

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

21. dubna 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 21. dubna 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

Sem doplňte ekvivalent abstraktu Vaší práce v angličtině.

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

Úvod	1
1 Analýza a návrh	3
1.1 Aplikace	3
1.2 Možnosti komunikace	3
1.3 Výhody a nevýhody použití selenia	8
1.4 Detailní popis programu Selenium	9
1.5 Shrnutí, výhody a nevýhody použití programu Selenium Web-Driver	11
2 Realizace	13
2.1 První generace facebookových účtů	13
2.2 Příprava experimentu	13
2.3 Problémy se zakládáním facebookových účtů	16
2.4 Strategie přidávání přátel k falešným účtům	17
2.5 Volba programovacího jazyka a databáze	18
2.6 Rozdělení aplikace	19
2.7 Stahování informací z facebookových profilů	19
2.8 Nahrávání dat na facebookové profily	22
Závěr	25
A Slovník použitých pojmů a zkratk	27
B Obsah přiloženého CD	29

Seznam obrázků

1.1	Schéma přihlašování k facebooku pomocí OAuth	5
2.1	Zadávání telefonního čísla na facebooku	14
2.2	Potvrzení přidání osoby do přátel na facebooku	23

Úvod

- motivace - prinos - zminit paper - fb, problemy, vek ... - zamereni na facebook - cile prace - prinos - cz prostredi - fb vs tw - vice moznosti na FB, znamejsi mezi mladymi

Analýza a návrh

1.1 Aplikace

- celkový popis fungování aplikace

1.2 Možnosti komunikace

Pro interakci s facebookem používá většina uživatelů webové rozhraní, které je na webové adrese www.facebook.com. Toto rozhraní je upraveno pomocí techniky takzvaného responzivního designu¹ 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 staré verzi. Dříve se kvůli změnám rozhraní nejednou zvedla velká vlna nevole (například změna osobních profilů v roce 2009², změna zobrazení zdi v roce 2010³, zavedení timeline v roce 2011⁴ anebo přepracování zpráv v roce 2012⁵). Nyní se kvůli těmto problémům snaží facebook oznamovat předem

¹http://en.wikipedia.org/wiki/Responsive_web_design

²<http://www.zive.cz/bleskovky/facebook-tento-tyden-ceka-velka-zmena-priblizi-se-twitteru/sc-4-a-146045/default.aspx>

³<http://www.zive.cz/bleskovky/facebook-opet-v-novem-kompletni-zmena-okoli-zdi/sc-4-a-150779/default.aspx>

⁴<http://computerworld.cz/internet-a-komunikace/facebook-chrli-dalsi-novinky-rozsiruje-moznosti-profilu-43866>

⁵<http://www.cnews.cz/clanky/facebook-se-chysta-prepracovat-zpravy-budou-se-vice-podobat-e-mailu>

navenek výhody a nové možnosti úprav designu a zapíná nové funkce postupně s možností dočasně používat starou verzi.

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**⁶. Jedná se o RESTové⁷ rozhraní webové služby. 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 aplikaci schválit a povolit oprávnění, jaká data může aplikace číst a měnit.

Celý proces přihlašování k facebooku používá OAuth autorizaci^{8,9}, při které je uživatel aplikace přesměrován na stránky facebooku, kde se musí přihlásit a při prvním použití aplikace povolit přístup této aplikaci k uživatelskému profilu. 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 1.1). 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 API, liší se 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 vždy jednat o aplikace pro interakci s facebookem, velké množství aplikací (například hry a podobné aplikace) data z facebooku pouze čtou a získávají informace o přátelích uživatele a podobně. Takovéto aplikace jsou pak zobrazeny v okně facebooku (vložení stránky pomocí tagu iframe) a mohou používat facebookové prvky (tlačítka, vyskakovací okna, ...¹⁰), takže méně zkušený uživatelé si ani neuvědomí, že používají aplikaci, která není kompletně pod kontrolou facebooku.

Facebook nepodporuje žádnou jinou možnost, jak aplikaci ovládat kromě

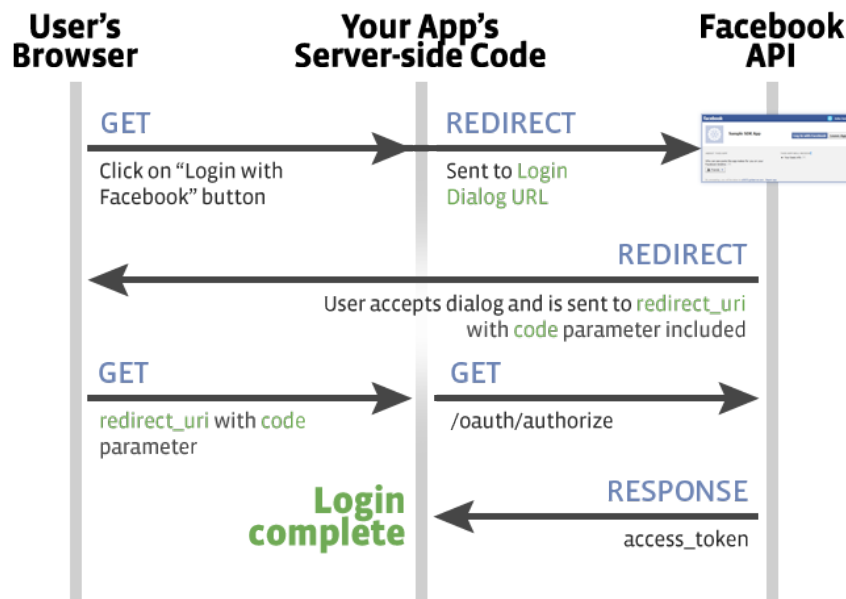
⁶<https://developers.facebook.com/docs/reference/apis/>

⁷http://en.wikipedia.org/wiki/Representational_state_transfer

⁸<http://oauth.net/>

⁹<https://developers.facebook.com/docs/howtos/login/server-side-login/>

¹⁰<https://developers.facebook.com/docs/plugins/>



Obrázek 1.1: Schéma přihlašování k facebooku pomocí OAuth

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

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 like-jackingem ¹¹ často zneužívána k podvodnému navyšování fanoušků skupin.

Na internetu se běžně objevují nabídky na nákup fanoušků do facebookových skupin (viz tabulka 1.1). 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/>). Možnosti jsou různé od nákupu statusů až po přidání do vztahu. Nabízí se otázka, kolik

¹¹Variace clickjackingu, 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>

1. ANALÝZA A NÁVRH

účtů, které služby nabízí, je pravých a jaká je záruka, že po převodu peněz účty nezmizí. Každopádně se jedná o zajímavý business.

Tabulka 1.1: Příklady webových stránek sloužících pro nákup Like do skupin na facebooku

Stránka	Cena
http://get-likes.com/facebook-likes-store/	1000 fanoušků za 30\$
http://fblikesmart.com/	1000 fanoušků za 50\$
http://officialfacebooklikes.com/	1000 fanoušků za 80\$

Facebook proto zrušil možnost dávat stránkám like skrze Graph API. S dalším zneužíváním funkcí facebook API se může stát, že budou dále zakázány i některé další funkce. Využití facebook API je tedy poněkud omezené a nejisté. Hlavní nedostatky facebook Graph API pro experimentální aplikaci jsou:

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

Kvůli těmto omezením jsem si tedy pro experimentální aplikaci vybral raději možnost obsluhovat facebook pomocí webového rozhraní za použití specializovaných programů a nástrojů.

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

Největším problémem tohoto přístupu je skutečnost, že facebook používá velké množství AJAXových¹² požadavků a dynamických JavaScriptových oken, které by mohly způsobit komplikace. Ztížilo by to především získávání

¹²asynchronní načítání částí stránek pomocí JavaScriptu anebo XML - [http://en.wikipedia.org/wiki/Ajax_\(programming\)](http://en.wikipedia.org/wiki/Ajax_(programming))

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 stáří příspěvků.

Mezi další nevýhody používání HTTP komunikace patří to, že facebook pro některé části HTML stránek generuje odlišný HTML kód (mění se například atributy ID u HTML značek a podobně). Nejvíce jsem se s těmito změnami setkal na hlavní stránce facebooku, kde pravděpodobně probíhá takzvané A/B uživatelské testování stránek¹³. Uživatelům se zobrazují různé grafické verze stránky a následně se měří rozdíly úspěšnosti měřených veličin (počet prokliků/počet nákupů) pro jednotlivé verze.

V poslední době se začal rozšiřovat nový protokol SPDY¹⁴, který slouží jako nadstavba nad tradičním HTTP protokolem pro rychlejší načítání stránek. Facebook tento protokol již na svých serverch používá a mohly by tak vzniknout problémy při HTTP komunikaci i když by měl být protokol SPDY teoreticky zpětně kompatibilní, nemuselo by vše být bez problémů.

Kvůli těmto problémům jsem se tedy rozhodnul využít raději hotové řešení a zabývat se komunikací s webovými stránkami na vyšších vrstvách.

Jako nejjednodušší z možností se nasklo 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 je především v počtu a kvalitě implementovaných specifikací a funkcionalit.

Programy pro zobrazování webových stránek se liší především implementovaným jádrem webového prohlížeče (Blink¹⁵ - Google Chrome, Gecko¹⁶ - Mozilla Firefox, Trident¹⁷ - Microsoft Internet Explorer, WebKit¹⁸ - Apple Safari, jiná nebo vlastní implementace). Jádro ovlivňuje to, jak bude webová stránka vykreslena. Mohou se lišit i způsoby vykreslení stejné webové

¹³http://en.wikipedia.org/wiki/A/B_testing

¹⁴<http://en.wikipedia.org/wiki/SPDY>

¹⁵Společnost 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))

¹⁶<https://developer.mozilla.org/en-US/docs/Mozilla/Gecko>

¹⁷[http://en.wikipedia.org/wiki/Trident_\(layout_engine\)](http://en.wikipedia.org/wiki/Trident_(layout_engine))

¹⁸<http://www.webkit.org/>

stránky mezi jednotlivými verzemi jádra. Existují i speciálně upravená jádra pro vykreslování internetových stránek na mobilních přístrojích, které často naschvál neimplementují některé pokročilejší funkce kvůli hardwarovým nárokům a rychlosti vykreslování.

Druhým důležitým parametrem takovýchto programů je, zda se webová stránka vykresluje jen v paměti počítače, nebo zda se výsledek vykresluje i na obrazovku. Existují programy, určené především na testování webových stránek, které pouze kontrolují výskyt a pozici jednotlivých elementů, ale nedokáží stránku vykreslit, mají vše uloženo v paměti. Díky tomu jsou ale i rychlejší.

Tabulka 1.2: Příklady programů pro zpracování webových stránek

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

Program selenium vzniknul jako nadstavba nad webovým prohlížečem Mozilla Firefox. Pomocí nainstalovaného pluginu v tomto prohlížeči jej lze ovládat pomocí 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 mohli vzniknout knihovny pro velké množství programovacích jazyků. Toto API Selenia je RESTové, takže se dají na internetu najít i neoficiální knihovny pro různé další jazyky. Dnes již existují k programu Selenium i pluginy, které umožňují místo prohlížeče Firefox používat prohlížeče Chrome, Operu, PhantomJS a další.

1.3 Výhody a nevýhody použití selenia

Hlavní výhoda programu Selenium spočívá v tom, že jsou webové stránky renderovány v reálném prohlížeči a operace, které se v prohlížeči odehrávají máme možnost vidět 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 optimalizovány a zcela jistě jsou v něm i facebookem testovány, pro správné zobrazování a fungování všech částí a v případě nefunkčnosti by facebook všechny nedostatky co nejdříve napravil.

1.4 Detailní popis programu Selenium

Selenium je soubor programů, které slouží primárně k testování internetových stránek přímo v prohlížeči. Původně se jednalo jen o rozšíření (takzvaný plugin) do prohlížeče Firefox, pomocí kterého je možné nahrát konkrétní uživatelské chování (například: zadat konkrétní text do uživatelského pole, kliknout na tlačítko, otevřít novou stránku a podobně). Všechny akce se nahrávají a ukládají do souboru (.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 IDE¹⁹.

Díky oblibě tohoto pluginu se postupem času utvořil kolem této technologie celý ekosystém pro automatizované testování pomocí webového prohlížeče. Do tohoto ekosystému patří programy: **Selenium IDE** (popsaný 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 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 informace o jednotlivých produktech lze nalézt na oficiálních stránkách jednotlivých produktů: <http://docs.seleniumhq.org/projects/>.

Pro můj experiment 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 jakékoliv lidské interakce, takže tyto úkony mohou být opakovatelné a prováděné na různých počítačích.

Celý mechanismus funguje na jednoduchém principu. Selenium WebDriver

¹⁹<http://docs.seleniumhq.org/projects/ide/>

je v podstatě jen jeden `.jar` soubor²⁰. Jedná se o webový server, který naslouchá na určeném portu našeho počítače a vystavuje RESTové rozhraní k ovládání prohlížeče. Z našeho programu se pak musíme připojit k webovému serveru Selenia a posíláme na něj požadované akce, které se mají v prohlížeči vykonat. Jelikož je webové rozhraní seleina nezávislé na programovacím jazyku, existují knihovny pro mnoho jazyků a je jednoduché dopsat si vlastní knihovnu. Program Selenium můžeme ovládat i pomocí nástrojů pro práci s HTTP protokolem (například `cURL`, `HTTPie`, ...). Celé rozhraní je detailně popsáno na stránkách projektu²¹.

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: 1.1) a pohybovat se po stránce (pozor, neplést s ovladači u hardware, nazývají se také *drivery*, ale jedná se o něco zcela jiného). Původně dokázal Selenium WebDriver ovládat pouze prohlížeč Mozilla Firefox, ale postupem času vznikly *drivery* 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 (defaultní prohlížeč na mobilních telefonech s operačním systémem Android), iPhone (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 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

Listing 1.1: Příklad otevření url v seleniu pomocí `CURL`

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

²⁰spustitelný program napsaný v programovacím jazyce Java

²¹<https://code.google.com/p/selenium/wiki/JsonWireProtocol>

1.5. Shrnutí, výhody a nevýhody použití programu Selenium WebDriver

```
# 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'}"
```

Vývojáři prohlížeče Opera vydali 12. února 2013 článek²² o tom, že budou pro svůj prohlížeč do budoucna používat jádro WebKit, které používají prohlížeče Chrome a Safari. Pár měsíců po tomto oznámení vydali vývojáři prohlížeče Google Chrome 3. dubna 2013 na svém webu článek²³ o tom, že vytvoří vlastní vykreslovací jádro Blink²⁴. Bude se jednat o fork doposud používaného opensource jádra WebKit. Bude zajímavé sledovat, jak se situace nakonec vyvine, nicméně funkčnost programu Selenium by to nemělo omezit.

1.5 Shrnutí, 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 k zpracování webových stránek je, že se stará o všechnu práci spojenou s HTTP protokolem, vykreslováním i ovládáním webových stránek a zpracováním všech skriptů ve stránce.

Selenium WebDriver nám umožňuje přistupovat k výsledným vyrenderovaným elementům prohlížeče a to jak textově, tak i k HTML zdrojákům 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 tedy zabývat nízkourovňovými záležitostmi spojenými s fungováním internetu. Dále můžeme použít různé vykreslovací programy a webové prohlížeče a to včetně programů určených pro mobilní

²²<http://my.opera.com/ODIN/blog/300-million-users-and-move-to-webkit>

²³<http://my.opera.com/ODIN/blog/300-million-users-and-move-to-webkit>

²⁴<http://www.chromium.org/blink>

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 a doplňků.

Výhody využití programu Selenium WebDriver

- hojně používaný program
- možnost změny vykreslovacího programu
- velké množství vykreslovacích programů
- možnost ručního ovládání browseru
- velké množství návodů a zdrojů na internetu
- RESTové rozhraní pro ovládání browseru
- možnost přistupovat k HTML 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 stránky

Realizace

2.1 První generace facebookových účtů

Před samotným startem celého experimentu jsem potřeboval otestovat, jak přibližně bude vše fungovat a jak se budou účty přibližně chovat. Založil jsem proto na facebooku pět testovacích účtů, které jsem nazval **první generace**. Pomocí těchto účtů jsem zjišťoval omezení i zádrhele celého experimentu a jak by vše mělo fungovat. Pomocí těchto účtů jsem vyvíjel aplikaci pro automatické stahování a nahrávání dat na facebook (více podrobně bude celý mechanismus popsán později).

Pomocí první generace jsem ověřil, že facebook nekontroluje, pokud se z jedné IP adresy přihlašuje více účtů. Je to i logické, jelikož v institucích jako jsou školy, knihovny a podobně se může k facebooku přihlašovat více účtů, které jsou k internetu připojeny skrze jednu

2.2 Příprava experimentu

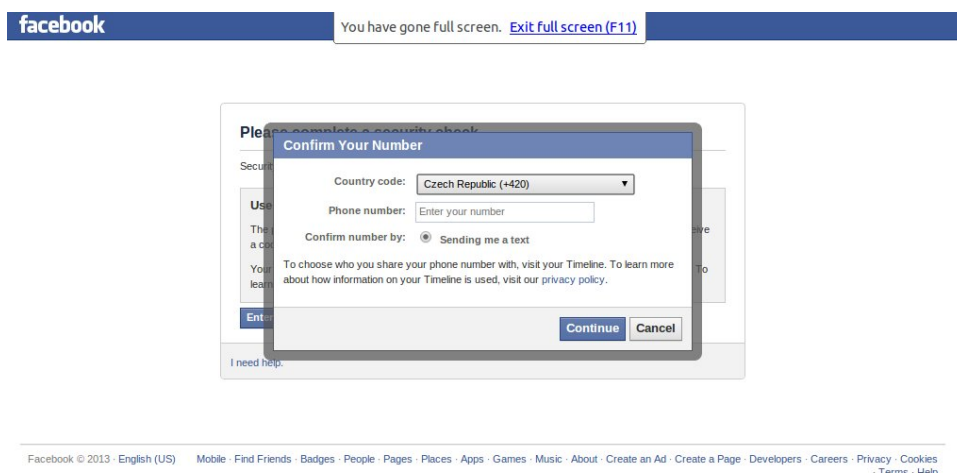
2.2.1 Zakládání e-mailových adres

Před samotným experimentem bylo potřeba založit facebookové účty pomocí kterých se bude experiment realizovat. Zakládání falešných účtů bylo poměrně složité, jelikož jsem chtěl původně spustit celý experiment v jeden den. Facebook při vytváření nového účtu kontroluje vždy IP adresu²⁵ počítače a nedovolí ze stejné IP adresy založit více než jeden účet denně.

²⁵IP adresa - číselná adresa počítače, pomocí které se připojuje k internetu

2. REALIZACE

Pokud se někdo pokusí založit více než jeden účet, zobrazí se na facebooku obrazovka s nutností opsat kontrolní CAPTCHA²⁶ obrázek a po úspěšném opsání facebook zobrazí stránku (viz obrázek 2.1) 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 složitě shání a jedná se o osobní údaj, který jsem nechtěl facebooku sdělovat.



Obrázek 2.1: Zadávání telefonního čísla na facebooku

Nejdříve bylo potřeba založit e-mailové adresy, sloužící k vytvoření falešných účtů na facebooku. Všechny e-mailové adresy byly založeny na české doméně a měly koncovku **.cz**, protože jsem chtěl provést experiment na uživatelích facebooku z České republiky, aby bylo vše věrohodnější. 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**.

Pro každý falešný účet na facebooku jsem nejdříve připravil náhodné jméno a příjmení. Hlavním požadavkem na jméno bylo, aby bylo co nejvíce náhodné, ale zároveň i uvěřitelné. 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 stažených facebookových profilů. U některých jmen byl změněn i rod příjmení. Nadále

²⁶CAPTCHA - jedná se o ochranu internetových stránek proti robotům, nejčastěji je realizovaná nutností opisání textu z deformovaného obrázku

jsem pro vytváření jmen použil službu www.zlatestranky.cz²⁷ (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).

Struktura e-mailových adres byla následující: `jmenoprijmeni@domena`. Přednost měla doména `seznam.cz`, pokud byla e-mailová adresa s tímto jménem již zabraná, použil jsem doména `email.cz` a 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 potřeba aktivovat pomocí e-mailu. Facebook ihned po registraci zašle na e-mail zadávaný při registraci zprávu s odkazem. Aktivace účtu probíhá otevřením ověřovacího odkazu z e-mailu.

Facebook poté na zadaný e-mail posílá upozornění na různé aktivity, které se na facebooku dějí (například žádost o přátelství, zaslání zprávy apod.). Typy zasílaných zpráv z facebooku lze nastavit v uživatelském nastavení. Pro experiment byly e-mailové adresy důležité pouze pro povolení facebookového účtu.

2.2.2 Internetové služby poskytující dočasné e-mailové schránky

Před začátkem experimentu jsem uvažoval nad použitím některé ze služeb pro dočasné 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 jen po omezeně dlouhou dobu a všechny e-mailové zprávy, které jsou odeslány na tuto adresu se zobrazí přímo na navštívené stránce (pouze pro návštěvníka, pro kterého byla daná e-mailová adresa vygenerována). Příklady dočasných e-mailových služeb: <http://10minutemail.com/>, <http://mailinator.com/>, <https://meltmail.com/> a další.

²⁷Jedná se o internetový seznam telefonních čísel fyzických a právnických osob v České republice

Zjistil jsem ale, že facebook registrace z takovýchto dočasných e-mailových adres blokuje a po registraci je potřeba zadat telefonní číslo, na které přijde ověřovací kód, stejně jako při registraci více účtů z jedné IP adresy. Také 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 budu mít přístup i později a nebude problém pokud by bylo potřeba dohledat některé informace posílané e-mailem (například zapomenuté heslo nebo důvod zablokování facebookového účtu).

2.3 Problémy se zakládáním facebookových účtů

Zakládal jsem tedy profily na facebooku již dopředu, abych mohl experiment spustit najednou ve stejný čas. Bohužel se mi ale stalo to, že dříve založené účty na facebooku byly neaktivní a když jsem se k nim přihlásil po nějaké 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 má několik přátel 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 spouštění 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.

Teoreticky by bylo možné celý proces zakládání 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.

Dále by se musel vyřešit problém s registrací z jedné IP adresy denně. Toto omezení by šlo obejít použitím několika spolehlivých proxy serverů²⁸,

²⁸Proxy 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

přes které by byly falešné účty registrovány. Rychlost registrovaných účtů na facebooku denně by se odvíjela od počtu využívaných proxy serverů.

Pravděpodobně by se muselo vyřešit i opisování kontrolního CAPTCHA kódu, pokud facebook detekuje podezřelou robotickou činnost. Na internetu ale fungují služby, které umí za poplatek řešit opisování CAPTCHA obrázků²⁹. 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.

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. Já jsem pro svůj experiment potřeboval pouze omezený počet facebookových účtů, takže jsem všechny účty zakládal ručně.

[todo] - etika - seznam uživatelů

2.4 Strategie přidávání přátel k falešným účtům

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 velmi vyplněný. Hlavní motivací pro takovéto rozdělení je, že více vyplněné facebookové profily by měly být pro ostatní návštěvníky věrohodnější.

Faktory ovlivňující vyplněnost profilu:

- profilový obrázek a

nalézt na internetu, například - <http://www.hidemyass.com/proxy-list/>

²⁹<http://decaptchablog.com/decaptcher-services>

- foto - 5 skupin - 5 levelu - ucty se navzajem neznaaji

2.5 Volba programovacího jazyka a databáze

Pro implementaci celé aplikace jsem si vybral programovací jazyk PHP³⁰. Jedná se o dynamicky typovaný programovací jazyk, který se používá především k vytváření webových stránek a internetových aplikací. Tento jazyk jsem si vybral, jelikož s ním mám největší zkušenosti. Díky dynamic-kému typování lze v tomto jazyce vytvořit aplikace velmi rychle. Protože aplikace neobsahuje složitý model a hlavní funkcionalitou je převážně volání RESTových služeb a ukládání dat do databáze, přišel mi tento jazyk jako nejlepší možnost pro implementaci programu tohoto typu.

Jako databázi jsem pro začátek zvolil MySQL³¹, která je často používána ve spojení s jazykem PHP. Do této databáze ukládám informace o založených účtech, se kterými pracuje stahovací program. V této databázi jsou také uloženy různé skupiny uživatelských účtů (například muži, ženy, apod). Pro ukládání data stažených z uživatelských profilů se používá NoSQL (nerelační) databáze MongoDB³². Do této databáze lze ukládat jakákoliv nestrukturovaná data ve formátu JSON³³ a lze s nimi pak nadále manipulovat a provádět nad nimi agregační dotazy. Tato databáze byla vybrána hlavně kvůli tomu, že umí pracovat s nestrukturovanými daty, to znamená, že cokoliv, co získáme z facebooku můžeme uložit. Hodí se to například k ukládání příspěvků ze zdi uživatelů a informací na profilových stránkách, nemusíme se starat o to, že někteří uživatelé mají omezenou viditelnost počtu přátel nebo počtu skupin.³⁴

³⁰Open source programovací jazyk: <http://www.php.net/>

³¹Open source relační databáze, kterou vlastní společnost Oracle, viz <http://www.mysql.com/>

³²Open source dokumentově orientová NoSQL databáze o kterou se stará společnost 10gen - <http://www.mongodb.org/>

³³JSON = JavaScript Object Notation - <http://www.json.org/>

³⁴Facebook v roce 2010 vyvinul vlastní NoSQL databázi nazvanou Cassandra (http://en.wikipedia.org/wiki/Apache_Cassandra), která slouží k vyhledávání na velkém objemu dat. Facebook tuto databázi později nahradil další NoSQL databází nazvanou HBase (<http://hbase.apache.org/>)

2.6 Rozdělení aplikace

Tato aplikace má dvě základní funkce, první z nich je sbírání dat o uživateli facebooku z jejich profilů a druhou funkcí je ovládní fiktivních účtů. Data uživatelů z facebooku slouží k analýze a k dalšímu porovnání, o tom pojednává celá následující kapitola [viz].

2.7 Stahování informací z facebookových profilů

Důležitou součástí aplikace je získávání informací o uživateli 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 kterou lze vykonávat dotazy pro získání jakýchkoliv statistických dat.

O zpracování stránek facebooku se stará Selenium WebDriver, pomocí kterého se načte konkrétní stránka (například zeď anebo přátelé konkrétního uživatele) a 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.

Pro každý námi vytvořený účet na facebooku 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

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 zobrazeny rozdílně. Například zobrazení like u statusů. Pro analýzu statusů potřebujeme ukládat pouze počet like, které status obdržel. Facebook bohužel ale někdy místo počtu liků zobrazuje i jména lidí, kteří like udělili. Stává se tak v případě, že like

2. REALIZACE

udělil, někdo z lidí, které má účet v přátelích anebo pokud je u statusu pouze jeden like.

Pro účely experimentu se ukládá ke statusu pouze počet like, který je pro nás nejdůležitější. V programu se o získání počtu like stará funkce, která sečte počet jmen a počtu like u statusů.

Podobná situace nastává u komentářů a statusů. Pokud status obsahuje dva až čtyři³⁵ nebo méně komentářů, tak se komentáře zobrazí i s textem. Pokud status obsahuje více komentářů, tak se zobrazí text pouze dvou posledních příspěvků v komentářích u statusu a všechny předchozí komentáře se zobrazí 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 prvních řádků komentáře. Pro zobrazení celého textu je potřeba rozkliknout odkaz za textem 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, jen v jedné úrovni a pokud chce někdo reagovat na předchozí komentář, musí označit autora předchozího komentáře pomocí znaku @ a jeho jména jeho profilu na facebooku. U skupin se komentáře zobrazují ve dvou úrovních, 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 Comments³⁶, který nabízí pro ostatní webové stránky na vložení.

Pro ukládání nebylo potřeba uchovávat text komentářů, takže se tento problém nemusel řešit. Ukládá se pouze počet komentářů, který lze získat zpracováním 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

³⁵Počet plně zobrazených komentářů se různí a není dán pevně pro každý status.

³⁶<https://developers.facebook.com/docs/reference/plugins/comments/>

s textem a pro kvalitní rozpoznání by bylo potřeba delší úseky textu. 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.

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. Jedná se o jeden ze směrů, kterým by se dal experiment dále rozšířit a pokračovat v něm v budoucnu.

Mimo 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 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, likování a komentování) postarali o šíření mezi další uživatele internetu. Reklama by mohla být i negativní. V dnešní době už existují placené blogové články na míru. 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. 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 promovat své příspěvky, aby se zobrazily většímu počtu kamarádů. Takto lze promovat příspěvky i u facebookových stránek.

Pro účely experimentu byla u statusů ukládána tato data:

- myWall - zda byl status stažen ze zdi našeho fiktivního účtu nebo ze

zdi někoho z přátel fiktivního účtu

- user - jméno uživatele, který status přidal
- fromUser - od kterého fiktivního uživatele byl status stažen
- status - text statusu
- commentsCount - počet komentářů
- likesCount - počet like
- timestamp - čas vložení statusu

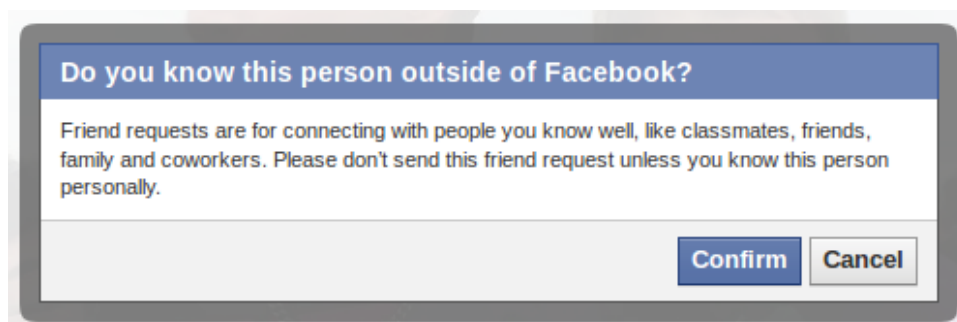
2.8 Nahrávání dat na facebookové profily

Druhá část aplikace pro experiment byla určena pro nahrávání dat na fiktivní facebookové profily. Hlavním cílem pro přidávání dat na fiktivní 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, líkovat a sdílet (poslat na svou zeď pro zviditelnění svým přátelům).

Kromě statusů lze na facebooku komunikovat pomocí zpráv. Systém je podobný jako u e-mailových zpráv s tím rozdílem, že pomocí zpráv lze komunikovat pouze v rámci facebooku. Dalším komunikačním kanálem je reálný chat. Facebookový chat je propojen se zprávami, takže veškeré zprávy, 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 přečetli.

Hlavním cílem interakce na facebooku byla věrohodnost pro ostatní facebookové uživatele. Důležité tedy bylo zautomatizovat přidávání přátel na facebooku. Ze začátku jsem po zaregistrování nového profilu plánoval přidávat pokaždé sedm přátel. Facebook ale při přidávání většího množství přátel (většinou okolo pěti, ale někdy i méně) zobrazil potvrzovací okno 2.2. Ustálil jsem tedy po tomto zjištění počet přidávaných přátel na pět za jedno přihlášení. [todo ...]

= získávání prvních přátel



Obrázek 2.2: Potvrzení přidání osoby do přátel na facebooku

= neznami mezi sebou

Další možností, jak na sebe na facebooku upozornit určitou osobu je možnost někoho štouchnout (poke). Funkčnost tlačítka štouchnout je jednoduchá. Pokud vás někdo štouchne, přijde vám upozornění a můžete štouchnutí opětovat. Štouchnutí je znázorněno ikonou lidské ruky s napřaženým ukazováčkem.

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 a jako zdroj vtipných statusů sloužily stránky: www.vtipy.net, vysmatej.cz a 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 fiktivních účtů.

Ostatní typy možnosti komunikace na facebooku nebyly automaticky zpracovávány. 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 fiktivní účet zná osobu, které byla odeslána žádost o přátelství a občas i štouchnutí.

2. REALIZACE

Facebookvé zprávy nebyly automaticky zpracovávány. Zpracovávání by totiž nepřinášelo přidanou hodnotu k našemu experimentu a zpracovávání textu by bylo složité, jak už bylo zmíněno dříve. 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.

- architektura

Chování uživatele + Metriky

Závěr

- Grafy - Vyhodnocení

[TODO] zdroje/citace??? :)

Slovník použitých pojmů a zkratek

[TODO] seradit

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

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).

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

Postování statusu - Vkládání příspěvku na svou zeď na facebooku.

Stránka na facebooku

Profil na facebooku

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.

A. SLOVNÍK POUŽITÝCH POJMŮ A ZKRATEK

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

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

Štouchnout na facebooku (Poke) -

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

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 L ^A T _E X
	text	text práce
	thesis.pdf	text práce ve formátu PDF
	thesis.ps	text práce ve formátu PS