

fontenc



## ZADÁNÍ BAKALÁŘSKÉ PRÁCE

**Student:** Jakub Kvasnica  
**Studijní program:** Otevřená informatika (bakalářský)  
**Obor:** Informatika a počítačové vědy  
**Název tématu:** Monitorovací nástroj pro sociální síť

### Pokyny pro vypracování:

1. Seznamte se se současnými metodami analýzy sociálních sítí, těch, které mají za účel propagovat autorské dílo jejich uživatelů (Sound Cloud, flickr, DeviantArt apod.).
2. Navrhněte a implementujte webovou aplikaci pro vybranou sociální síť. Tato aplikace umožní uživateli sledovat vlastní pozici v této síti (např. sledovat svůj vliv na ostatní členy sítě apod.). Využijte API poskytnuté touto sítí.
3. Realizujte testy použitelnosti vaší aplikace. Rozsah konzultujte s vedoucím práce.

**Seznam odborné literatury:** Dodá vedoucí práce.

**Vedoucí bakalářské práce:** Ing. Adam Sporka, Ph.D.

**Platnost zadání:** do konce zimního semestru 2013/2014

L.S.

prof. Ing. Vladimír Mařík, DrSc.  
vedoucí katedry

prof. Ing. Pavel Ripka, CSc.  
děkan

V Praze dne 10. 1. 2013



## BACHELOR PROJECT ASSIGNMENT

**Student:** Jakub Kvasnica  
**Study programme:** Open Informatics  
**Specialisation:** Computer and Information Science  
**Title of Bachelor Project:** Social Network Observation Tool

### Guidelines:

1. Review the state of art of the metrics and methods for analysis of the social networks intended to facilitate the promotion of original artwork of their users, such as SoundCloud, flickr, or DeviantArt.
2. Design and implement a web-based application for a selected social network. The application will enable the user to observe the development of the user's position in this network (i.e. comparison of one's impact with other members of the network etc.). Utilize the API of that particular social network.
3. Perform usability testing of your application. Consult the extent of the assignment with your thesis supervisor.

**Bibliography/Sources:** Will be provided by the supervisor.

**Bachelor Project Supervisor:** Ing. Adam Sporka, Ph.D.

**Valid until:** the end of the winter semester of academic year 2013/2014

L.S.

prof. Ing. Vladimír Mařík, DrSc.  
**Head of Department**

prof. Ing. Pavel Ripka, CSc.  
**Dean**

Prague, January 10, 2013





**CZECH TECHNICAL UNIVERSITY IN PRAGUE**

**Faculty of Electrical Engineering**

**Department of Cybernetics**

---

## **Social Network Observation Tool**

Bachelor Thesis

Study Programme: Open Informatics  
Branch of study: Computer and Information Science

Thesis advisor: **Ing. Adam Sporka, Ph.D.**

**Jakub Kvasnica**

---

**Prague 2014**





I would like to thank my supervisor Ing. Adam Sporka, Ph.D. for his pedagogical approach and valuable advice with implementation as well as for his patience and determination. I would also like to thank my family and friends for the continuous support they have given me throughout the time of making this thesis.



### **Prohlášení autora práce**

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

### **Declaration**

I hereby declare that I have completed this thesis independently and that I have listed all the literature and publications used.

In Prague on May 23, 2014

.....



### **Abstract**

The aim of this bachelor's project is to review metrics and methods for analysis of the social networks, specifically SoundCloud, and design and implement a web-based application for it. The application will allow user to observe the development of the user's position in SoundCloud network.

### **Anotace**

Úlohou této práce je seznámení se se současnými metodami analýzy sociálních sítí, konkrétně SoundCloud, a navrhnout a implementovat pro to webovou aplikaci, která umožní uživateli sledovat svoji vlastní pozici mezi ostatními uživateli.



# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Background</b>	<b>3</b>
2.1	Social networks generally . . . . .	3
2.1.1	Definition . . . . .	3
2.1.2	History . . . . .	4
2.1.3	Popularity of social networking sites . . . . .	4
2.2	More social networks . . . . .	4
2.2.1	Myspace . . . . .	4
2.2.2	Facebook . . . . .	5
2.2.3	Yahoo! 360 . . . . .	5
2.2.4	SoundCloud . . . . .	6
2.2.5	DeviantArt . . . . .	6
2.2.6	Flickr . . . . .	6
2.2.7	Youtube . . . . .	7
<b>3</b>	<b>SoundCloud</b>	<b>9</b>
3.1	Overview of SoundCloud . . . . .	9
3.1.1	Purpose . . . . .	10
3.1.2	Features . . . . .	10
3.1.3	Purpose of our APP . . . . .	10
3.1.4	API available . . . . .	11
3.2	Users position among others . . . . .	11
3.3	Spam problem . . . . .	12
3.3.1	Fake following . . . . .	12

3.3.2	Against spam . . . . .	13
<b>4</b>	<b>Design</b>	<b>15</b>
4.1	Use cases . . . . .	15
4.2	User interface . . . . .	16
4.2.1	Followers graph . . . . .	16
4.2.2	List of followers . . . . .	17
4.3	Database . . . . .	17
<b>5</b>	<b>Implementation</b>	<b>21</b>
5.1	Language . . . . .	21
5.1.1	HTML . . . . .	21
5.1.2	PHP . . . . .	21
5.1.3	SQL . . . . .	22
5.1.4	JavaScript . . . . .	22
5.2	SoundCloud API . . . . .	22
5.2.1	Registration . . . . .	22
5.2.2	Authentication . . . . .	22
5.2.3	Getting data . . . . .	24
5.2.4	Searching and pagination . . . . .	25
5.3	HighCharts . . . . .	26
5.4	Server side . . . . .	26
<b>6</b>	<b>Testing</b>	<b>27</b>
6.1	Basic web functionality . . . . .	27
6.2	Database . . . . .	27
6.3	Users . . . . .	28
6.4	Final appearance . . . . .	28
<b>7</b>	<b>Conclusion</b>	<b>31</b>
	<b>Bibliography</b>	<b>32</b>
	<b>List of Figures</b>	<b>35</b>



<b>A</b>	<b>List of abbreviations</b>	<b>37</b>
<b>B</b>	<b>Contents of the CD</b>	<b>39</b>



# Chapter 1

## Introduction

The popularity of social networking sites has increased hugely over last few years as well as the popularity of the internet. Users are looking for many kinds of services and benefits to help them connect with other people, stay in touch with friends or share their opinion with other likeminded people. [1]

On social networking sites, where people share not only their personal informations, but some works like music, artworks, photos, videos, etc. or just simply creating some form of audience, users usually want to know what is their position among other people.

This project will be engaged in SoundCloud, where users upload, record, promote and share their originally-created sounds.

Our primary goal is to find out, how users observe their position and how they track their progress among other users. Then is up to us to design and create an application to provide them tool where they can easily observe their progress and position among others over time. The secondary goal is to find out, what spam problem users must confront and how can we help them with that.



# Chapter 2

## Background

### 2.1 Social networks generally

There are quite a lot of them on the internet, that anyone who is using internet a little bit, already had come across on at least one of them. It's called "social" because of the possibility to interact with other people or friends in some way of sharing messages, pictures, videos, songs, thoughts, etc.

#### 2.1.1 Definition

Social network is: network of social interactions and personal relationships. dedicated website or other application which enables users to communicate with each other by posting information, comments, messages, images, etc.. [3]

What is unique for social networks unlike for other websites is ability to link your own profile with profiles of your friends or other peoples in form of friend list. In some of SN sites there is a feature to create group of people with same interests, thus possibility to meet new unknown people in a way, you would hardly achieve without such SN site. This is also one of the reasons, why are social networks so popular. [2]

### 2.1.2 History

According to definition of social networking sites, the first major sites that we can call social networking site launched in 1997. SixDegrees.com allowed users to create profiles, list their Friends and surf the Friends lists. Of course that each of these features existed before in some form. For instance AIM and ICQ supported list of friends or profiles existed on most dating sites, but SixDegrees.com was the first to combine these features. [4]

Popularity of social networking sites grew as well as popularity of the internet. Lots of these sites appeared and vanished since then.

### 2.1.3 Popularity of social networking sites

Popular social networks are designed in such way, that they are very much user friendly and very easy to navigate in. Most important part is the connection with your friends and the possibility to create new friends, who you wouldn't met in other circumstances. They offer a better way to connecting with new people than other internet channels like emails and forums. Last but not least, they are free to use.

## 2.2 More social networks

Currently there is more than 200<sup>1</sup> major (well-known) social networking sites [5] but not all of them are suited for purpose of this work. I'm looking for social network site I'm at least a little familiar with and that site has to be somehow involved in presenting creativity or art of its users, such as pictures, music, perhaps videos. I won't list all of them, because its not necessary and it's time consuming. There is a few of the most known and popular:

### 2.2.1 Myspace

- launched in August 2003

---

<sup>1</sup>[http://en.wikipedia.org/wiki/List\\_of\\_social\\_networking\\_websites](http://en.wikipedia.org/wiki/List_of_social_networking_websites)

- strong music and video emphasis
- most visited social networking site in the world in years 2005 - 2008, then its leadership was overtaken by Facebook
- now over 50 million users
- web API available

I don't know anyone, who is using Myspace and I'm not using it myself.

### **2.2.2 Facebook**

- founded in February 2004
- the most used social networking service since 2009
- now active 1.19 billion users
- web API available

Very popular website that I'm well familiar with but it's not focused on presenting art or creations of its users.

### **2.2.3 Yahoo! 360**

- full launch June 2005
- for sharing photos and creating blogs
- officially closed on July 2009
- This service is not active anymore.

### 2.2.4 SoundCloud

- started in August 2007
- online audio distribution platform
- enables its users to upload, record, promote and share their originally-created sounds
- In May 2010 reached a million users
- now 40 million registered users
- web API available

My experience with SoundCloud is long and I know few people creating music and presenting it there. Web API is also well build and has features, that are necessary for our application.

### 2.2.5 DeviantArt

- launched on August 2000
- platform for any artist to exhibit and discuss artworks
- now over 25 million members
- web API available

I'm member of DeviantArt more than two years now. I'm presenting there my own art. Also lots of my friends are using this website too. Sadly, API is quite poor for our purpose. It's missing some features, we need for collecting datas.

### 2.2.6 Flickr

- launched February 2004
- photo and video hosting site
- pretty good API options available



### 2.2.7 Youtube

- video-sharing website
- web API available



# Chapter 3

## SoundCloud

### 3.1 Overview of SoundCloud

The idea of SoundCloud was originally started in Stockholm, Sweden, but was established in Berlin, Germany in August 2007 by Swedish sound designer Alex Ljung and Swedish artist Eric Wahlforss.

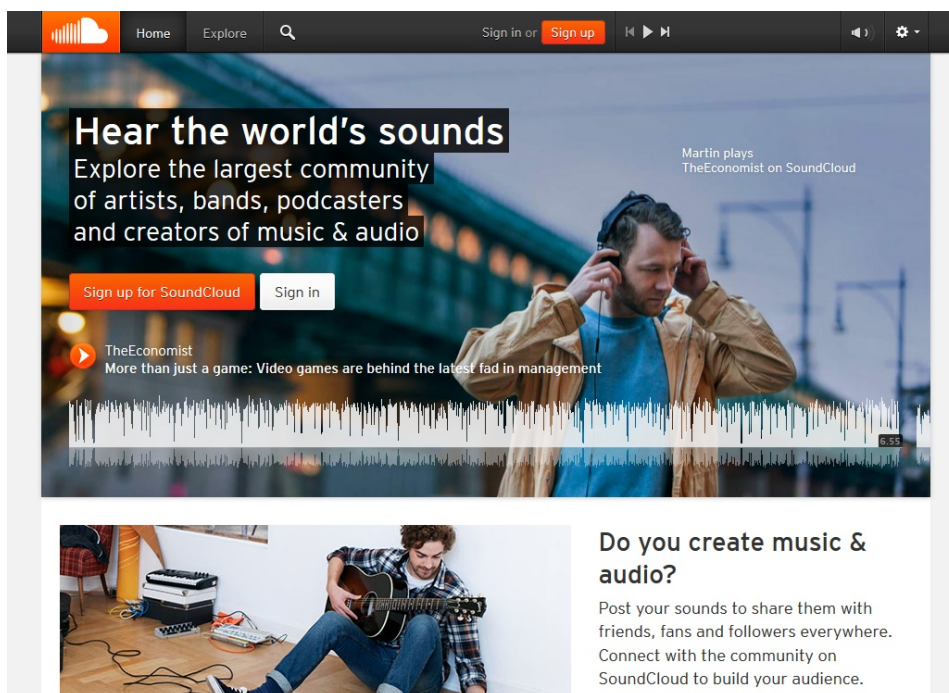


Figure 3.1: SoundCloud homepage, screenshot, [6]

### 3.1.1 Purpose

Soundcloud is an online audio distribution platform based in Germany, that enables its users to upload, record, promote and share their originally-created sounds. Intention of SoundCloud is based on allowing musicians to share recordings with each other. Later it transformed into full publishing tool with possibility to distribute users music tracks. [6]

Its a social network, where users can not only present their music, sounds, etc. but they can also get involved as an audience or ordinary users, who just enjoys music.

### 3.1.2 Features

Just like any other social networks, Soundcloud offers lot of features, to fulfill users wishes and needs and also to keep up with rest of the social networking sites. [6]

- create account with your personal information
- browse and explore all kinds of music and sounds
- comment, favourite and share music
- discover and browse other peoples user accounts
- follow other users
- upload, record, promote and share your originally-created sounds

### 3.1.3 Purpose of our APP

Primary goal of this APP is to provide tool where users can easily observe their progress and position among others over time. The secondary goal is to spam problem help them with the spam problem they must confront on SoundCloud. SoundCloud provides a good tool to collect lots of data about users and their followers, sounds, etc.. For this reason of data mining, there is API available on SoundCloud on their web page for developers.

### 3.1.4 API available

HTTP API on SoundCloud is free of charge and permitted for most purposes (within API Term of Use). It is available in five various SDKs. Python, Ruby, PHP iOS and JavaScript. Every of these SDKs are well documented on [developers.soundcloud.com](http://developers.soundcloud.com) website as well with various guides, references and examples. [8]

My choice of PHP is just personal preference, based on my previous experience with this language. It is a server-side scripting language designed for web development.

## 3.2 Users position among others

How do people find, what is their position among other users? What is important to the users is subjective. Let's consider, that getting famous in SoundCloud community is what users want to achieve. If we want to measure, how famous or successful is one at SoundCloud, we can use numbers, that are already available or can be calculated from SC profile. It is, number of:

- followers
- tracks
- plays on each track
- likes on each track

Getting more followers generally means more than having more tracks uploaded. The fact that one is being followed adds to the value of one's profile. Because the more followers you have the more people will get notification about your new uploaded track, that means more plays, more likes, comments and reposts of your song. That leads to increasing number of people, that will notice your tracks and your profile thus increasing number of followers.

Getting more tracks can also lead to increasing popularity, but if your tracks are not that good, it doesn't matter whether you are uploading more and more of them, number of your followers won't increase that much.

That and everything else of course depends on quality and popularity of your tracks, but judging quality of tracks is not subject of this work. So it's up to us to decide, what to state as most important attribute to us. In my opinion and my experience, number of followers is best way how to determine what is your position among others.

### 3.3 Spam problem

"Spam behavior" is generally understood as an activity whose purpose is to raise awareness of own product or service at the expense of convenience of the user. [7]

E-mail is a medium that is traditionally connected with spam but other media allow for spam to thrive. Over time this phenomenon has affected other kinds of Internet communication, eg. discussion forums, comments, instant messaging, etc., including the social networks. In case of SoundCloud, the spam behavior includes:

#### 3.3.1 Fake following

The purpose of following is to subscribe usually your favourite artist to monitor all his activity on SoundCloud. Mainly uploading new tracks is the most important thing, users want to know. The fact that one is being followed adds to the value of one's profile. Often, there is reciprocation of following, in short: "I follow you, and you'll follow me in return."

People sometimes abuse this in a way of following as much users as possible (thousand or more), usually for shorter period of time but long enough to get a chance of this "I follow you, and you'll follow me in return" occurring without even listening one of his songs. This way the user artificially (and fraudulently) increase number of his followers.

There is a second meaning of "fake following". The fact that having more followers is adding a value to one's account brings here a threat to artificially (and again fraudulently) increasing those numbers with fake accounts. Hundreds and thousands of fake accounts created only for this purpose. It's hard

to find such a service these days and they because it is forbidden and they are usually paid.

Other reason of this "spam fake following" is the same as the original idea of spam, advertising.

Those "fake following" accounts are often characterized by:

- act of following someone without even listening any of his sound
- very high number of following users
- high ratio following/followers
- machine generated name such like "user13579"

### **3.3.2 Against spam**

Even when SoundCloud itself is continually working to fight against spam, sometimes its impossible to catch it all. Users can be very helpful. They have the ability to "Block and Report" any account you suspect to be a fake, or spam account. This also block them from interacting the user itself. Our Application can very much help with fighting against spam on SoundCloud by helping users identify such user accounts.





# Chapter 4

## Design

There is a lot of theory around web design but I want to be this application as easy to use as possible. Simple design and maximizing clarity with fewer elements will definitely do. It will serve just for one purpose so it do not need to be complicated.

### 4.1 Use cases

We can divide using of this APP into three layers. Each of this layer will have different rights on browsing our web app. See fig. 4.1

1. visitor (unregistered user)
  - it is also user, that is not logged in through the SoundCloud.
  - can only view basic information about the app, how to login, etc..
  - as soon as the user logs in through the SoundCloud, system will automatically register him/her in database
2. user (registered user)
  - the main, most important use case
  - can view his profile, graphs and tables of subscribers
  - after logging out he becomes visitor

## 3. power user

- in addition to registered user has ability to view graphs and tables of other users
- can view all registered users

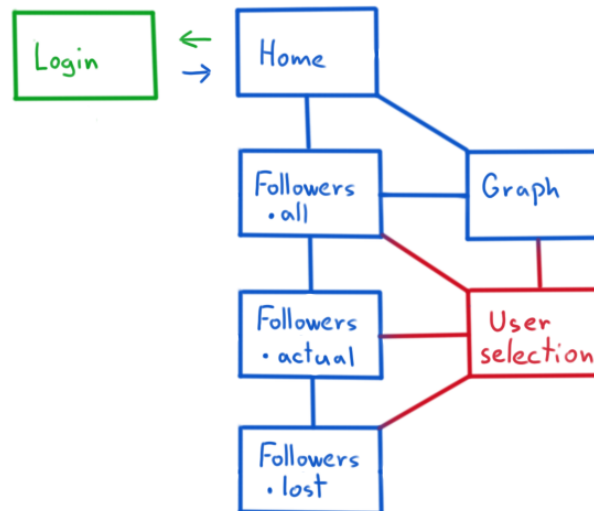


Figure 4.1: Site map (green - visitor, blue - user, red - power user)

## 4.2 User interface

This app is created for SoundCloud users, so I thought I could base my web design on old SoundCloud's one. Its really simple, header with logo at the top, menu on the left content on the right. See fig. 4.2 on page 17.

### 4.2.1 Followers graph

In a previous chapter we stated, how do people evaluate their position on the SoundCloud. I decided to take number of followers as a most important attribute. A good way of showing development of your position is to do a graph with followers as a function of time. Another good thing to show is

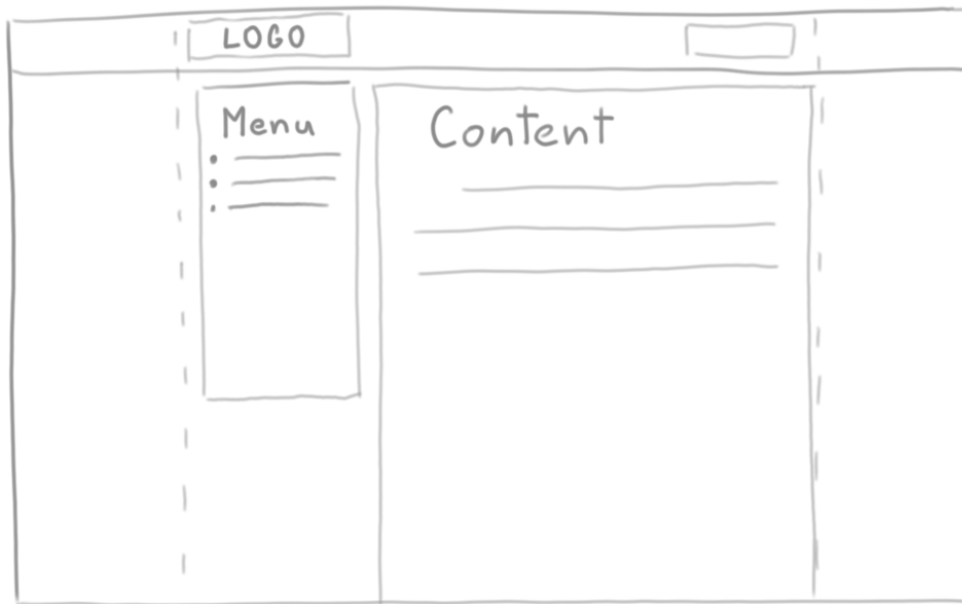


Figure 4.2: Application GUI concept, based on old SoundCloud design. [my picture]

how your follower count increased or diminished depending on a day when you uploaded a sound. See fig. 4.3 on page 18.

### 4.2.2 List of followers

I want to give user a way to observe all his actual or past followers (depends, when user joined our APP). For this purpose, we have List of followers table. It has 3 modes.

- showing all gained and lost followers together.
- showing actual followers
- showing just lost followers

## 4.3 Database

Because we want to have followers count as a function of time, we need something, where to store those numbers, because SoundCloud doesn't allow us to

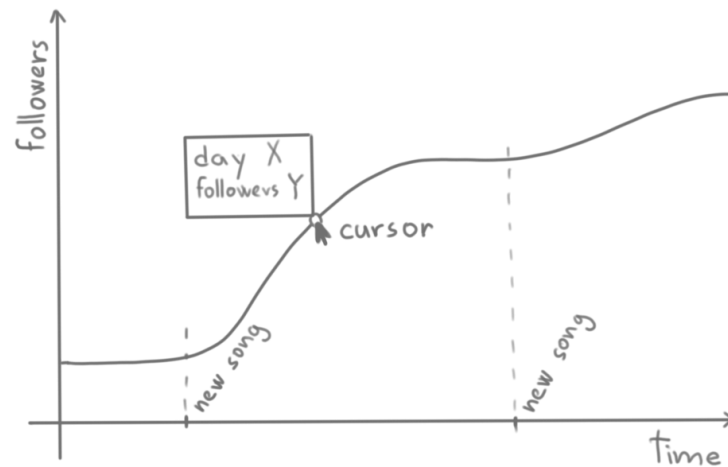


Figure 4.3: Number of followers as a function of time. [my picture, concept]

collect data from API as we desire. Only possible way of getting number of followers is actual number to this actual moment. That is why we need to collect this information every day and store it to our database. See fig. 4.4 on page 19.

Our database needs to be just that simple. Only thing, that we need to store is number of followers daily for each user [graphdata] and each follower start and end following date independently for each user. We just need to save user (follower) ID. Rest of the informations about users, songs and everything else we can get from SoundCloud API.

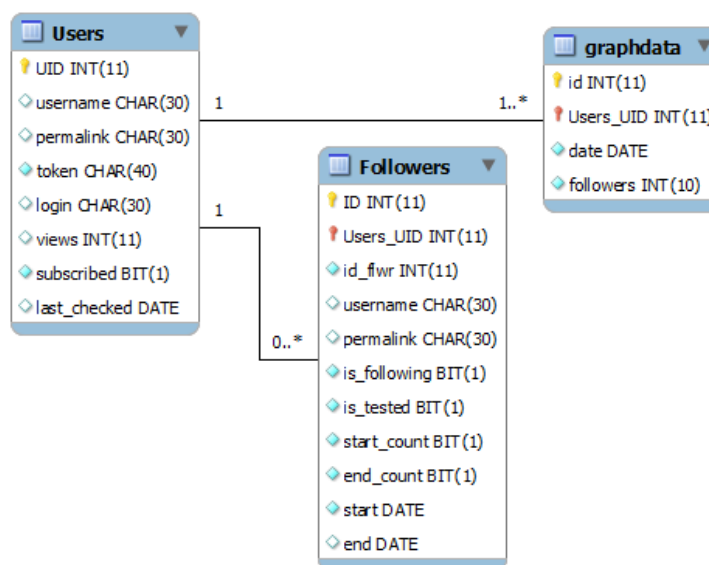


Figure 4.4: Relational database model



# Chapter 5

## Implementation

From the the assignment is clear, that we have to build web application and handle SoundCloud API. In previous chapter we also found out, that we need database for storing daily data of followers.

### 5.1 Language

In the implementation part of this thesis I used 4 programing languages. I will list them all and describe their use.

#### 5.1.1 HTML

HTML (Hyper Text Markup Language) is a language for describing web pages. Our web application is written in HTML with use of CSS (Cascading Style Sheets) to style HTML elements. It consist of 4 major pages accessible to the user. For detailed info see fig. 4.1 on page 16.

#### 5.1.2 PHP

PHP (PHP Hypertext Preprocessor) is a server scripting language, and is a powerful tool for making dynamic and interactive Web pages. This is main language for this web application. I used it for:

- creating interactive and dynamic HTML code for my web app

- handling all the logic and other important things of my app
- handling entire MySQL database communication

### 5.1.3 SQL

SQL (Structured Query Language) is a standard language for accessing databases. Every query for our MySQL database is written in SQL. The SQL code itself is handled in PHP code and executed on database server.

### 5.1.4 JavaScript

JavaScript is the programming language of the Web. In my web app serves just marginal role. I need JavaScript because of interactive graph, that I'm using. See section 5.3 HighCharts on page 26.

## 5.2 SoundCloud API

[8] As I mentioned earlier, SoundCloud API is available in five various SDKs. Python, Ruby, PHP iOS and JavaScript. Every of these SDKs are well documented on SoundCloud developers website. My choice of SDK was PHP. I will describe each individual step of use of SoundCloud API.

### 5.2.1 Registration

Our API needs to be registered on SoundCloud website to get permission to authenticate users. After filling our name and website of our APP, we get our ClientID and Client Secret to properly authenticate.

### 5.2.2 Authentication

Authentication is required for every user to get permission from a SoundCloud to access all data we need. That means that my APP will have access to their account, including any private sounds or sets that they have created or have had shared.



SoundCloud authentication uses OAuth 2.0. OAuth 2.0 allows users to authorize your application without disclosing their username and password. This is the way to authenticate our app and create authorization URL so user can log in and approve your requests. After successful authorization, user will be redirected to the redirect uri, specified while registering APP.

---

```
<?php
require_once 'Services/Soundcloud.php';

// create client object with app credentials
$soundcloud = new Services_Soundcloud('CLIENT_ID', 'CLIENT_SECRET',
    'REDIRECT_URL');

// create image link to authorization page
$authURL = $soundcloud->getAuthorizeUrl(); //login link
echo "<a href='$authURL'><img src='img/connect-sc.png'></a>";
```

---

We need now to extract the code parameter from the query string and use it to obtain an access token.

---

```
<?php
require_once 'Services/Soundcloud.php';

// create client object with app credentials
$soundcloud = new Services_Soundcloud('CLIENT_ID', 'CLIENT_SECRET',
    'REDIRECT_URL');

// exchange authorization code for access token
$code = $_GET['code'];
$access_token = $soundcloud->accessToken($code);
```

---

Now I will store this access token into session variable and into our database associated with user. With this access token, I don't need to send user through authorization cycle again, until it expires. We can use this token to access SoundCloud data through authenticated user.

---

```

<?php
require_once 'Services/Soundcloud.php';

// create client object and set access token
$soundcloud = new Services_Soundcloud('CLIENT_ID', 'CLIENT_SECRET',
    'REDIRECT_URL');
$soundcloud->setAccessToken('YOUR_ACCESS_TOKEN');

```

---

### 5.2.3 Getting data

After setting access token to our SoundCloud connection, as shown above, we have now access to the SoundCloud data. This example will print users SoundCloud username and number of followers.

```

// make an authenticated call
$current_user = json_decode($soundcloud->get('me'));
print $current_user->username;
print $current_user->followers_count;

```

---

List of all properties and subresources we can get from a specified user (when we get his ID) or from current logged user is:

```

// all properties
{
    "id":,
    "permalink":,
    "username":,
    "uri":,
    "permalink_url":,
    "avatar_url":,
    "country":,
    "full_name":,
    "city":,
    "description":,
    "discogs_name":,

```

```
"myspace_name":,
"website":,
"website_title":,
"online":,
"track_count":,
"playlist_count":,
"followers_count":,
"followings_count":,
"public_favorites_count":
}
// all subresources
{
/users/{id}  a user,
/users/{id}/tracks  list of tracks of the user,
/users/{id}/playlists  list of playlists (sets) of the user,
/users/{id}/followings  list of users who are followed by the user,
/users/{id}/followings/{id}  a user who is followed by the user,
/users/{id}/followers  list of users who are following the user,
/users/{id}/followers/{id}  user who is following the user,
/users/{id}/comments  list of comments from this user,
/users/{id}/favorites  list of tracks favorited by the user,
/users/{id}/favorites/{id}  track favorited by the user,
/users/{id}/groups  list of joined groups,
/users/{id}/web-profiles  list of web profiles
}
```

---

#### 5.2.4 Searching and pagination

With valid, unexpired access token we can search through resources. Most results from SoundCloud API are returned as a collection. The number of items in the collection returned is limited to 50 by default. With limit and offset parameters our app can page through collections. When we receive 0 items in a response, we assume that it's the end of the collection. The maximum value is 200 for limit and 8000 for offset.

This code will search all followers of user 18917562 and stores them in \$personFollowers array. I use there page size 100 (number of collections returned). Page offset will allow us to get all the followers, even if the user has more then 100 of them.

---

```
$person_id = 18917562;
$page_size = 100;
$page_offset = 0;
$personFollowers = array();
while (count($tempFollowers =
    json_decode($soundcloud->get("users/$person_id/followers", array(
        'limit' => $page_size,
        'offset' => $page_offset * $page_size)))) != 0) {
    $personFollowers = array_merge($personFollowers, $tempFollowers);
    $page_offset++;
}
```

---

### 5.3 HighCharts

Highchart is library written in JavaScript. It offers an easy way of adding interactive charts to our web APP. We used this library to visualise our followers flow as a function of time. Its free (for noncommercial use) graph JavaScript library available at <http://www.highcharts.com>. [9]

### 5.4 Server side

Website runs on a linux web server with PHP, MySQL and cron services available. Cron<sup>1</sup> is a time-based job scheduler in Unix-like computer operating systems. It allows us to run script that "feeds" our database every day without need to run it manually. I also own my own web domain with DNS record pointing to this linux web server.

---

<sup>1</sup><http://en.wikipedia.org/wiki/Cron>

# Chapter 6

## Testing

This chapter is a list of tested functionalities. If it is necessary, there is provided a method or reason of testing.

### 6.1 Basic web functionality

- Website is running and it is accessible on the internet.
- Authentication on SoundCloud is working. It's redirecting the user on main page of our app right after allowing authorization access on SoundCloud authorize page.
- Without authentication, visitors cant access anything than a login page.
- Power user can view other users tables and graphs.
- Disconnect button deauthorize user properly.

### 6.2 Database

- Connection to the database tested on a testing table in our database.
- I created a PHP script testing all queries we need for application, including creating temporary tables. All these test were made on a backup tables and test datas.

- I have tried to simulate all the possibilities, that can occur in our database to fix and prevent unnecessary or false data in our database. (i.e. following and unfollowing twice in a single day)

### 6.3 Users

- Proper function of our APP for a user with no or little Followers, large amount of followers or a user that just joined in our APP that has no data collected yet.
- Correct character encoding in APP and database because of users with username including some special characters. It rarely happened, that user with some special characters in username broke my SQL queries resulting in database corruption.

### 6.4 Final appearance

The final appearance came out perfectly. The old SoundCloud style gives it simplicity and better readability. With help of Highcharts library the graph looks amazing.

Fig. 6.1 - Index page with some basic information about user (nick, avatar) and about APP. Right after user authenticates/logs in.

Fig. 6.2 - Graph page. This is the followers flow over time (followers as a function of time). Blue squares on the date axis with letter "S" shows the moment of uploading new track. In the tooltip of this square is detailed info. On the left under Menu, you can see power user option to choose a user in our database and view his graph.

Fig. 6.3 - The table of followers with number of followers, following and tracks of the user. Those numbers turn RED when user gets suspect from "fake following". Again on the left, option only for power user to pick a different user to view.

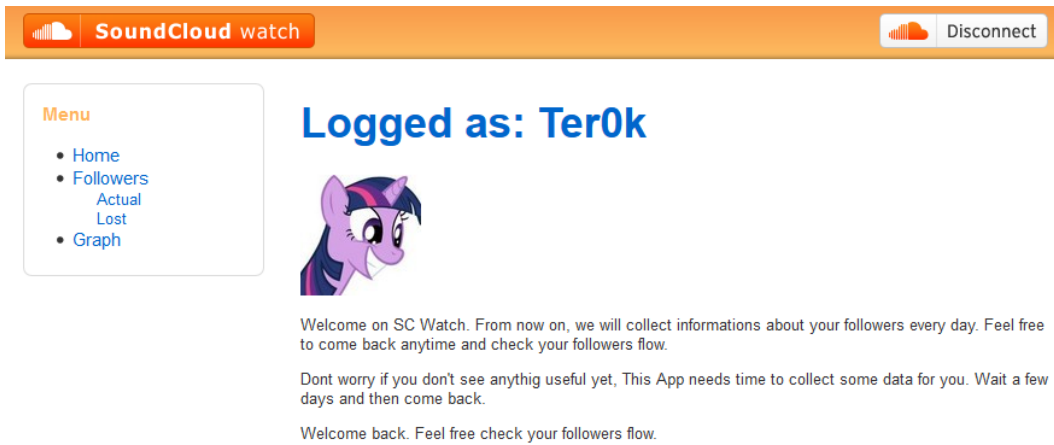


Figure 6.1: Index, Main page after logging in



Figure 6.2: Graph

SoundCloud watch Disconnect

**Menu**

- Home
- Followers
  - Actual
  - Lost
- Graph

**Users**

- Ter0k
- Darmonlor
- AdamJ
- IvoryKeysADSR
- Dbleki
- TheoryOn
- LeonD
- A.F.C.
- Tewok
- iFlare

## Followers

17

All				
Username	Followers	Following	Tracks	Date
Medi Kay	79	856	5	started: 2014-05-22
EXTRATWREXTRIAL	1002	1572	15	ended: 2014-05-22
NUGG3T (Official)	1352	500	80	ended: 2014-05-21
dj furious_77	20	464	0	started: 2014-05-21
Edm Music	1930	1495	6	started: 2014-05-21
JG\$ FBC	589	1270	5	started: 2014-05-20
Marina223s	-	-	-	ended: 2014-05-20
MurLock	-	-	-	ended: 2014-05-20
white wito	87	430	0	started: 2014-05-19
Mowmhead	1045	2000	6	ended: 2014-05-19
Hater (GSC)	110	319	10	started: 2014-05-18
Pissed Pat	389	308	27	ended: 2014-05-16
η•ω•σ• @••• pr0d•	1068	20	30	ended: 2014-05-16
Rod. Santos	38	86	7	started: 2014-05-16
Yoann Rense	335	1238	5	started: 2014-05-13
FeriziBeatz	308	14	18	ended: 2014-05-13
Filip Havlik	421	0	3	ended: 2014-05-10
Nenax 3b	261	528	0	ended: 2014-05-09
gregzorba	585	1	0	ended: 2014-05-08
di_marty	92	1	4	ended: 2014-05-06

Figure 6.3: Followers table



# Chapter 7

## Conclusion

The aim of this study was to review the metrics and methods for analysis of the social networks. Even when there is a lots of studies on this subject I had hard time to find something that will be a little useful for my selected social network. My experiences with soundcloud and other social networks were important for finding what I needed.

Our primary goal has been successfully completed. We defined the value of users account with number of followers and created a tool, which help users to observe this, with time development.

Due to limitations of SoundCloud API, the secondary goal was finished only partially. Because there is no way to check via API, whether user, who followed someone, played at least one of his tracks or not. There remains only one option, that we guess fake followings by number of followers, following and tracks. This needs fine tuning the rules of classifying the "bad ones" to minimize mistakes of guessing.

I chose PHP for programming application and implementing API to it. Web page itself is formed with HTML and CSS. Data are stored in MySQL database. Everything runs on linux-based web server with all needed services, including Cron.

Application has been thoroughly tested throughout all the programming by the author. Thanks to that all major bugs deficiencies were detected and corrected on time, causing only few problems. After finished programing phase,

a complete test of functionality was launched.

Author plans to continue on this project, because there is no other application for SoundCloud, that solves this kind of problem. In a nearest future I plan to add a better way to distinguish a fake follower from normal users. The main goal is also listen to the users feedback and requests.

# Bibliography

- [1] 6 Reasons Why Social Networking is So Popular These Days. *web*. <http://www.brandignity.com/2012/11/6-reasons-why-social-networking-is-so-popular-these-days/>
- [2] What is a social networking site? *web*. <http://computer.howstuffworks.com/internet/social-networking/information/social-networking.htm>
- [3] Definition of social network. *Oxford Dictionary*. <http://www.oxforddictionaries.com/definition/english/social-network?q=social+network>
- [4] Nicole B. Ellison and danah m. boyd: *Michigan State University*. Social Network Sites: Definition, History and Scholarship. [http://www.postgradolinguistica.ucv.cl/dev/documentos/90,889,Social\\_network\\_boyd\\_2007.pdf](http://www.postgradolinguistica.ucv.cl/dev/documentos/90,889,Social_network_boyd_2007.pdf)
- [5] The Brief History of Social Media. *uncp.edu*. <http://www2.uncp.edu/home/acurtis/NewMedia/SocialMedia/SocialMediaHistory.html>
- [6] About SoundCloud. *SoundCloud website*. <https://soundcloud.com/pages/contact>
- [7] Spamming. *Wikipedia*. <http://en.wikipedia.org/wiki/Spamming>
- [8] SoundCloud for Developers. *SoundCloud*. <https://developers.soundcloud.com/>

- [9] Highcharts, Highstock. *charting library*.  
<http://www.highcharts.com/products/highcharts>

# List of Figures

3.1	SoundCloud homepage, screenshot, [6]	9
4.1	Site map (green - visitor, blue - user, red - power user)	16
4.2	Application GUI concept, based on old SoundCloud design. [my picture]	17
4.3	Number of followers as a function of time. [my picture, concept]	18
4.4	Relational database model	19
6.1	Index, Main page after logging in	29
6.2	Graph	29
6.3	Followers table	30



# Appendix A

## List of abbreviations

**API** - Application programming interface

**CSS** - Cascading style sheets

**GUI** - Graphical user interface

**HTML** - Hyper text markup language

**ID** - Identification

**PHP** - PHP hypertext preprocessor

**SC** - SoundCloud

**SDK** - Software development kit

**SN** - Social network

**SQL** - Structured query language





# Appendix B

## Contents of the CD

dir / pdf – contains the pdf file

dir / source – contains the source codes

dir / latex – contains latex version of a document

dir / netbeans – contains netbeans project with source codes