  title="Tuesday, 23 April 2013 at 22:47"
  data-utime="1366750038"
  class="timestamp livetimestamp">
```

<sup>21</sup><https://www.facebook.com/?app=1&sk=translations>

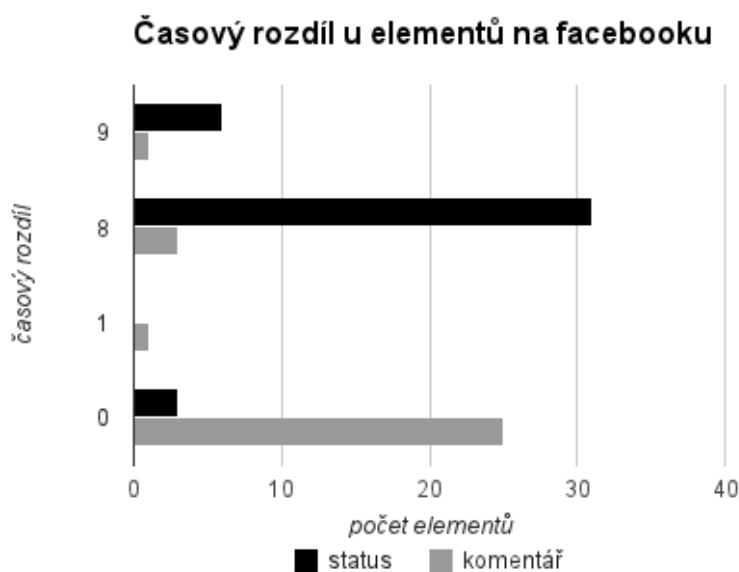
### 3. IMPLEMENTACE

---

10 minutes ago  
</abbr>

Stejné elementy používá facebook pro zobrazení času u statusů a komentářů. Zkoumal jsem vztah mezi těmito dvěma časovými údaji uvedenými u elementu. Zjistil jsem, že tyto dva časové údaje nejsou stejné. Stáhnul jsem tedy z facebooku 70 elementů s časovými údaji, z toho bylo 30 komentářů a 40 statusů. Všechna data byla získána ze tří rozdílných uživatelských zdí na facebooku.

Zjistil jsem, že u 28 elementů byly údaje v rámci tagu stejné, u jednoho elementu se čas lišil o 1 hodinu, u 34 elementů byly časové údaje rozdílné o 8 hodin a u 7 elementů se čas lišil o 9 hodin. Přehledně jsou všechna data shrnuta v grafu 3.6.



Obrázek 3.6: Časový rozdíl u atributů title a data-utime u elementů zobrazujících čas na facebooku

U některých elementů se dokonce lišil čas o rozdílný počet hodin u statusu a odpovídajících komentářů pod ním. Tento problém bude pravděpodobně způsoben různými časovými pásmy. Facebook má několik datových center po celém světě. Časový rozdíl 8 hodin odpovídá časovému posunu mezi západem USA a Greenwichským časem (základní časové pásmo).



Rozdíl mezi Českou republikou a Greenwichským časem je pak 1 hodina. Různou kombinací těchto časových posunů pravděpodobně dojde k takovýmto rozdílům, které způsobují nekonzistence mezi časy uvedenými u stejného HTML elementu.

Největším překvapením pro mne bylo to, že se různé rozdíly v časových údajích vyskytovaly na stejných zdech a dokonce se lišily i časové údaje u statusu a pod ním přidaných komentářů. Je možné, že facebook vybíral statusy z jiných datových center než komentáře a tím vznikly tyto nekonzistence.

Nakonec jsem po tomto zjištění otestoval, který ze dvou stahovaných časových údajů odpovídá realitě. Test byl naštěstí poměrně jednoduchý. Vše jsem vyzkoušel vložním statusu a komentáře na facebook a poté jsem kontroloval, jak se obě data liší oproti realitě.

Z tohoto testu jsem zjistil, že realitě odpovídá údaj vložený v atributu `title`, tedy textová reprezentace data vložení. Pro data ukládání do naší databáze jsem tedy používal tuto variantu pro parsování času.

Hlavní nevýhodou použití atributu `title` pro parsování data je, že datum je uloženo ve formátu: Tuesday, 23 April 2013 at 22:47. Nemáme tedy k dispozici údaj, ve které sekundě byl příspěvek vložen, ale pro náš experiment není tento údaj tolik důležitý. Pokud bychom potřebovali doplnit ve které sekundě byl status nebo komentář vložen, mohli bychom tento údaj získat z druhého atributu elementu (`data-utime`).

#### 3.7.2 Změna webového rozhraní facebooku

Největším problémem při použití programu Selenium WebDriver a vlastně jakéhokoli programu pracujícím s webovým rozhraním facebooku je fakt, že toto rozhraní se občas mění. Někdy se jedná o změny drobné, které běžní uživatelé této sítě ani nezpozorují. Jednou za čas na facebooku ale dojde k razantnější změně webového rozhraní.

Během března 2013 facebook oznámil novou verzi profilové stránky (timeline) [6]. Jednalo se o jednu z větších úprav webového rozhraní facebooku. Tato nová funkčnost byla navíc spouštěna postupně, takže někteří facebookoví uživatelé mají starou verzi timeline a některé účty již používají verzi novou. Vše funguje tak, že pokud je na vašem profilu zapnutá nová verze timeline, tak vidíte novou verzi na profilových stránkách všech vašich

přátel a svojí profilové stránce. Ke staré verzi timeline se již po zapnutí facebookem nejde nikdy vrátit. Nelze ani ovlivnit, kdy bude nová verze zapnuta právě pro váš profil.

Pro čtení dat z facebooku pomocí programu Selenium WebDriver se používá XPath (XML Path Language). Což je jazyk pro vybírání elementů z XML a HTML dokumentů. Vše funguje tak, že se vybírají konkrétní elementy anebo se vybere konkrétní element a procházejí se jeho podelementy. Konkrétní elementy se vybírají podle atributů, především `id` a `class`, u jednotlivých elementů.

Obával jsem se, že při změně vizuálního vzhledu timeline na facebooku budu muset předělávat velkou část stahování tak, aby aplikace mohla pracovat s novým HTML zdrojovým kódem. Naštěstí se ale HTML elementy na stránce měnily pouze minimálně. Takže stačilo upravit některé XPath dotazy a zbytek aplikace zůstal zcela funkční.

Změna timeline, která se částečně týkala i dalších stránek, ale měla jiný dopad. Na starém profilu se zobrazoval počet přátel, které má uživatel přidané na facebooku (pokud to neměl uživatel zakázané v nastavení). Se změnou na novou vizuální verzi timeline, ale tato položka zmizela a údaj o celkovém počtu přátel je dostupný až po přijetí žádosti o přátelství od dotyčného člověka. V této chvíli tedy neexistuje žádný způsob, jak zjistit celkový počet přátel na facebooku u cizích profilů.

Facebook začal od dubna 2013 povolovat maximálně pouze 100 pozvaných uživatelů na událost najednou a každá událost může mít maximálně 300 nevyřízených pozvánek. Facebook tak nutí organizátory akcí vybírat jen hosty, které bude daná akce zajímat a budou se chtít účastnit [7] [4].

## 3.8 Nahrávání dat na facebookové profily

### 3.8.1 Možné způsoby interakce na facebooku

Druhá část aplikace pro experiment byla určena pro nahrávání dat na facebookové profily. Hlavním cílem pro přidávání dat na profily byla simulace chování reálných uživatelů a zvýšení věrohodnosti účtů. Na facebooku je velké množství možností k interakci s ostatními uživateli. Nejdůležitějším prvkem na facebooku je zeď se statusy přátel. Statusy je možné komentovat, přidávat k nim like a sdílet (share = poslat na svou zeď pro zviditelnění

svým přátelům).

Kromě statusů lze na facebooku komunikovat pomocí zpráv. Systém je podobný jako u e-mailové pošty s tím rozdílem, že pomocí zpráv lze komunikovat pouze v rámci facebooku. Zprávy jsou viditelné pouze pro uživatele přidané do komunikace, stejně jako je tomu u klasických e-mailových zpráv.

Dalším komunikačním kanálem je chat, který probíhá v reálném čase. Facebookový chat je propojen se zprávami, takže veškeré informace, které jsou posílány přes chat, jsou viditelné i ve zprávách. Výhoda chatu oproti zprávám je, že můžeme vidět, kteří uživatelé jsou právě online a zda si zaslanou zprávu již přečetli.

Facebookový chat funguje pomocí otevřeného protokolu XMPP (Extensible Messaging and Presence Protocol). Tento protokol se používá pro komunikaci pomocí takzvaného Jabberu, takže pro používání facebookového chatu lze využít jakýkoli program podporující Jabber.

Další možností, jak na sebe na facebooku upozornit určitou osobu je možnost někoho štouchnout (anglicky poke). Funkčnost tlačítka štouchnout je jednoduchá. Pokud vás někdo štouchne, přijde vám upozornění a můžete štouchnutí opětovat.

Při implementaci jsem se soustředil především na automatické nahrávání statusů na uživatelské profily. V implementované aplikaci je sekce Tasks (úkoly), kde je možné nastavit jaké akce se mají pro které profily provést. V aplikaci jsou implementovány následující automatické akce publikování na facebook. Každá akce má jméno a parametry. Identifikátor je jednoznačné id na facebooku, které má každý prvek. Jedná se o několikamístné číslo a tento identifikátor lze získat z URL adresy na facebooku.

- `postStatus(text, [attachement])` - přidá na zeď uživatele status (parametr `text`), pokud je určena i příloha (`attachement`), což může být obrázek, video nebo odkaz, tak se vloží na facebook jako speciální objekt i s popisem
- `likeGroup(id)` - přidat se ke skupině (Group nebo Page), parametrem je identifikátor skupiny
- `addFriend(id)` - požádá konkrétního uživatele o přátelství na facebooku, parametrem je identifikátor uživatele

### 3. IMPLEMENTACE

---

- `rsvpEvent(id, [rsvpType])` - přidat se ke konkrétní události na facebooku, parametrem je identifikátor události, druhým parametrem je účast, možné hodnoty jsou: `attend` (účastnit se), `maybe` (možná), pokud není druhý parametr uveden, jako výchozí se bere možnost `attend`
- `inviteEvent(id)` - pozvat všechny facebookové přátele na událost, parametrem je identifikátor události

O ukládání těchto úkolů se staral v aplikaci `TaskService` a vše bylo ukládáno do databáze `MongoDB`. Ostatní typy možností komunikace na facebooku nebyly automaticky zpracovávány. Nebylo by složité doimplementovat některé další funkce jako štouchnutí všech přátel, ale pro tento experiment jsem se chtěl zaměřit na tyto hlavní funkce.

Filosofie nahrávání na jednotlivé profily je taková, že v sekci `Tasks` se uloží všechny akce s volitelným přiřazeným uživatelem. Poté se při přihlášení k jednotlivým facebookovým profilům spustí pro každý profil provedení uložených úkolů na facebooku. Z nabídky se vyberou všechny požadované akce s tím, že jsou upřednostněny akce s nastaveným publikováním pro aktuální účet (akce se zobrazují nejdříve a jsou odděleny od dalších úkolů). Uživatelské rozhraní nahrávání úkolů na facebookový profil je na obrázku 3.7.

## Facebook client

- [Users](#)
- [Users groups](#)
- [Tasks](#)
- [Instagram](#)
- [Quote](#)

### Tasks



Tasks:

Add

Obrázek 3.7: Uživatelské rozhraní nahrávání úkolů na facebookové profily

Celý datový model aplikace jsem shrnul na následujících přehledech. První přehled pro databázi MongoDB je na obrázku 3.8. Druhý přehled se týká dat v databázi MySQL, ten je na obrázku 3.9.

StatUSES	Friends	Profiles	Queue
myWall user fromUser status commentsCount likesCount timestamp	name url friendsCount mutual fromUser downloaded	user friend friendsCount photosCount likesCount eduwork family relationship yearly Bio Birthday Sex Gender Interested In Hometown Current City Languages Relationship Status Religious Views Political Views Anniversary Mobile Phones Address Neighborhood Email Facebook Quote Screen Name Website downloaded	action param1 param2 user time done

Obrázek 3.8: Přehled ukládaných dat do jednotlivých kolekcí v databázi MongoDB

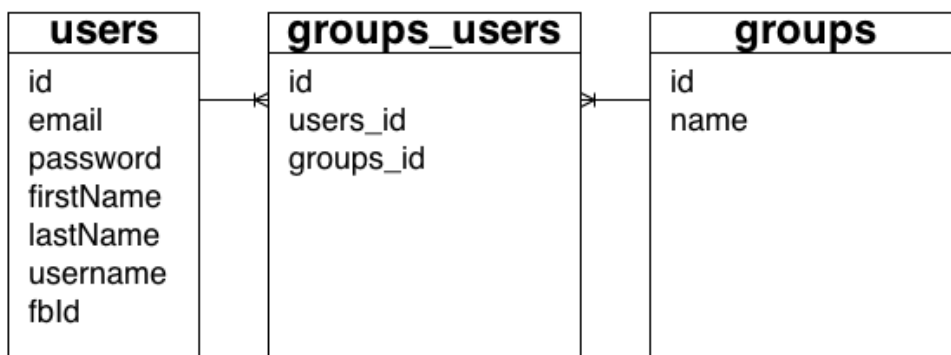
## 3.9 Části aplikace sloužící pro získání dat pro statusy

Poslední části celé aplikace byly dva pomocné části pro získávání dat z jiných stránek. Jedná se o získání fotografií ze sociální sítě Instagram a získávání citátů z webové stránky citáty.net.

Pro získávání fotek ze stránek instagramu jsem použil oficiální API na adrese <http://instagram.com/developer/>. Vše funguje podobně jako

### 3. IMPLEMENTACE

---



Obrázek 3.9: Přehled ukládaných dat do jednotlivých tabulek v databázi MySQL

Graph API pro facebook. Pro komunikaci s tímto API jsem na internetu vybral knihovnu pro PHP <https://github.com/galen/PHP-Instagram-API>. Přes toto API pak z Instagramu vybíráme fotky s určitým tagem, o vše se v aplikaci stará InstagramService.

Pro načítání citátů ze stránky citaty.net jsem použil knihovnu DOM (Document Object Model), která je součástí jazyka PHP. Vše funguje tak, že se pomocí PHP stáhne stránka s náhodným citátem, která má URL adresu <http://citaty.net/nahodny-citat/>. Ze staženého textu se pak vybere pouze část s citátem, který se vypíše. O tuto funkčnost se v aplikaci stará třída QuoteService.

## 3.10 Běh aplikace

Celá aplikace byla spouštěna lokálně na mém počítači s operačním systémem Ubuntu Linux 12.04 LTS. Teoreticky by se dalo vše nahrát na vzdálený server a ovládat přes internet, ale tím bychom ztratili výhodu kontroly prováděných operací v prohlížeči za použití programu Selenium WebDriver. Existují i internetové služby, které nabízejí pronájem výpočetního času ovládaných prohlížečů, takže lze mít celou aplikaci lokálně a spouštět program Selenium WebDriver přes internet vzdáleně. Problém je ale stejný jako v předchozím případě, nelze tyto prohlížeče ovládat.

Pokud by se aplikace rozrostla anebo by ji používalo více uživatelů, pravděpodobně by se přenesení na vzdálený server vyplatilo. Přineslo by to ale

problémy s nutností správy serveru a přísnějšího ošetřování neočekávaných stavů v aplikaci (například přidání formulářů do stránek, nebo neočekávaná vyskakovací okna).





## Průběh experimentu

### 4.1 První generace facebookových účtů

Před samotným startem celého experimentu jsem potřeboval otestovat, jak přibližně bude vše fungovat a jak se budou účty pravděpodobně chovat. Založil jsem proto na facebooku pět testovacích účtů, které jsem nazval **první generace**. Pomocí těchto účtů jsem zjišťoval omezení i komplikace celého experimentu a jak by vše mělo fungovat. S využitím účtů z první generace jsem vyvíjel aplikaci pro automatické stahování a nahrávání dat na facebook (viz předchozí kapitola).

Pomocí první generace jsem ověřil, že facebook nekontroluje, pokud se z jedné IP adresy přihlašuje velké množství účtů. Denně jsem se přihlašoval k sedmi různým facebookovým účtům a nikdy jsem nenarazil na žádný problém. Je to i logické, jelikož v institucích jako jsou školy, knihovny a podobně se může k facebooku přihlašovat velké množství účtů, které jsou k internetu připojeny skrze jednu IP adresu<sup>22</sup>, přes kterou jsou instituce připojovány k internetu.

### 4.2 Příprava experimentu

#### 4.2.1 Zakládání e-mailových adres

Před samotným provedením experimentu bylo potřeba založit facebookové účty pomocí kterých se bude celý experiment realizovat. Zakládání expe-

---

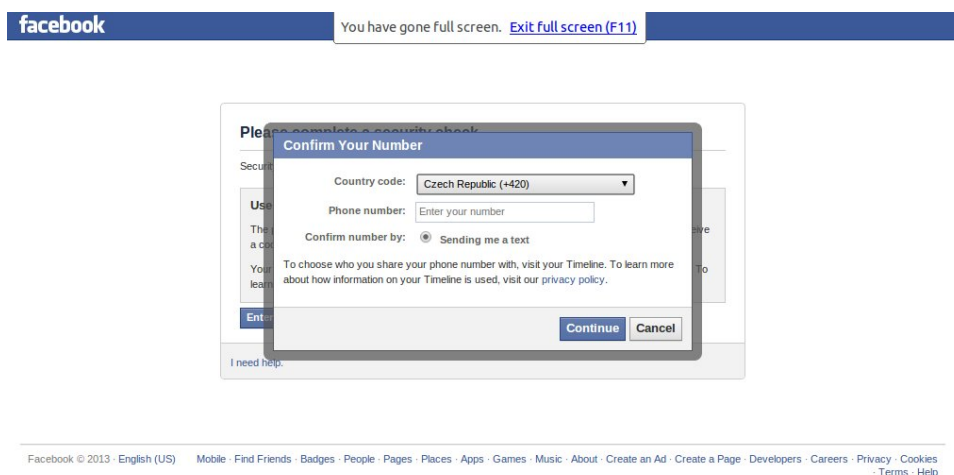
<sup>22</sup>IP adresa - číselná adresa počítače, pomocí které se připojuje k internetu

#### 4. PRŮBĚH EXPERIMENTU

---

rimentálních účtů bylo poměrně složité, jelikož jsem chtěl původně spustit celý experiment v jeden den. Facebook při vytváření nového účtu kontroluje vždy IP adresu počítače a nedovolí ze stejné IP adresy založit více než jeden účet denně.

Pokud se někdo pokusí založit více než jeden účet, zobrazí se na facebooku obrazovka s nutností opsat kontrolní CAPTCHA<sup>23</sup> obrázek a po úspěšném opsání facebook zobrazí stránku (viz obrázek 4.1) s polem pro zadání telefonního čísla, na které má být zaslán aktivační kód. Nepokoušel jsem se pak už dále dostat přes tento krok, jelikož telefonní čísla se složité shání a jedná se o osobní údaj, který jsem nechtěl facebooku sdělovat.



Obrázek 4.1: Zadávání telefonního čísla pro ověření na facebooku

Nejdříve bylo potřeba pro všechny facebookové účty založit e-mailové adresy, sloužící k vytvoření účtů na facebooku. Všechny e-mailové adresy byly založeny na české doméně s koncovkou `.cz`, protože jsem chtěl provést experiment na uživatelích facebooku z České republiky a abych se mohl kvalitněji zaměřit na určitou skupinu uživatelů facebooku.

Pro vytváření e-mailových adres jsem použil především bezplatnou e-mailovou službu od portálu seznam.cz (jedná se v první řadě o internetový vyhledávač, e-mailové schránky jsou provozovány jako jedna z doplňkových služeb). Na tomto portálu lze vytvářet e-mailové adresy s koncovkou `@seznam.cz` a `@email.cz`.

<sup>23</sup>CAPTCHA - jedná se o ochranu internetových stránek proti robotům, nejčastěji je realizovaná nutností opisání textu z deformovaného obrázku

Pro každý experimentální účet na facebooku jsem nejdříve připravil náhodné jméno a příjmení. Hlavním požadavkem na jméno bylo, aby se jednalo o co nejvíce náhodné, ale zároveň i uvěřitelné jméno. Nechtěl jsem proto vytvářet nová jména, ale raději využít již existující. Ve velkém množství případů se jednalo o spojení různých křestních jmen a příjmení získaných ze dříve stažených facebookových profilů. U některých jmen byl změněn i rod příjmení.

Nadále jsem pro vytváření jmen použil službu [www.zlatestranky.cz](http://www.zlatestranky.cz). Jedná se o internetový seznam telefonních čísel fyzických a právnických osob v České republice. V současné době stránky obsahují pouze seznam právnických osob a změnila se grafika celých stránek, ale v době kdy jsem účty zakládal byl na stránkách i seznam fyzických osob. V dnešní době by bylo možné použít některý podobný internetový seznam fyzických osob.

Struktura e-mailových adres byla následující: `jmenoprijmeni@domena.cz`. Přednost měla doména `seznam.cz`, pokud byla e-mailová adresa s tímto jménem již zabraná, použil jsem doménu `email.cz`. Pokud nebyla k dispozici ani tato e-mailová schránka, tak se za jméno vložila číslice (například: `jannovak5@seznam.cz`) anebo byla použita modifikace křestního jména (domácký výraz, zdvojnásobení a podobně).

Při vytváření e-mailových adres jsem na žádná omezení nenarazil. Webových stránek nabízejících bezplatně e-mailové adresy je velké množství a to i pokud potřebujeme e-mailovou adresu pouze z české domény. Po úspěšné registraci na facebook bylo vždy třeba aktivovat facebookový účet pomocí e-mailu. Facebook ihned po odeslání registračního formuláře zašle na e-mail zadávaný při registraci zprávu s ověřovacím odkazem. Aktivace účtu probíhá otevřením ověřovacího odkazu z e-mailu. Facebook tak kontroluje existenci a funkčnost této e-mailové schránky.

Facebook po úspěšné registraci na zadanou e-mailovou adresu posílá upozornění na různé aktivity, které se na facebooku dějí (například přijatá žádost o přátelství, obdržení facebookové zprávy, označení na fotce a podobně). Všechny možné typy zasílaných zpráv z facebooku lze nastavit v uživatelském nastavení. Pro experiment byly e-mailové adresy důležité pouze pro registraci facebookového účtu.

### 4.2.2 Internetové služby poskytující dočasné a veřejné e-mailové schránky

Před začátkem experimentu jsem uvažoval nad použitím některé ze služeb poskytujících dočasné nebo veřejné e-mailové adresy. Jedná se o služby, které vám při návštěvě vygenerují náhodnou e-mailovou adresu (například: `e1461664@rmqkr.net`). Takovéto e-mailové adresy fungují většinou jen po omezeně dlouhou dobu (v řádu několika desítek minut až hodin) a všechny e-mailové zprávy, které jsou odeslány na tuto adresu se zobrazí přímo na navštívené stránce veřejně (u dočasných e-mailů jen pro návštěvníka, pro kterého byla daná e-mailová adresa vygenerována).

Takovéto internetové služby slouží především jako ochrana proti e-mailové nevyžádané poště (takzvaný spam) a pro registraci na pochybné internetové služby. Po uplynutí časového intervalu bývá e-mailová adresa zrušena.

Příklady dočasných a veřejných e-mailových služeb:

- `10minutemail.com` - vytvoří náhodnou e-mailovou adresu na různých doménách platnou 10 minut, tato doba lze postupně prodlužovat, vždy o dalších 10 minut
- `mailinator.com` - možnost vygenerovat si jakoukoli e-mailovou adresu na doméně `mailinator.com` nebo `spamherelots.com`, k této e-mailové schránce může přistupovat kdokoli, ale takováto e-mailová adresa má časově neomezenou platnost
- `meltmail.com` - funguje tak, že po omezenou časovou dobu přeposílá veškeré e-maily zaslané na vygenerovanou adresu na e-mail zadaný uživatelem, platnost vygenerovaných e-mailových adres může být 3, 6, 12 nebo 24 hodin

Zjistil jsem ale, že facebook registrace z takovýchto e-mailových adres blokuje a po registraci je ihned potřeba zadat telefonní číslo pro ověření pravosti osoby, na které přijde ověřovací kód, stejně jako při registraci více účtů z jedné IP adresy. Dále se mi nepodařilo najít službu s dočasnými e-mailovými adresami, které by obsahovaly doménu s koncovkou `.cz`.

Použil jsem proto služby českých e-mailových serverů. Hlavní výhodou je to, že k e-mailovým schránkám jsem měl přístup i později a nebyl by pro

mne problém pokud by bylo potřeba dohledat některé informace posílané e-mailem (například zapomenuté heslo nebo důvod zablokování facebookového účtu).

### 4.3 Problémy se zakládáním facebookových účtů

Profily na facebooku tedy byly zakládány již dopředu, aby mohl být experiment spuštěn najednou ve stejný čas. Bohužel se ale stalo to, že dříve založené účty na facebooku byly neaktivní a když jsem se k nim přihlásil po delší době, tak facebook požadoval zadání telefonního čísla, pro ověření totožnosti uživatele (stejně jako pokud se pokoušíme zaregistrovat více účtů denně z jedné IP adresy). Problém byl v tom, že facebook bere účty jako věrohodné až pokud mají přidáno několik přátel (přibližně deset) a vyvíjí aktivitu na facebooku.

Stalo se tedy to, že jsem zakládal účty, aktivoval jsem je pomocí e-mailové adresy, ale už na účtech nebyla žádná další aktivita. Když jsem se poté v den spuštění experimentu k účtům vrátil, tak se mi objevilo textové pole s prosbou o ověření účtu zadáním telefonního čísla. Musel jsem tedy spustit experiment jen s funkčními účty s tím, že jsem po spuštění experimentu zakládal další nové účty.

Všechny facebookové účty jsem zakládal ručně, protože jsem se nechtěl v experimentu zabývat automatizací tohoto procesu. Celý experiment by šel rozšířit o tuto funkčnost, ale složitost tohoto problému je poměrně velká a v následující kapitole popisuji, jak by šlo zakládání facebookových účtů teoreticky plně zautomatizovat.

### 4.4 Skupiny facebookových účtů pro experiment

Při vymýšlení celého experimentu bylo hlavním cílem simulovat reálné chování uživatelů na facebooku. Chtěl jsem také vyzkoušet různé strategie přidávání přátel. Na začátku experimentu jsem tedy rozdělil všechny účty do pěti kategorií podle vyplněnosti profilu. Každá kategorie měla známku od 1 do 5, přičemž známka 1 znamená, že profil je nevyplněný a známka 5 znamená, že profil je zcela vyplněný. Hlavní motivací pro takovéto rozdělení

je, že více vyplněné facebookové profily by měly být pro ostatní návštěvníky věrohodnější a lidé by si je měli teoreticky více přidávat.

Na následujícím seznamu jsou faktory ovlivňující **vyplněnost profilu**:

- profilová fotka
- úvodní fotka (velká fotka na profilové stránce)
- vyplněny informace na stránce o sobě - především město ve kterém osoba právě žije, datum narození a oblíbené citáty
- veřejná viditelnost informací na profilu

Profily, které měly úroveň vyplněnosti na profilu 1, neměly použitou ani jednu z možností uvedených výše. Profily s úrovní 2 měly náhodně vybranou jednu z výše zmíněných možností zatraktivnění profilu. Profily s úrovní 5 měly vyplněny všechny výše zmíněné možnosti.

K získání fotografií k profilům posloužily internetové seznamky a sociální sítě. Z českého prostředí se jednalo především o stránky: [libimseti.cz](http://libimseti.cz) a [www.lide.cz](http://www.lide.cz). Ze zahraničních webů posloužila nejlépe internetová stránka [www.myspace.com](http://www.myspace.com). Výhodou těchto stránek je, že lze vybírat profily podle pohlaví a věku, což bylo ideální, protože bylo jednoduché najít profilové fotky odpovídající pohlaví a věku jakéhokoli experimentálního facebookového profilu. Veškeré obrázky jsou navíc veřejně dostupné bez nutnosti přihlášení na zmíněné služby.

##### 4.4.1 Strategie přidávání prvních přátel na facebooku

Pro experimentální facebookové účty jsem pro experiment vytvořil pět různých strategií přidávání nových přátel. Cílem bylo otestovat, jak úspěšné strategie budou a zda bude rozdíl v počtu přidávaných přátel mezi různými strategiemi.

**Strategie přidávání nových přátel:**

- **many friends** - přidávání osob s vysokým počtem přátel na facebooku

- **fake profiles** - přidávání falešných facebookových profilů
- **facebook suggestions** - přidávání profilů na facebooku, které doporučí facebook
- **mutual friends** - přidávání profilů s největším počtem společných přátel
- **facebook groups** - přidávání skupin na facebooku a následné přidávání profilů, které mají u skupiny like

Všechny experimentální účty byly zařazeny do jedné z těchto pěti skupin. Toto rozdělení profilů bylo ještě navíc zkombinováno se skupinami vyplněnosti profilů popsaných výše. V každé skupině bylo tedy vždy pět účtů s různou skupinou vyplněnosti. Vznikl tak soubor pětadvaceti facebookových profilů, z nichž každý byl zařazen do jiné skupiny vyplněnosti a také rozdílné skupiny pro přidávání nových přátel.

U některých profilů zařazených do určitých skupin přidávání nových přátel na facebooku bylo nutné ze začátku přidat první přátele. Jednalo se o skupiny many friends a mutual friends, tyto strategie používaly již existující přátelé u profilu. U těchto skupin byly první uživatelé přidáni z některé náhodně vybrané skupiny na facebooku.

K přidávání prvních přátel jsem využil facebookové pluginy, které jsou veřejně dostupné na některých internetových stránkách. Jedná se o takzvaný facebook like box [3], kde jsou zobrazeny profilové fotky a odkazy na facebookové profily uživatelů, kteří mají danou stránku v oblíbených. Využil jsem tedy takto veřejně dostupné náhodné facebookové profily a přidával je k experimentálním účtům ve výše zmíněných skupinách.

V původním plánu bylo využití událostí a her na facebooku pro přidávání nových přátel, ale po vyzkoušení na první generaci jsem tento způsob zavrhl. Prvotní myšlenka byla taková, že se experimentální účet bude přidávat na facebookové události a bude si postupně přidávat účastníky akcí. Po vyzkoušení jsem ale zjistil, že je k tomuto účelu lepší využít facebookové skupiny. Velké množství událostí s vysokým počtem účastníků zakládají kluby nebo organizace a ty mají ve většině případů založeny své facebookové skupiny, pomocí nichž akce promují. Navíc si uživatelé těžce spojovali souvislost mezi událostí a zaslanou žádostí o přátelství.

#### 4. PRŮBĚH EXPERIMENTU

---

Využití facebookových her pro přidávání nových přátel by mohlo být velmi zajímavé, jelikož v hrách existují různé výhody za zaslání pozvánek facebookovým přátelům a někdy můžete přátele na facebooku požádat, aby vám poslali některé vylepšení nebo s nimi vyměňovat suroviny ve hrách. Čím více přátel na facebooku do hry tedy účet pozve, tím větší výhody ve hře získává. Při hraní her bývá běžné, že si uživatelé zakládají na facebooku nové účty jen aby mohli získat co nejvíce těchto výhod.

Bohužel má ale každá hra jiné ovládání a velké množství her je napsáno v technologii Adobe Flash<sup>24</sup> nebo podobných, které nejsou přístupné přes HTML zdrojový kód stránky a nešel by tedy pro interakci s nimi použít program Selenium WebDriver. Pro ovládání těchto her by musel být vyvinut mnohem složitější způsob procházení stránek. Musel jsem tedy od tohoto záměru upustit, ale jedná se o zajímavý segment, kde by se mohly účty sami doporučovat a posílat pozvánky na facebooku.

Je důležité zmínit, že žádné dva experimentální účty nebyly sobě navzájem přáteli na facebooku. Ani se nestalo, že by se některé účty přátelily na facebooku se stejnými lidmi. V pozdější fázi některé aktivní facebookové účty se staly přáteli více experimentálních účtů, ale záměr byl co nejvíce simulovat chování reálných uživatelů a aby byly počáteční přátelé účtů co nejvíce náhodní.

Celkově jsem kvůli problémům se zakládáním facebookých účtů pracoval s 25 experimentálními účty. K těmto účtům jsem se přihlašoval vždy jednou za tři až čtyři dny. Vše vycházelo tak, že jsem se za den přihlásil k osmi až devíti účtům z těchto pětadvaceti. Za jeden týden jsem se tedy ke každému experimentálnímu účtu přihlásil přibližně dvakrát s tím, že pokud se u některého účtu objevilo podezření z jakékoli nestandardní činnosti. Provedl jsem požadované akce pro daný účet a pokračoval jsem s dalšími účty v sérii až následující den.

Po každém navštívení všech pětadvaceti účtů jsem provedl jednodenní pauzu a nepřihlásil jsem se k žádnému z experimentálních účtů. Navštívení všech účtů i s pauzou tedy trvalo celkem 4 dny. Přihlašování probíhalo v pravidelných půltýdenních intervalech.

Začátek experimentu byl naplánován na pondělí 1. dubna 2013 a trval přesně 31 dnů až do 2. května 2013. Všechny účty byly za tuto dobu běhu

---

<sup>24</sup>Adobe Flash - <http://www.adobe.com/products/flash.html>



experimentu navštíveny přesně osm krát. Výsledky získané během toho experimentu byly velmi zajímavé.

##### 4.4.2 Fáze experimentu

Strategie přidávání přátel popsané v předchozí kapitole byly využívány ze začátku experimentu, když bylo potřeba nabrat určitou základnu přátel na facebooku, aby profil vypadal pro ostatní facebookové uživatele jako aktivní a důvěryhodný. Tato fáze trvala dva a půl týdne a hlavním cílem bylo získat přátele, kteří budou ochotni přijmout žádost o přátelství na facebooku od experimentálních účtů.

V pozdější fázi experimentu byla využívána pouze strategie facebook suggestions. Tato strategie totiž počítala s přidáním přáteli k experimentálnímu účtu a pracovala hlavně se společnými a naposledy přidávanými přáteli. Hlavní výhodou bylo, že se přidávali uživatelé z podobných okruhů a přátelé přátel. Ostatní strategie pouze přidávaly přátele bez jakýchkoli vazeb na stávající facebookové přátele.

Celý experiment probíhal tak, že jsem pro každý experimentální účet nejdříve spustil stahování všech dat z profilu. V rámci tohoto stahování se stáhla zeď přihlášeného účtu a při té příležitosti byl udělen like třem statusům uživatelů (statusy od facebookových stránek se přeskakovaly) s nejvyšším počtem like. Like byl přidáván pouze textovým statusům, facebook na zdi zobrazuje i aktivity, například, že někdo změnil své město na stránce s informacemi o uživateli nebo že byl uživatel označen na fotce a podobně, všechny tyto aktivity byly odfiltrovány.

Pro přidávání like ke statusům se procházelo pouze dvacet prvních statusů na stránce, aby nedocházelo k přidávání like k neaktuálním statusům. Při menším počtu přátel u experimentálního účtu na facebooku se totiž zobrazovaly na zdi statusy starší než několik měsíců. Všechny parametry, tedy počet přidávaných statusů za jedno přihlášení i počet statusů pro které se počítá počet like by šlo upravit, ale tyto hodnoty mi přišly při testování na první generaci ideální.

Občas se stávalo, že všechny tři like u statusů za jedno přihlášení získaly statusy jednoho uživatele na facebooku. Jednalo se především o velmi aktivní profily s velkým počtem přátel - tím pádem měly velký počet like u veškeré své aktivity na facebooku a zároveň se na zdi zobrazovalo velké množství jejich statusů. Vše by šlo vylepšit kontrolováním autorů statusů

a přidáváním like pouze u jednoho statusu pro každého autora. Naštěstí k přidávání like u statusů pouze jednoho uživatele nedocházelo často, takže jsem algoritmus neměnil.

Po stažení všech informací přišlo na řadu nahrávání informací na uživatelské profily. Pro každý profil bylo připraveno několik možných statusů na nahrání na facebook a stačilo jen vybrat konkrétní status a odeslat formulář. Statusy je možné připravit dopředu a přiřadit pro přidání na konkrétní profil nebo pro kterýkoli profil.

Přidávání přátel na facebooku probíhalo podobným způsobem. Nejdříve se otevřelo okno pro přidávání přátel daného profilu. Na pozadí se stáhlo prvních 10 přátel, které facebook danému profilu doporučil. Jedná se o stránku: <https://www.facebook.com/find-friends/browser/>. Experimentální aplikace zobrazila jména těchto doporučených přátel a bylo možné vybrat, kterým uživatelům bude zaslána žádost o přátelství.

Ze začátku jsem chtěl při experimentu vždy automaticky zaslat žádost o přátelství prvním pěti nabídnutým facebookovým účtům, ale při testování na první generaci jsem zjistil, že se takto k profilu přidávaly účty ze zahraničí, které nerozumněly přidávaným statusům (viz dále). Chtěl jsem, aby nově přidávané profily byly převážně z České republiky, proto bylo potřeba přidat mezikrok s vybíráním relevantních přátel.

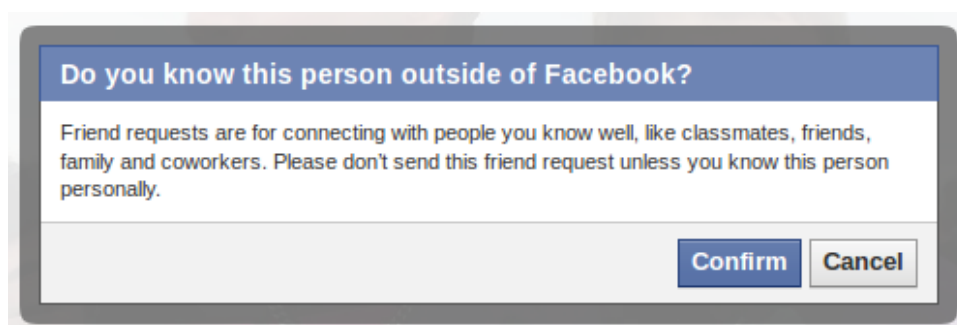
#### 4.4.3 Četnost přidávání přátel

Hlavním cílem interakce na facebooku byla věrohodnost pro ostatní facebookové uživatele. Ze začátku jsem po zaregistrování nového profilu plánoval přidávat pokaždé přibližně sedm až osm přátel. Když se zaregistruje nový uživatel na facebook, tak nemá skoro žádné možnosti využití, dokud nemá nějaké přátele, většina uživatelů se tak snaží zpočátku přidávat více přátel.

Facebook ihned po registraci nového profilu nabízí možnost stáhnutí kontaktů z e-mailu, z programu Skype nebo z programu ICQ za účelem nalezení nových přátel na facebooku. Vše funguje jednoduše tak, že stažené kontaktní údaje se porovnávají s registrovanými uživateli facebooku a pokud se najde shoda, tak facebook nabídne tyto účty ke vzájemnému spřátelení.

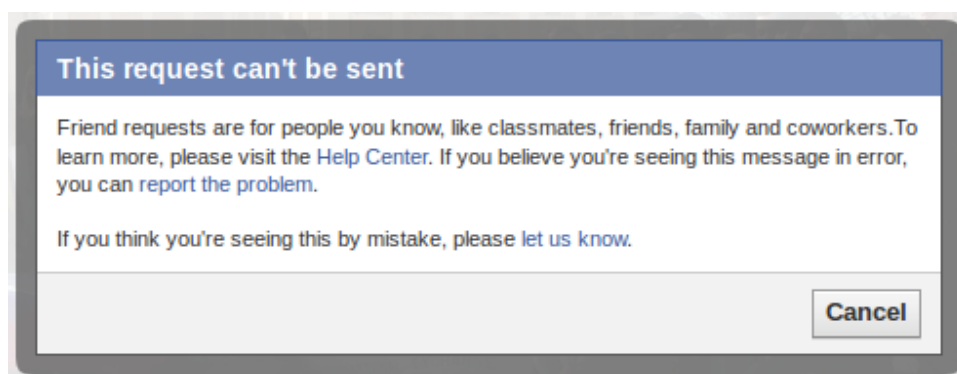
Facebook ale při přidávání většího množství přátel (většinou okolo pěti, ale někdy i méně) zobrazil potvrzovací okno, viz obrázek 4.2. Jedná se o potvrzení, zda známe daný facebookový účet i v reálu. Tento dialog stačí

potvrdit a vše funguje. Ustálil jsem tedy po tomto zjištění počet přidávaných přátel na pět za jedno přihlášení. Během prvního kola přihlašování bylo zasláno sedm až osm žádostí o nová přátelství a od druhého kola přihlašování už byl počet vždy pět.



Obrázek 4.2: Potvrzení zda se znají osoby v reálu při zasílání žádosti o přátelství na facebooku

U některých účtů, především s málo přidanými přáteli, se mi ale stávalo, že se mi zobrazilo podobné okno jako v předchozím případě. Jednalo se ale o nepovolení žádosti o přátelství facebookem. Obrázek zprávy viz 4.3. Facebook tuto zprávu zobrazil ihned po aktivování tlačítka pro odeslání žádosti o přidání do přátel, nejednalo se tedy o zrušení ze strany uživatele, kterému byla pozvánka zaslána.



Obrázek 4.3: Zrušení žádosti o přátelství na facebooku

Počet pěti nových žádostí o přátelství jsem ponechal u každého účtu až do konce experimentu. Pokud se ale stalo, že náš účet obdržel některé žádosti o přátelství od ostatních uživatelů, tak byly všechny přijaty.

Počet nově přidanych přátel za jedno přihlášení se pak snižoval o počet přijatých žádostí o přátelství. Pokud počet nových přijatých žádostí o přátelství překročil pět, tak už nebyly přidávány další nové profily.

Snažil jsem se, aby nově přidávání přátelé byli české národnosti. Hlavním důvodem bylo, aby přidané profily rozuměly publikovaným statusům a aby byly požadavky o přátelství realističtější. Při testování první generace se mi stalo, že po přidání několika účtů s arabskými jmény mi facebook začal nabízet pouze arabské přátele. Někteří žádost o přátelství přijali, ale většinou jen ze zvědavosti a daný účet začal dostávat pozvánky na facebookové skupiny s arabskými názvy. Proto jsem se při experimentu snažil přidávat pouze facebookové uživatele z České republiky.

Pro zachování autentičnosti chování uživatelů facebooku jsem se při přidávání přátel snažil, aby si účty přidávaly ve větší míře profily opačného pohlaví. Tedy aby si experimentální profily se ženským jménem přidávaly více mužských přátel na facebooku a naopak.

#### 4.4.4 Nahrávání statusů k experimentálním účtům na facebooku

Pro experiment jsem sdílení omezil na přidávání statusů na svou zeď. Jedná se totiž o hlavní možnost, jak dát na facebooku o sobě vědět. Frekvence přidávání statusů byla nastavena na jeden status za přihlášení.

Obsah vkládaných statusů byl obecný a neutrální. Jednalo se většinou o citáty a výroky známých i neznámých osobností na internetu, vtipy a neutrální a spontánní komentáře aktuálního dění (například o počasí). Jako zdroj pro statusy citátů sloužila webová stránka: [citaty.net](http://citaty.net) a jako zdroj vtipných statusů sloužily stránky: [www.vtipy.net](http://www.vtipy.net), [vysmatej.cz](http://vysmatej.cz) a [vtipy.peoplelovepeople.com](http://vtipy.peoplelovepeople.com). Snažil jsem se vyhýbat rasistickým a politickým vtipům, abych nepobouřil některého z přidanych přátel na facebooku. Spontánní texty statusů jsem vymýšlel sám nebo jsem se inspiroval statusy na zdi ostatních účtů.

Mimo klasických textových statusů byly na účty nahrávány i obrázky doplněny textovým popisem. Na facebooku jsou tyto obrázky velmi oblíbené, existují webové stránky jako [9gag.com](http://9gag.com) nebo [www.rouming.cz](http://www.rouming.cz), které obsahují vtipné obrázky s popisky a facebookoví uživatelé je velmi rádi sdílejí a dávají jim like. Pro stahování dalších obrázků pak sloužila sociální síť in-

stagram. Jako zdroje obrázků pro statusy sloužily obrázky nalezené na ostatních stahovaných profilech nebo na stránkách zmíněných výše. Obzvláště oblíbené jsou v posledních týdnech animované obrázky anebo obrázky doplněné vtipnými texty (takzvané memy).

[todo]- stahovani friend - friend dat

#### 4.4.5 Zajímavosti na které jsem během experimentu narazil

Po přihlášení k účtu, stažení informací a spuštění automatizovaných akcí jsem vždy ručně kontroloval, co se na facebookovém účtu děje. Ve většině případů se u účtu zobrazovaly notifikace o potvrzení žádostí o přátelství a několik zpráv o tom, zda náš účet zná osobu, které byla odeslána žádost o přátelství a občas i šfouchnutí.

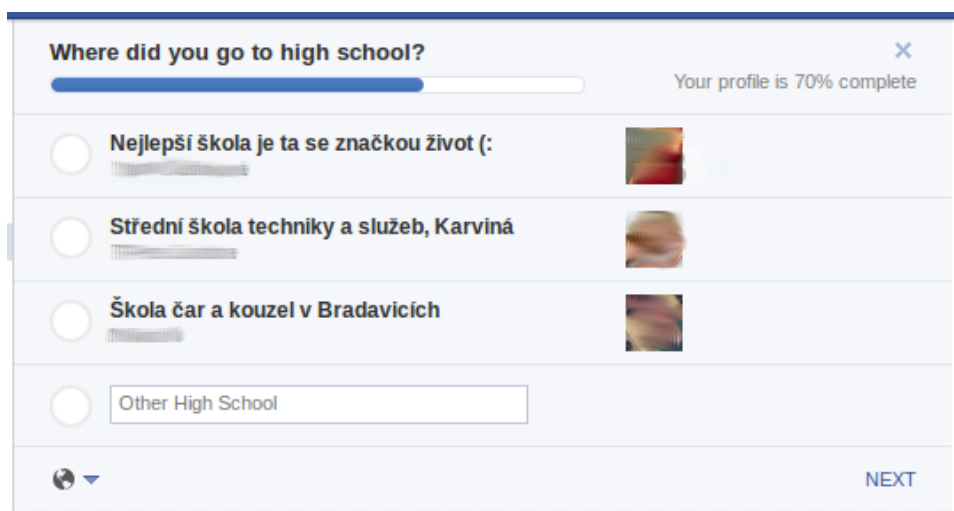
Facebookové zprávy nebyly automaticky zpracovávány. Zpracovávání by totiž nepřinášelo přidanou hodnotu k našemu experimentu a zpracovávání textu by bylo složité. Na žádné zprávy jsem nikdy neodpovídal. Teoreticky by bylo možné připravit univerzální odpověď na otázku, zda se dané facebookové účty znají. Například vymyšlení univerzálního místa, kde se mohly dané osoby setkat - škola, práce, apod., anebo přiznání, že se účty neznají. Pak by ale zcela určitě pokračovalo další dotazování pomocí facebookových zpráv, odkud se dané účty znají anebo proč byla odeslána žádost o přátelství na facebooku. Nejen, že by takovéto zpracovávání zabralo mnoho času a nebylo by možné plně automatizovat, ale někdy by mohlo takovéto chování být i kontraproduktivní.

Při procházení profilů jsem ale občas narazil na zajímavé jevy. Pokud účet neměl na facebooku vyplněné některé informace, konkrétně školu nebo bydliště. Zobrazilo se na facebooku na výpisu zdi (což je stránka, která se zobrazí ihned po přihlášení) nad výpisem statusů upozornění o doplnění těchto informací i s nabídkami. Příklad takového formuláře je vidět na obrázku 4.4.

Kromě přidaných formulářů na stránce zdi se po přihlášení u některých účtů zobrazil tříkrokový průvodce pro vyplnění bezpečnostních informací. Jedná se o zadání kontaktních informací při ztrátě facebookového hesla nebo při odcizení facebookového účtu cizí osobou (například při prozrazení hesla k účtu). Pokud by byla potřeba, facebook umožňuje resetovat heslo a zaslat

#### 4. PRŮBĚH EXPERIMENTU

---



Obrázek 4.4: Formulář na facebooku nabádající k vyplnění vystudované střední školy

nové heslo na jeden ze zadaných kontaktních údajů. Jedná se také o ověření, že bude heslo změněno jen skutečným vlastníkem účtu. Zadávat se tyto informace: e-mail, telefonní číslo a bezpečnostní otázka včetně odpovědi.

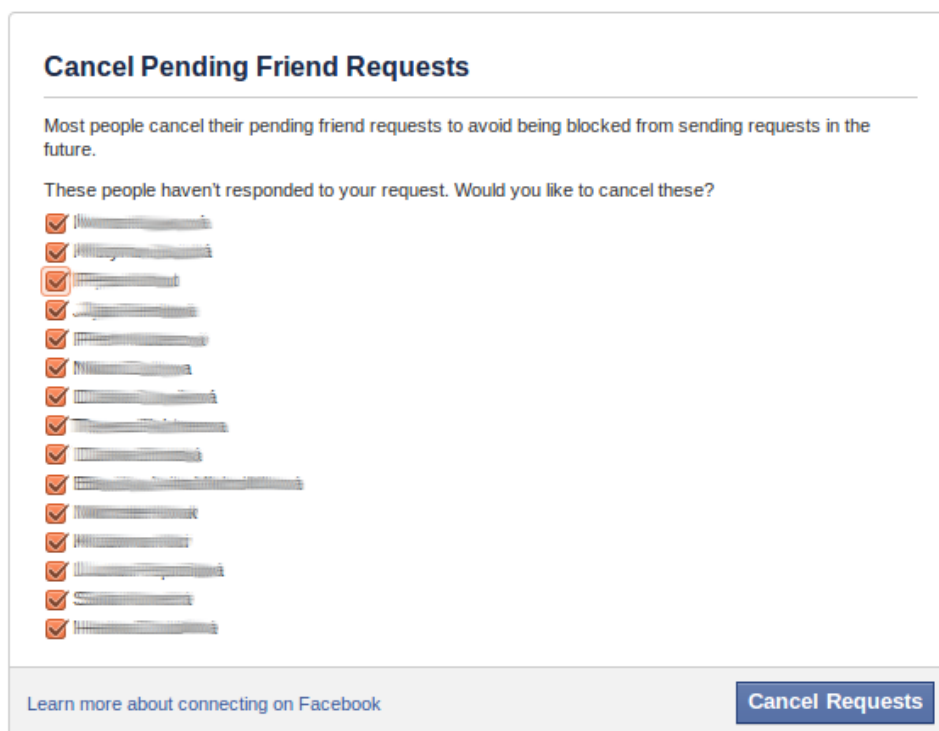
Mezi další formuláře pro doplnění informací na facebooku patří možnost výběru URL adresy facebookového profilu. Jde o jednoznačný identifikátor každého uživatele. Adresa profilu pak vypadá následujícím způsobem: `http://www.facebook.com/jan.novak`. Tento identifikátor se používá i pro e-mailovou adresu na facebooku, kterou má každý účet. Například: `jan.novak@facebook.com`. Jakékoli zprávy odeslané na tuto e-mailovou adresu se zobrazí mezi zprávami na facebooku. Pokud si uživatel žádný identifikátor nevybere, získá adresu vygenerovanou od facebooku spojením jména příjmení a náhodného čísla, například: `jan.novak.145`.

Občas se u některých elementů na facebooku objeví malé vyskakovací okno s nápovědou, která popisuje možnosti na facebooku. Jedná se například o oznámení, že lze na facebook nahrávat fotky a odkazy pouhým přetažením do textového pole pro zadávání statusů a podobně.

U jednoho experimentálního účtu se mi po přihlášení 24. dubna 2013 objevilo okno s oznámením, že byly z tohoto účtu zasílány žádosti o přátelství lidem, které daný účet nezná. Pravděpodobně došlo k tomu, že většina

zaslaných pozvánek na facebooku byla zamítnuta a lidé nepotvrdili, že by daný účet znali i v reálném životě. Facebook v tomto okně radí přidávat si lidi, jako jsou: rodinní příslušníci, spolužáci nebo spolupracovníci a lidi, které známe z reálu. Je potřeba odškrtnout potvrzení, že již nebudou rozesílány žádosti o přátelství lidem, které neznáme z reálu, jinak může být daný účet zablokován.

Na další obrazovce se objevil seznam zaslaných nepřijatých a nepotvrzených žádostí o přátelství s možností jakékoli žádosti zrušit (jak je vidět na obrázku 4.5).

The image shows a Facebook interface for managing pending friend requests. At the top, the title is "Cancel Pending Friend Requests". Below it, a message states: "Most people cancel their pending friend requests to avoid being blocked from sending requests in the future." The main section is titled "These people haven't responded to your request. Would you like to cancel these?" and contains a list of 15 entries, each with a red checkmark icon and a blurred profile picture. At the bottom, there is a link "Learn more about connecting on Facebook" and a blue button labeled "Cancel Requests".

Obrázek 4.5: Formulář na facebooku pro zrušení nepřijatých a nepotvrzených žádostí o přátelství

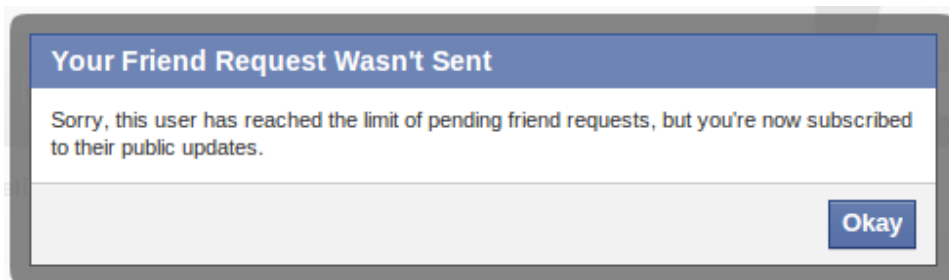
Zrušil jsem tedy všechny tyto žádosti a zaslal jsem pět nových žádostí jako u každého experimentálního účtu. Při příštím přihlášení 28.4.2013 se mi zobrazil formulář s nutností opsat kontrolní kód a na další obrazovce se objevilo oznámení o zablokování účtu s formulářem pro zadání telefonního čísla (naprosto stejný jako při zakládání více účtů ze stejné IP adresy).

#### 4. PRŮBĚH EXPERIMENTU

---

Pravděpodobně došlo odebráním čekajících žádostí o přátelství k snížení důvěryhodnosti profilu, jelikož účet obsahoval málo přátel na facebooku. V době posledního přihlášení před zablokováním obsahoval účet osm potvrzených přátel. Z nichž někteří si mohli do dalšího přihlášení tento účet odebrat. Tím došlo k zablokování daného profilu po 26 dnech od spuštění experimentu. Po zablokování jsem se již k danému účtu nepokoušel přihlašovat. [todo] Otázkou zůstává, zda by došlo k zablokování profilu i bez smazání žádostí o přátelství.

U jedné odeslané žádosti o přátelství se mi podařilo narazit na další omezení facebooku. Maximální počet přátel, které může každý účet na facebooku mít je pět tisíc. S tím se pojí další limit, jeden účet může mít na facebooku maximálně jeden tisíc nevyřízených (nezamítnutých nebo nepotvrzených) žádostí o přátelství. Při jedné ze zaslaných žádostí o přidání mezi facebookové přátele se mi zobrazilo vyskakovací okno (viz obrázek 4.6), že kvůli tomuto omezení nebyla žádost o přátelství zaslána. Bohužel s tímto nemůžeme nic udělat, musíme počkat, až účet jemuž byla žádost zaslána tyto nevyřízené žádosti o přátelství potvrdí nebo zamítne.



Obrázek 4.6: Informace o dosažení maximálního počtu nevyřízených žádostí o přátelství na facebooku

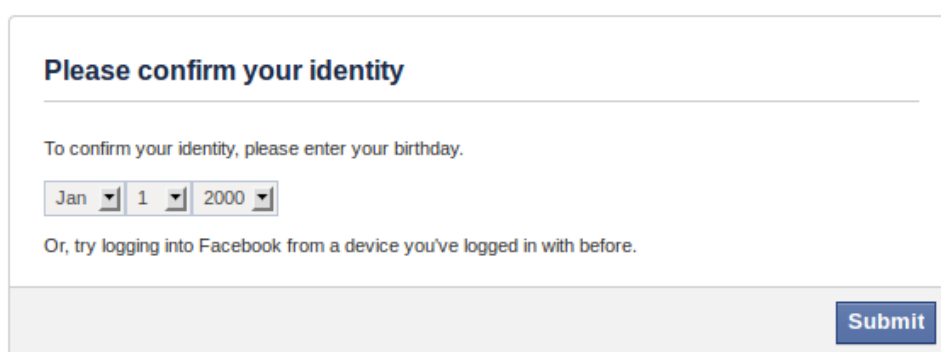
Ve většině případů jsem experimentální účty navštěvoval z domova, z mé domácí IP adresy. Ve dvou dnech jsem se ale připojoval k účtům z jiného místa a IP adresa byla odlišná od mé domácí adresy. Při tomto připojování dne 26. dubna 2013 se mi u jednoho z účtů ihned po přihlášení zobrazilo oznámení, že je účet dočasně zablokovaný. Po pokračování na následující obrazovku se mi zobrazil formulář pro ověření identity. Celý formulář je k vidění na obrázku 4.7.

Formulář obsahoval pole pro zadání data narození daného profilu. Naštěstí jsem měl pro každý experimentální účet poznamenané datum narození,



teré bylo vyplněno při registraci. Po zadání správného data narození už byl účet odblokovaný a bez jakýchkoli omezení.

Jak je na obrázku formuláře vidět, dočasné zablokování bylo pravděpodobně způsobeno přihlašováním k facebooku z jiné IP adresy. Pod formulářem je napsáno, že pro odblokování účtu se lze místo zadávání data narození přihlásit ze zařízení (IP adresy), ze kterého se daný účet přihlašoval naposledy.

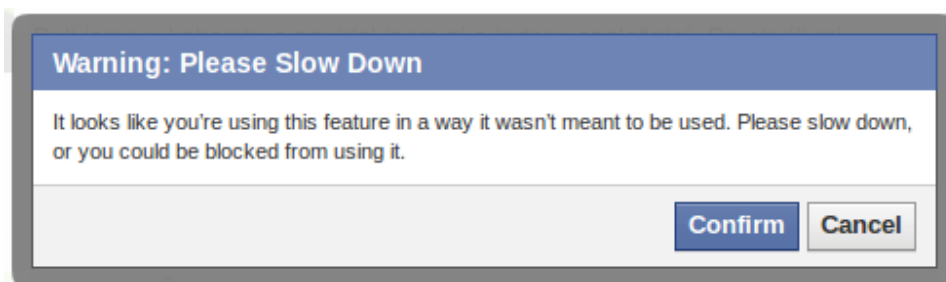
The image shows a web form titled "Please confirm your identity". Below the title, it says "To confirm your identity, please enter your birthday." There are three dropdown menus for the birthday: "Jan", "1", and "2000". Below these, it says "Or, try logging into Facebook from a device you've logged in with before." At the bottom right of the form is a blue "Submit" button.

Obrázek 4.7: Formulář pro ověření totožnosti zadáním data narození na facebooku

Při vývoji aplikace jsem používal program Selenium WebDriver k ovládání webového prohlížeče. Při přidávání like ke statusům na zdi jsem vždy musel stáhnout nejnovější statusy a zjistit, kolik mají jednotlivé příspěvky uděleno like. Po stažení všech příspěvků a vyhodnocení, které mají nejvíce like jsem používal Selenium WebDriver k okamžitému udělení like k vybraným statusům.

Při programování aplikace jsem ale jednou omylem vložil část pro udělování like ke statusům, což bylo v internetovém prohlížeči realizováno klikem na tlačítko, do špatné sekce v programu. Klik na tlačítko pro udělení like se kvůli tomuto omylu opakoval vždy několikanásobně v rychlém sledu za sebou.

Facebook mi kvůli těmto rychlým interakcím zobrazil upozornění, které je vidět na obrázku 4.8. Jednalo se o upozornění, že se děje něco, k čemu by docházet nemělo a že může dojít i k zablokování facebookového účtu. Při experimentu jsem si na tento problém již dával pozor a mezi interakcemi



Obrázek 4.8: Facebookové upozornění kvůli příliš rychlé interakci

jsem dělal prodlevy, aby facebook nemohl odhalit automatizaci interakce. Později jsem na toto upozornění již nenarazil.

Nejzajímavější zjištění pro mne nastalo při návštěvě jednoho z profilů. Zjistil jsem totiž, že od posledního navštívení profilu, získal jeden z účtů 48 nových žádostí o přátelství a to 22. dubna 2013. Takovéto množství přidanych přátel jsem v žádném případě na začátku experimentu nepředpokládal.

Konkrétně se jednalo o profil se skupinou vyplněnosti 4 a se strategií přidávání profilů s velkým množstvím přátel. Zajímavé na všem bylo, že celkový dosavadní počet přátel profilu byl 18 uživatelů. Jedná se tedy o téměř třinásobné navýšení počtu přátel. Je nutné dodat že naprostá většina těchto žádostí o přátelství pocházela od profilů mužů na facebooku (pouze jeden byl od ženského profilu).

Profily, které tomuto experimentálnímu účtu zaslaly pozvánku byly z různých států světa (USA, Egypt, Thajsko a další). Účty byly velmi různorodé, některé vypadaly na první pohled jako falešné účty vytvořené pouze za účelem přidávání přátel, ale některé profily byly naopak velmi vyplněné a obsahovaly autentické fotografie jedné osoby s několikaletou historií celého profilu, takže se pravděpodobně nejednalo o falešné účty. Nikdy to ale nelze zcela jistě určit.

S největší pravděpodobností facebook tento profil vyhodnotil jako zajímavý pro ostatní uživatele, tím že si ho přidávali na facebooku vlivné osoby (profily s velkým počtem uživatelů). Díky tomu mohl facebook tento experimentální účet doporučovat pro přátelení s ostatními profily a tak mohl získat velké množství žádostí o přátelství.

Dalším faktorem tohoto jevu mohla být poměrně pěkná profilová fotka experimentálního účtu. Je možné, že se odkaz na tento facebookový profil objevil na některé internetové stránce a díky tomu si tento profil přidávali ostatní uživatelé. Na facebooku bohužel není žádná možnost zjištění kolik lidí navštíví facebookový profil a odkud uživatele přicházejí.

Další možností k tomuto rozšíření experimentálního profilu mezi další facebookové uživatele, která mne napadá, je možnost doporučení přátel. Facebook po spřátelení dvou profilů dává každému možnost doporučit novému profilu někoho ze svých přátel. Těmto účtům pak přijde upozornění, že jim byl doporučen facebookový účet k přidání mezi své přátele. Uživatel pak může tomuto účtu poslat žádost o přátelství anebo toto doporučení zamítnout. Je tedy pravděpodobné, že některý ze starších přátel experimentálního profilu doporučil svým přátelům tento profil k zaslání žádosti o přátelství.

Při dalším přihlášení, tedy 25. dubna 2013 měl tento účet dalších 28 nových žádostí o přátelství, což je méně než v předchozím případě. Ale následující přihlášení 30. dubna 2013 se u stejného účtu objevilo 71 nových žádostí. Tento jev tedy nebyl náhodný, ale tendence přidávání přátel k tomuto účtu se časem ještě zvyšovala.

Jak k takovému vysokému nárůstu přátel mohlo dojít asi nepůjde dostupnými prostředky nikdy stoprocentně zjistit. S největší pravděpodobností půjde o kombinaci výše zmíněných faktorů na facebooku i mimo něj.



## Závěr

### 5.1 Možné způsoby rozšíření a pokračování v experimentu

#### 5.1.1 Automatizace zakládání facebookových účtů

Teoreticky by bylo možné celý proces zakládání facebookových účtů plně zautomatizovat. E-mailové adresy pro registrované účty by šlo jednoduše zařídit na vlastní doméně nastavením doménového koše. To znamená, že veškeré e-maily, které byly odeslané na adresu naší domény budou doručeny na jednu konkrétní e-mailovou adresu. Získáme tak teoreticky neomezené množství e-mailů k registraci na facebook. Otázkou je, zda by facebooku nevadila stejná doména, ale některé firmy a školy vytvářejí pro své zaměstnance a studenty e-mailové adresy s doménou instituce, takže by to nemělo ničemu vadit.

Dále by se musel vyřešit problém s registrací z jedné IP adresy denně. Toto omezení by šlo obejít použitím několika spolehlivých proxy serverů<sup>25</sup>, přes které by byly facebookové účty registrovány. Množství registrovaných účtů na facebooku denně by se odvíjelo od počtu využívaných proxy serverů. Čím více, tím lépe.

Pravděpodobně by se musel vyřešit i problém s opisováním kontrol-

---

<sup>25</sup>Proxy server - server, prostřednictvím kterého se lze připojit k internetu, výhodou je že není vidět naše IP adresa, ale IP adresa proxy serveru, seznam proxy serverů lze nalézt na internetu, například - <http://www.hidemyass.com/proxy-list/>

ního CAPTCHA kódu, pokud facebook detekuje podezřelou robotickou činnost. Na internetu ale fungují služby, které umí za poplatek řešit opisování CAPTCHA obrázků<sup>26</sup>. Vše funguje automaticky pomocí OCR (Optical character recognition - optické rozpoznávání znaků) anebo lidským opisováním textů. Výhodou takovýchto služeb je, že si lze vše pronajmout a nemusíme se starat o nic jiného.

Na druhou stranu, by se ani problém s nutností řešit opisování kontrolních kódů z obrázků řešit nemusel. Ve většině případů totiž po zobrazení stránky s opsáním kontrolního CAPTCHA kódu následovala stránka s nutností zadat telefonní číslo a přes tento krok by bylo příliš složité se hromadně dostávat.

Pravděpodobně by bylo potřeba řešit i jiné problémy spojené s automatickým zakládáním facebookových profilů, které by se objevily při hlubším bádání nad problémem. Každopádně se ale určitě jedná o velmi zajímavé téma, které by šlo zpracovat více do hloubky. Protože by rozsah takového rozpracování pravděpodobně vydal na samostatnou diplomovou práci. Pro svůj experiment jsem potřeboval pouze omezený počet facebookových účtů, takže jsem všechny účty zakládal ručně.

## 5.2 Vyhodnocení

- Grafy - Vyhodnocení

## 5.3 Možnosti pokračování experimentu

Pro naši aplikaci nebylo potřeba uchovávat texty komentářů u statusů, takže se tento problém nemusel řešit. Ukládá se pouze počet komentářů, který lze získat zpracováním zobrazeného textu. Ukládání textu komentářů by mohlo být užitečné, pokud by bylo možné zjistit citové zabarvení komentářů (negativní/pozitivní/neutrální). Pro takovéto analýzy jsou ale potřeba pokročilé algoritmy pro práci s textem a pro kvalitní rozpoznání by bylo potřeba delší úseky textů. Takovéto analýzy se využívají pro sledování nálady ve společnosti (například při volbách apod.) anebo pro sledování spokojenosti s určitým produktem nebo brandem (značkou) na sociálních sítích.

---

<sup>26</sup><http://decaptchablog.com/decaptcher-services>

Hlavní cíl experimentu ale byl odlišný a počet facebookových účtů, které byly do experimentu zahrnuty nebyl dostatečně velký (hlavně počet facebookových přátel účtů), aby měly výsledky vypovídající hodnotu. Každopádně by se při větším množství účtů zahrnutých do experimentu mohlo podařit získat vypovídající masu lidí, kde by mohly být výsledky zajímavé. Zatím jsou takováto data získávána pouze z veřejně dostupných zdrojů anebo ze stránek na facebooku a takováto data mohou být zkreslená (omezením vzorku jen na určitou skupinu uživatelů). Jedná se o jeden ze směrů, kterým by se dal experiment dále rozšířit a pokračovat v něm v budoucnu.

Kromě získávání nálady ve společnosti by bylo možné i ovlivňovat ostatní profily. Vkládáním pozitivních statusů o určitém příspěvku by mohlo sloužit jako zajímavá reklama. Lidé jsou odmítaví ke klasickým formám reklamy (letáky, televizní reklamy, apod.), ale pokud by se jednalo o pozitivně laděné sdělení, které by nevypadalo na první pohled jako reklama, mohlo by se jednat o zajímavou možnost ovlivnit mínění facebookových uživatelů. Teoreticky by nemuselo být sdělení pozitivní, někdy by mohla stačit zmínka na facebooku a uživatelé by se díky sociální aktivitě (sdílení, dávání like a komentování) postarali o šíření mezi další uživatele internetu. Reklama by teoreticky mohla být i negativní a bylo by zajímavé porovnávat, zda by se více šířily pozitivní nebo negativní statusy. V dnešní době už existují služby pro placené blogové články na míru (Například [plac1a.cz](http://plac1a.cz)). Podobně by to mohlo fungovat i u facebooku.

Facebook na podzim roku 2012 zavedl možnost placení za zviditelnění statusů. Na facebooku se nyní nezobrazují všechny statusy od vašich přátel, ale jen vybrané statusy, které facebook považuje pro vás za nejhodnotnější a nejkvalitnější. Vše jde nastavit a ovlivnit skupinami do kterých si přátele zařadíte (algoritmy pro zobrazování příspěvků nejsou veřejné). Nikdy ale nemáte jistotu, že váš příspěvek uvidí všichni (nebo určití) přátelé. Za poplatek \$6.99 (přibližně 140 Kč - duben 2013) pak uživatelé facebooku mohou zviditelnit své příspěvky, aby se zobrazily většímu počtu kamarádů. Takto lze promovat příspěvky i u facebookových stránek, kde tato funkce dává větší smysl.





---

## Literatura

- [1] effectix.com: STATISTIKY SOCIÁLNÍCH SÍTÍ. 2013. Dostupné z: <http://www.doba-webova.com/cs/statistiky-pro-socialni-sit>
- [2] Facebook developers: Graph API. Duben 2013. Dostupné z: <https://developers.facebook.com/docs/reference/api/>
- [3] Facebook developers: Like Box. Duben 2013. Dostupné z: <https://developers.facebook.com/docs/reference/plugins/like-box/>
- [4] Facebook help center: How many people can I invite to an event? 2013. Dostupné z: <https://www.facebook.com/help/202545109787461/>
- [5] Facebook Investor Relations: Facebook Reports First Quarter 2013 Results. Květen 2013. Dostupné z: <http://investor.fb.com/releasedetail.cfm?ReleaseID=761090>
- [6] Facebook Newsroom: Improvements to Timeline. Březen 2013. Dostupné z: <http://newsroom.fb.com/News/584/Improvements-to-Timeline>
- [7] Inside Facebook: Facebook puts limits on event invites. Duben 2013. Dostupné z: <http://www.insidefacebook.com/2013/04/09/facebook-puts-limits-on-event-invites/>
- [8] Opera Developer News: 300 million users and move to WebKit. Únor 2013. Dostupné z: <http://my.opera.com/ODIN/blog/300-million-users-and-move-to-webkit>

- [9] The Chromium Blog: Blink: A rendering engine for the Chromium project. Duben 2013. Dostupné z: <http://blog.chromium.org/2013/04/blink-rendering-engine-for-chromium.html>
- [10] The Facebook Blog: Welcome to Your New Home Page. Březen 2009. Dostupné z: <https://blog.facebook.com/blog.php?post=59195087130>
- [11] The Facebook Blog: Introducing Your New Navigation. Únor 2010. Dostupné z: <https://blog.facebook.com/blog.php?post=287459122130>
- [12] The Facebook Blog: See the Messages that Matter. Únor 2011. Dostupné z: <https://blog.facebook.com/blog.php?post=452288242130>
- [13] The Facebook Blog: Timeline: Now Available Worldwide. Prosinec 2011. Dostupné z: <https://blog.facebook.com/blog.php?post=10150408488962131>
- [14] Yazan Boshmaf, Ildar Muslukhov, Konstantin Beznosov, Matei Ripeanu: The Socialbot Network: When Bots Socialize for Fame and Money. Prosinec 2011. Dostupné z: [http://www.acsac.org/2011/openconf/modules/request.php?module=oc\\_program&action=summary.php&id=175](http://www.acsac.org/2011/openconf/modules/request.php?module=oc_program&action=summary.php&id=175)

## Slovník použitých pojmů a zkratek

**HTML** - Hypertext markup language - zdrojový formát vzhledu webových stránek, který je zpracováván internetovým prohlížečem (<http://en.wikipedia.org/wiki/HTML>).

**Like skupiny na facebooku** - Like = Líbí se mi - způsob připojení se ke skupně (Group) nebo stránce (Page) na facebooku

**Like statusu** - Like = Líbí se mi - způsob vyjádření, že se uživateli líbí status na facebooku, bez nutnosti status komentovat. Jedná se o modré tlačítko, na kterém je znázorněn palec nahoru.

**Přidat se do vztahu na facebooku** - Na facebooku lze v informacích o sobě vyplnit i svůj partnerský a rodinný vztah, například: ve volném vztahu, zadaný, v komplikovaném vztahu, ženatý, rozvedený a další. K tomuto stavu můžeme přiřadit i konkrétní osobu s účtem na facebooku, pokud dotyčná osoba vše schválí, zobrazí se pak v informacích odkaz na účet se kterým je osoba ve vztahu.

**Přítel na facebooku** - Jedná se o virtuální kamarádství mezi dvěma lidmi, které funguje tak, že oba zúčastnění musí přátelství potvrdit. Prakticky vše probíhá tak, že můžeme komukoli na facebooku zaslat pozvánku k přátelství, dotyčný pak může pozvánku přijmout anebo odmítnout.

**Profil na facebooku**

## A. SLOVNÍK POUŽITÝCH POJMŮ A ZKRATEK

---

**(internetový) prohlížeč = browser** - Program pro zpracování a zobrazování internetových stránek. Zpracovává takzvaný HTML kód webové stránky.

**REST** - Representational state transfer - Architektura rozhraní pro komunikaci s webovými službami (Web services), postavený nad HTTP protokolem  
([https://en.wikipedia.org/wiki/Representational\\_state\\_transfer](https://en.wikipedia.org/wiki/Representational_state_transfer)).

**Status** - Příspěvek na svou facebookovou zeď od uživatelů (Facebook nabádá k odpovědi na otázku: „What is on your mind?“ = „Co se vám honí hlavou?“).

**Skupina = stránka na facebooku**

**Štouchnout na facebooku (Poke)** -

**Zeď na facebooku** - V terminologii facebooku se zeď používá k označení seznamu příspěvků svých a svých kamarádů.

**URL**

## Obsah přiloženého CD

	readme.txt .....	stručný popis obsahu CD
	exe .....	adresář se spustitelnou formou implementace
	src	
	impl .....	zdrojové kódy implementace
	thesis .....	zdrojová forma práce ve formátu $\text{\LaTeX}$
	text .....	text práce
	thesis.pdf .....	text práce ve formátu PDF
	thesis.ps .....	text práce ve formátu PS