

Virginia Tech

TrackPunch

Music Discovery Redefined via Social Influence Models

Jonas Weigert

CS 4624: Multimedia, Hypertext, and Information Access

Dr. Fox

Thursday, May 8th 2014

Table of Contents

1. Executive Summary	p. 3
2. User Manual	p. 5
a. Getting Started	
b. Playing Music	
c. My Music	
d. Discover Music	
e. My Friends	
f. Search	
3. Developer Manual	p. 6
a. Getting Started	
b. Development Environments	
c. Important Notes	
4. Timeline	p. 8
5. Attachment #1: Player Screenshots	p. 11
6. Attachment #2: Crawler Monitoring Screenshots	p. 17
7. Attachment #3: Data Acquisition Charts	p. 18

Executive Summary

TrackPunch is a music streaming service that turns music discovery into a social experience. Our platform provides users with an elegant way to find new music based on their friends' listening habits. We aggregate music data from top music sources such as Spotify, Rdio, and Pandora to find the person that influences each user individually. Our users can search, listen to, and share high quality content from YouTube, GrooveShark, and SoundCloud for free. On TrackPunch, your next favorite song awaits.

Since I am the technical co-founder of this early stage social platform, I am in charge of the entire development process. This includes determining the technical specifications for the features requested by our beta users. These requirements then need to be implemented into the existing architecture, while paying close attention to maintaining the scalability of the entire system.

To accomplish this goal over the course of the semester we decided to focus on the languages I know well. This resulted in the entire front-end to be developed using PHP, HTML5, CSS3, and JavaScript. To reduce the production time, I decided to use several frameworks for the different programming paradigms. **Smarty** was used to enforce the MVC model within PHP. Smarty allows the PHP file accessed by a user to run some basic operations based on the Model (pure PHP classes and libraries) to then render the View that is described in the form of a *.tpl template file. These files are compiled by Smarty for execution efficiency. **Bootstrap** is a website user interface framework that was originally developed by Twitter to quickly expand their offerings and has since become its own open source entity. I used Bootstrap's powerful HTML structure, CSS, and JavaScript to generate the functionality while at the same time writing a large amount of custom CSS to make the experience truly unique. **The SoundManager API and the YouTube Player API** were used to enable the playback. These are powerful, yet very different JavaScript libraries that we wrapped in a single player.js file that allows TrackPunch to easily play any track on our platform.

There were many more libraries provided by the different service providers that were used to enable only the player's functionality. For the player alone, I wrote over 10,000 lines of custom code taking advantage of the aforementioned libraries to create a product totaling over 133,000 lines of code in the various languages.

The next challenge I faced was to efficiently import our user's social data so that the experience remained exceptionally fast while providing access to a large amount of information we knew were available to our users. At first, I decided that a data collection service, built in Google's high performance language GO would provide the best results. However, the challenges with running GO outside of Google's AppEngine proved to cause too many reliability issues. I then redesigned

the data collection to be powered by PHP like the rest of the program while exceeding the linear performance limits set forth by the language. I was able to create simultaneous workers that executed as independent processes on the server directly, not on the Apache VM. This was accomplished using priority job queuing through the highly flexible beanstalkd job-worker model. The data collection system is currently able of running 51 simultaneous workers the result in a 200 rpm (web requests per minute) throughput rate without exceeding the rate limits set forth by the social APIs. All of this processing is done on a dual core VPS with 1.0GB RAM. Since the rate limits are never exceeded through the priority queue design, this throughput rate remains possible even at high traffic times of over simultaneous 250 users. Another advantage to using beanstalkd is that, if necessary the workers can be rewritten in more efficient languages that allow us to improve the throughput rates even further. To ensure reliability of this system I also enabled monitoring by the powerful cloud platform NewRelic.com. If error occur at a rate of 5% or more, or the throughput rates drop below 100rpm, I get notified and can resolve any issue before it causes a complete collapse in the data acquisition process.

The success of the semester project was always contingent on a successful launch during the course of the semester. After I had implemented, tested, and created sufficient monitoring tools to be able to ensure product reliability the website went live on April 29th 2014 acquiring over 200 users in the first day and indexing over 659,000 individual actionable data points in only 36 hours. That is a rate of over 5 data points per second. This is even more impressive considering the fact that by hour 36 we only had access to 83 users information. This means that for every user that signs up our platform is able to find 7,939 tracks that their friends have listened to.

After the successful launch of the platform the team has evaluated the next steps that are overall performance improvement in the way the social data is served to the users which will be achieved by implementing a Redis caching server, mobile device enabled playback, and more discovery features.

Overall the project was a success since my work was able to launch a maintainable platform during the course of the semester. There is much more work to be done over the coming months to utilize this ever growing data set that allows our users to find their next favorite song every day in new ways. Concrete plans are to build a mobile version of our player to enjoy tracks on the go as well as being able to explore the discography of any artist a user is interested in.

User Manual

Getting Started

To start using TrackPunch you can sign up with just an email address and a password. This allows you to explore global music trends and search for one of over 40 million tracks. TrackPunch is a social experience and is even better when you link your existing Facebook account. This allows us to find you the latest and greatest content even faster.

Playing Music

Anywhere on TrackPunch where you see a play icon, just go for it and click it. It may be your next summer anthem. It is free to listen to as many songs as you want and there are no interruptions from ads.

My Music

Here you will find all the content that relates to you: your music listens, library and playlists. If you just want to listen to your favorite workout playlist on the way to the gym or get your party started off right you can access your music right here.

Discover Music

Right as you land on first screen of the player you will see the global “public” stream of music. These are all the tracks listened to by our users and their friends. But don’t worry; we make sure that anyone’s listening habits are between them and their friends. If you signed up with Facebook or want to link your existing account now that you see how awesome the platform is, head over to the “Friends Stream” and take a look at what is going to be the hottest song at the next party, during the next road trip, or the latest workout jam.

My Friends

This section allows you to see a particular friend’s music. You will see all their recent music listens from any platform or, if they are a TrackPunch user, you can see their entire library and playlists and start adding some of that content to your collection. This is perfect to find out what music your crush is currently into.

Search

Searching music is still an option even with our amazing discovery at your fingertips. Feel free to take the unbeaten path and find music no one has ever heard of. You can find the throwbacks, latest remixes, even music videos.

Developer Manual

Getting Started

TrackPunch is designed to run in the browser. No downloading any software or content just sign up and enjoy. To achieve this goal there are three major components to the TrackPunch experience.

1. **TrackPunch // Player**
The Player allows our users to interact with a variety of content types.
2. **TrackPunch // Card API**
The Card API compiles the data from the crawler into a targeted stream.
3. **TrackPunch // Crawler**
The crawler indexes your friend's data so we can serve it to our users.

Programming Environments

All of the components of the TrackPunch architecture are built using well-established web technologies that require no special environments. You can see a detailed breakdown below:

// Player Features	Software / Language / Deails
Server	Rackspace VPS & Apache
Database	Rackspace CloudDB
Layout & UI	HTML5 & CSS3
Dynamic Content	JavaScript & jQuery
Backend	PHP
Template Engine	Smarty
Audio Playback	SoundManager.js, YouTube Player API
Content APIs	Facebook Open Graph GrooveShark Data & Stream APIs SoundCloud Data & Stream APIs YouTube Data & Stream APIs Last.fm API

// Card API Features	Software / Language / Details
Server	Rackspace VPS & Apache
Database	Rackspace CloudDB
Backend	PHP
Restful / JSON API Framework	Slim (PHP)

// Crawler Features	Software / Language / Details
Server	Rackspace VPS & Apache
Database	Rackspace CloudDB
Backend	PHP
Crawler Job Queue	Beanstalkd
Crawler Job Workers	PHP & Pheanstalk (Beanstalk for PHP)
APIs	Facebook Open Graph

Important Notes

Facebook Open Graph

Facebook has introduced complex and severe rate limits into their Open Graph API to ensure that developers and third parties in general do not abuse their API access privileges. This was a major challenge for the TrackPunch Crawler as the goal is to index as much data as possible to serve to the user without hitting the rate limits. The beanstalkd-based integration of worker cues allows us to easily regulate the access rates by starting or killing workers on the fly. The performance of the entire crawler system is monitored using NewRelic.com. The monitoring includes CPU usage, memory usage, database transaction times, error tracking. NewRelic in combination with our beanstalkd monitoring console has also allowed us to eliminate several choke points in the data acquisition process.

Timeline

Here is the detailed timeline of the progress during the semester from the beginning to launch on April 29th 2014.

Week 1: January 19th – January 25th

- **General Project Reassessment**
(Development started early December 2013)
- **Goals & Guidelines for the Coming Months**
 - Focus on social music discovery
 - Additional Streaming Source (SoundCloud)

Week 2: January 26th – February 1st

- Basic UI Upgrades
- Playback JavaScript Integration
 - Playback now possible with lists of tracks
- Examining Scalability Issues with existing Database Structure
 - Discussed concerns with Professor B. Aditya Prakash

Week 3: February 2nd – February 8th

- Restructured Database for Scalability
- Adapted existing backend PHP to match new schema
- Added basic support for Playlists & Finding Track Definitions

Week 4: February 9th – February 15th

- Added Grooveshark as Streaming Source
 - Searching Grooveshark Content
 - Playback Integration
- Beginning of User Interaction tracking within TrackPunch

Week 5: February 16th – February 22nd

- Refined adding Tracks to Playlists (see screenshots)
- Added Playlist privacy support & UI components

Week 6: February 23rd – March 1st

- UI Cleanups
- Beginnings of Card UI designs for Public & Private Streams

Week 7: March 2nd – March 8th

- Shared Player UI & Playback
 - Users are now able to send shared links to anyone via Facebook Message & Post the Shared Player to Facebook Wall / Group
 - Embedding Support to integrate player into the websites / blogs

Week 8: March 9th – March 15th

- First set of social Data
- First Social Data Scalability Issues
- Created Dedicated Server for Social Data Integration

Week 9: March 16th – March 22nd

- General Codebase Cleanup
- Evaluating Pre-Launch Issues with the Player
- Evaluating Crawler Requirement to select appropriate language
 - Selected GO
 - Became major issue down the road

Week 10: March 23rd – March 29th

- Card UI Redesign to current “dark” theme
- Added graphic heavy UI with massive album artwork
- Simplified the buttons to create sleeker look

Week 11: March 30th – April 5th

- Testing & Debugging of the Player UI components pre-launch
- First GO based crawler unusable for production due to scalability issues
 - Facebook rate limits exceeded constantly
 - Unpredictable crashes (concurrency issues?)

Week 12: April 6th – 12th

- Instant playback of content from other platforms
- Single click find the same song on our streaming platforms by providing only a title and artist to the search function

Week 13: April 13th – April 19th

- Ensured event tracking functional within TrackPunch Player
- Improved Database Schema performance by adding necessary indexes

Week 14: April 20th – April 26th

- Rewrote Crawler in PHP using Beanstalkd
- Wrote Card API to power “Public” and “Friends” Stream
- Integrated Crawler and Card API into the TrackPunch Player

Week 15: April 27th – May 3rd

- Preparing for launch (testing)
- Launched TrackPunch.com on April 29th.

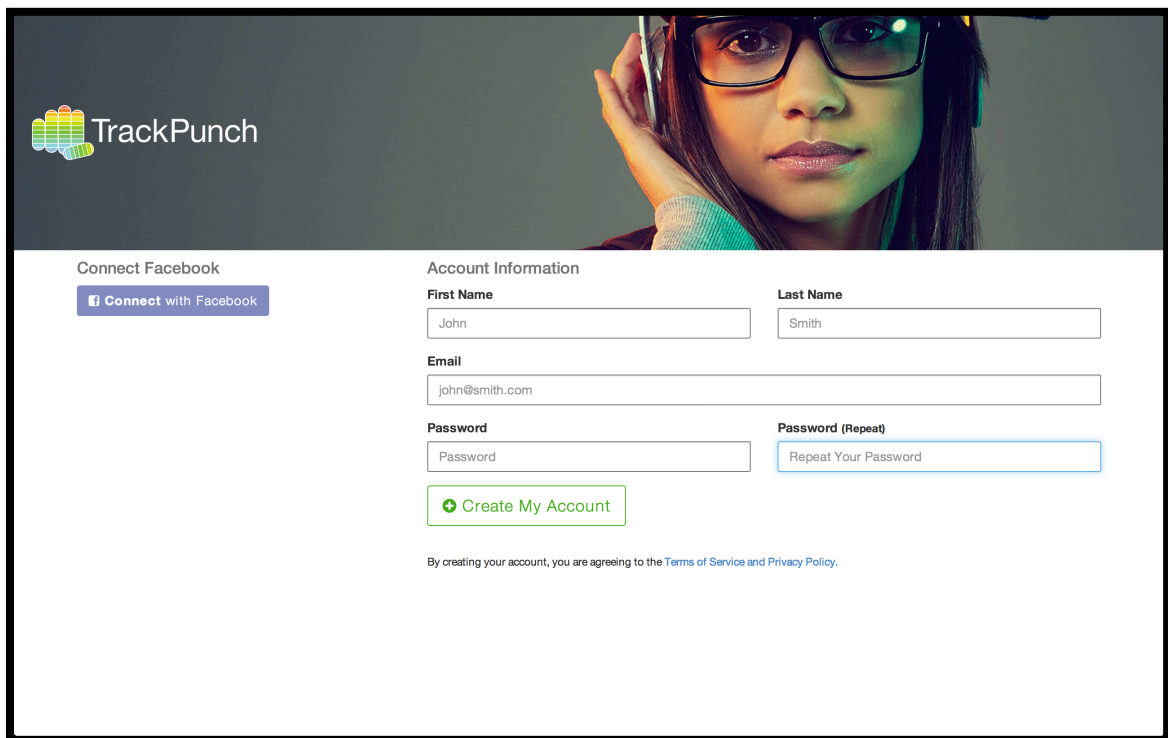
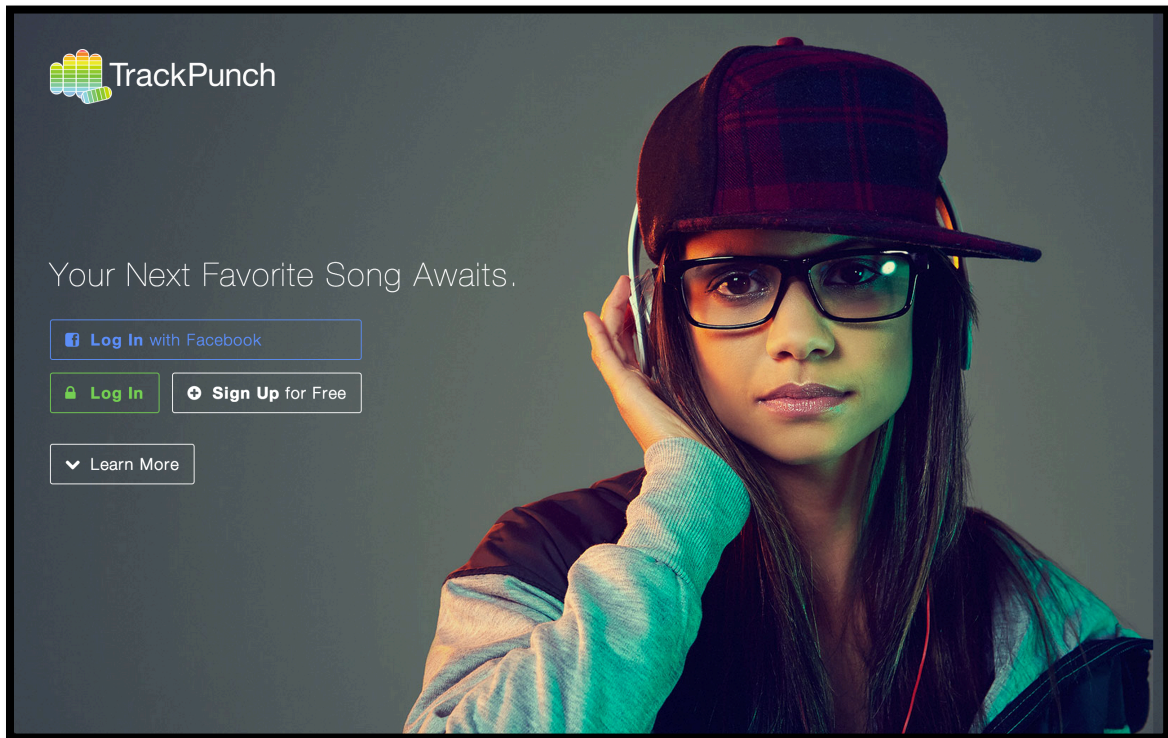
Acknowledgments:

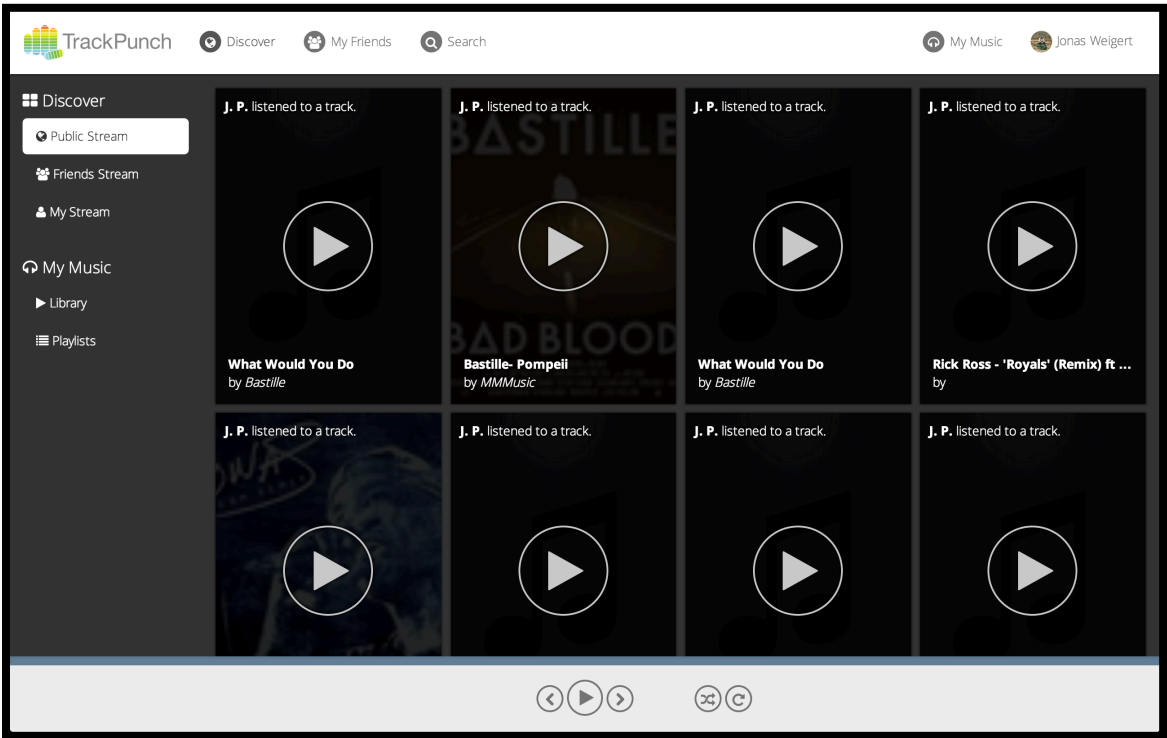
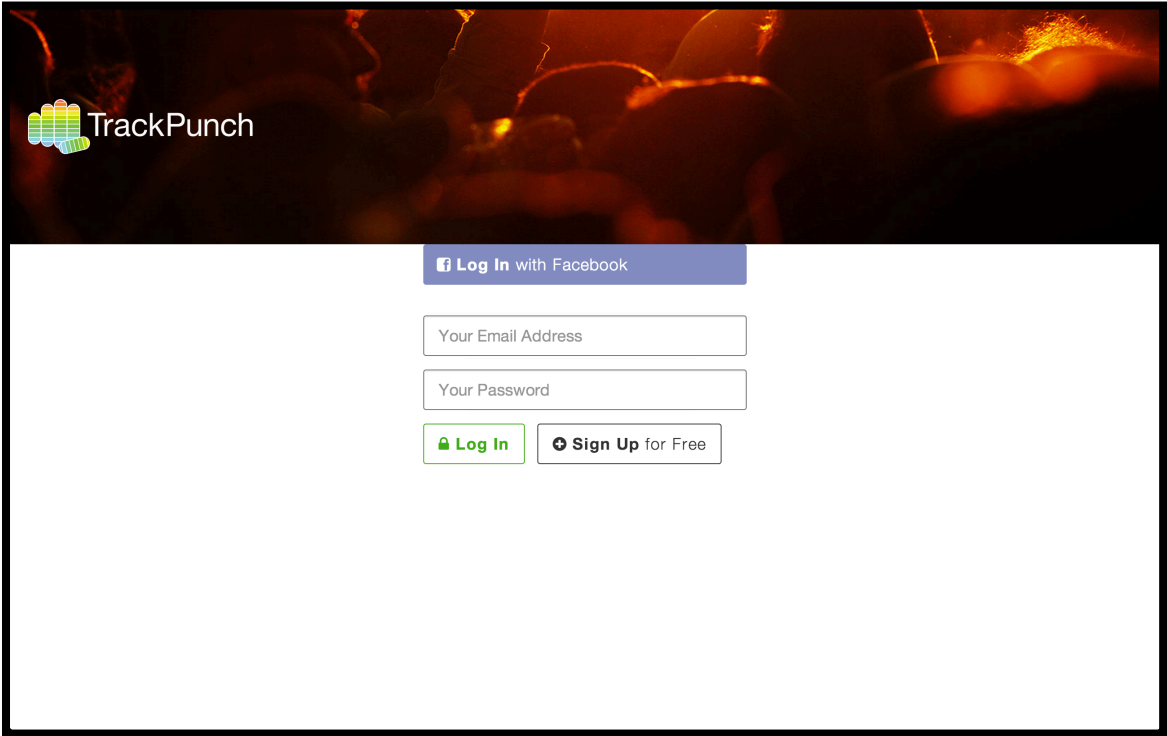
A special thanks goes to Dr. Fox for allowing me to pursue a project I had been working on outside of class previously and bring it to completion in a semester.

Sources:

The Nielsen Company, "Music Discovery still dominated by radio, says Nielsen Music 360 Report," *The Nielsen Company*, 8 Aug. 2012; <http://www.nielsen.com/us/en/press-room/2012/music-discovery-still-dominated-by-radio--says-nielsen-music-360.html>.

Attachment #1: Screenshots of the Player





TrackPunch Discover My Friends Search My Music Jonas Weigert

Discover Public Stream Friends Stream My Stream My Music Library Playlists

Filter & Find Tracks, Artists, Albums, and Genres

Title	Artist	Album	Genre
Listen	(Hed) P.E.	Single	Unknown Genre
God Must Be A Lonely Man	Ashworth	Unknown Album	Unknown Genre
Treasure (feat. Bruno Mars)	Audien	Unknown Album	Unknown Genre
Space Jam	Audio Push Feat. Lil Wayne	Unknown Album	Unknown Genre
Still in This Bitch (feat. Juicy J & T.I.)	B.o.B	Fuck Em We Ball	Unknown Genre
Strange Clouds (feat. Lil Wayne)	B.o.B	Strange Clouds	Unknown Genre
The Night Is Young (ft. Cherub)	Big Gigantic	The Night Is Young	Unknown Genre
Ain't No Sunshine (Lido Remix)	Bill Withers	Lido	Unknown Genre
23 feat. Miley Cyrus, Wiz Khalifa, Juicy J)	Caked Up	Unknown Album	Unknown Genre
Applause (feat. Lady Gaga)	Caked Up	Unknown Album	Unknown Genre
Dark Horse	Caked Up	Unknown Album	Unknown Genre
Gas Pedal (feat. Sage the Gemini)	Caked Up	Unknown Album	Unknown Genre
Royals (feat. Lorde)	Caked Up	Unknown Album	Unknown Genre
Wrecking Ball (feat. Miley Cyrus)	Caked Up	Unknown Album	Unknown Genre

Navigation icons: Previous, Play/Pause, Next, Repeat, Shuffle

TrackPunch Discover My Friends Search My Music Jonas Weigert

Discover Public Stream Friends Stream My Stream My Music Library Playlists

13 Tracks

End of a Long Day
Kick Back, and Relax

10 Tracks

Westy Roatrip
One Mile at a Time Until It Breaks ...

Navigation icons: Previous, Play/Pause, Next, Repeat, Shuffle

TrackPunch Jonas Weigert

My Library My Friends **Coming Soon** Filter for Tracks...

Listen (Hed) P.E. (Single)	God Must Be A Lonely Man Ashworth ()	Treasure (feat. Bruno Mars) Audien ()
Space Jam Audio Push Feat. Lil Wayne ()	Still in This Bitch (feat. Juicy J & T.I.) B.o.B (Fuck Em We Ball)	Strange Clouds (feat. Lil Wayne) B.o.B (Strange Clouds)
The Night Is Young (ft. Cherub) Big Gigantic (The Night Is Young)	Ain't No Sunshine (Lido Remix) Bill Withers (Lido)	23 feat. Miley Cyrus, Wiz Khalifa, J... Caked Up ()
Applause (feat. Lady Gaga) Caked Up ()	Dark Horse Caked Up ()	Gas Pedal (feat. Sage the Gemini) Caked Up ()
Royals (feat. Lorde) Caked Up ()	Wrecking Ball (feat. Miley Cyrus) Caked Up ()	Thinking About You (feat. Ayah ...) Calvin Harris (18 Months)
5th Element Classified ()	Rather Be (Cash Cash X Valley Re... Clean Bandit ()	Turn Down For What (feat. Lil John) DJ Snake ()
Tennis Court (feat. Lorde) Flume ()	Remember the Name (feat. Style... Fort Minor (The Rising Tied)	Give It All We Got Tonight George Strait (Give It All We Got Toni...
Hyperparadise (GANZ Flip) Hermitude x Flume ()	End of the Story Hucci (Rose Gold)	Fancy (Yellow Claw Remix) Iggy Azalea ()

9 Riders on the Storm The Doors Feat Shoop Dogg Need for Speed Underground 2 OST Unknown Genre

10 All Of The Lights (feat. Kanye West) Pretty Lights Unknown Album Unknown Genre

11 Lost and Found (ODESZA Remix) - The Hidde... Pretty Lights Unknown Album Electronic

12 Rest of My Life SOJA Unknown Album Unknown Genre

https://www.trackpunch.com/player/#

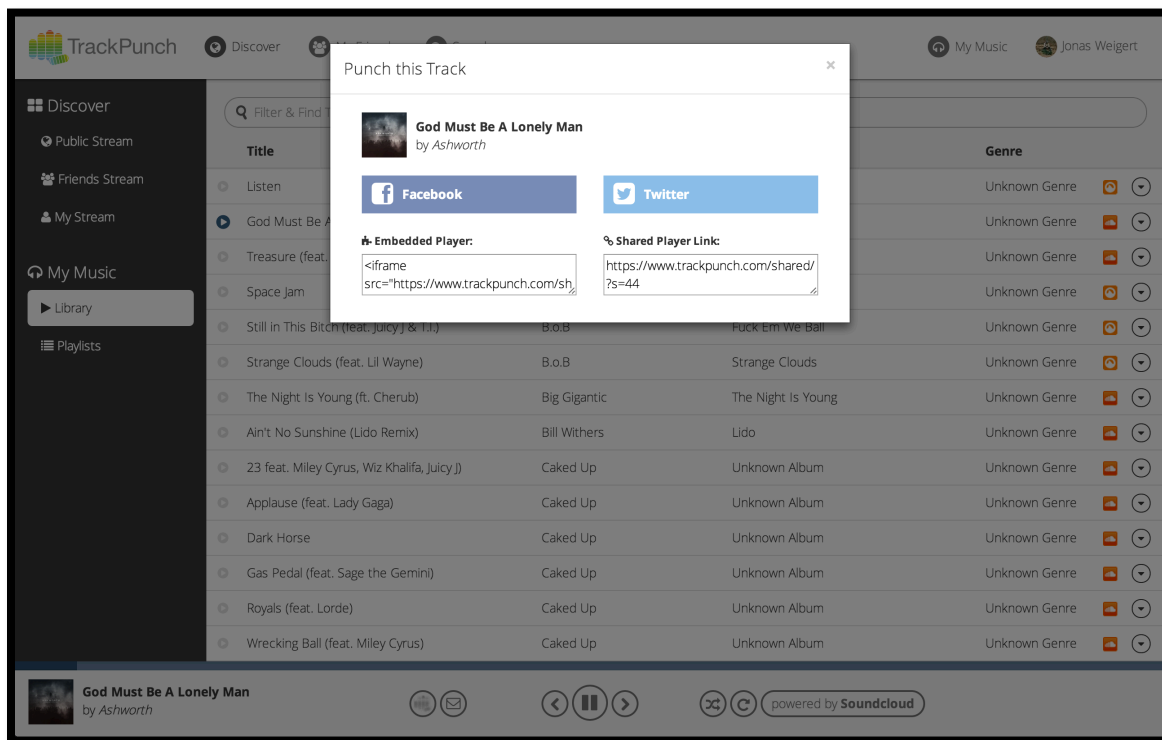
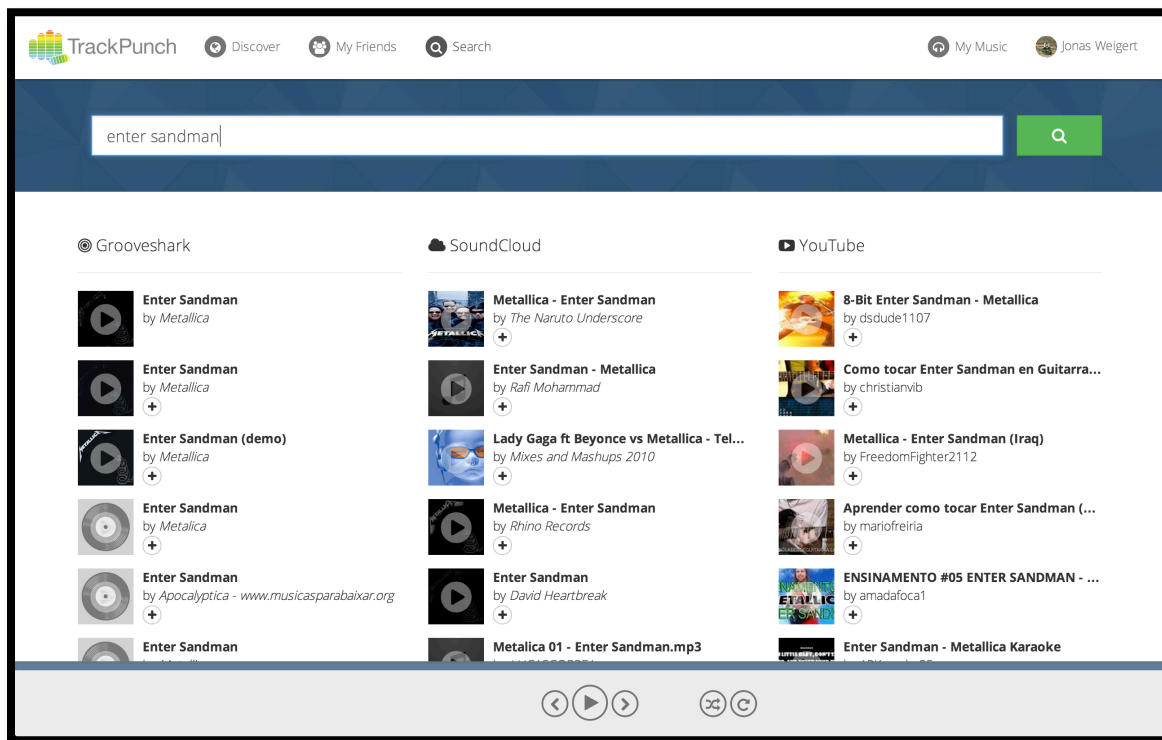
TrackPunch Discover My Friends Search My Music Jonas Weigert

My Friends

Explore Your Friends' Music from TrackPunch, Spotify, Beats Music, Pandora...

Find a Friend...

Aj Vizcarrondo	Alex Rhea	Alyssa Bane	Anthony Sosso
Ashu Desai	Banner White	Baylee Boo Hodson	Becca Hosey
Becky Stanski	Ben Adrian	Ben Maddock	Ben Wang
Ben Weinstein-Raun	Blair Gallacher	Božena-Franziska Woi...	Braden Croy
Brandon Carroll	Brian Clark	Carl-Auguste Sommer	Carly Sokach



TrackPunch Discover My Friends Search My Music Jonas Weigert

Filter & Find Tracks, Artists, Albums, and Genres

Title	Artist	Album	Genre
Listen	(Hed) P.E.	Single	Unknown Genre
God Must Be A Lonely Man	Ashworth	Unknown Album	Unknown Genre
Treasure (feat. ...)			Unknown Genre
Space Jam			Unknown Genre
Still in This Bit			Unknown Genre
Strange Clouds			Unknown Genre
The Night Is Yo			Unknown Genre
Ain't No Suns			Unknown Genre
23 feat. Miley C			Unknown Genre
Applause (feat			Unknown Genre
Dark Horse			Unknown Genre
Gas Pedal (feat. Sage the Gemini)	Caked Up	Unknown Album	Unknown Genre
Royals (feat. Lorde)	Caked Up	Unknown Album	Unknown Genre
Wrecking Ball (feat. Miley Cyrus)	Caked Up	Unknown Album	Unknown Genre

Send a Message

To:

Message:

God Must Be A Lonely Man by Ashworth
www.trackpunch.com
 TrackPunch lets you listen & share music with your friends. A Youtube music player that brings social listening to new heights.. For Free!

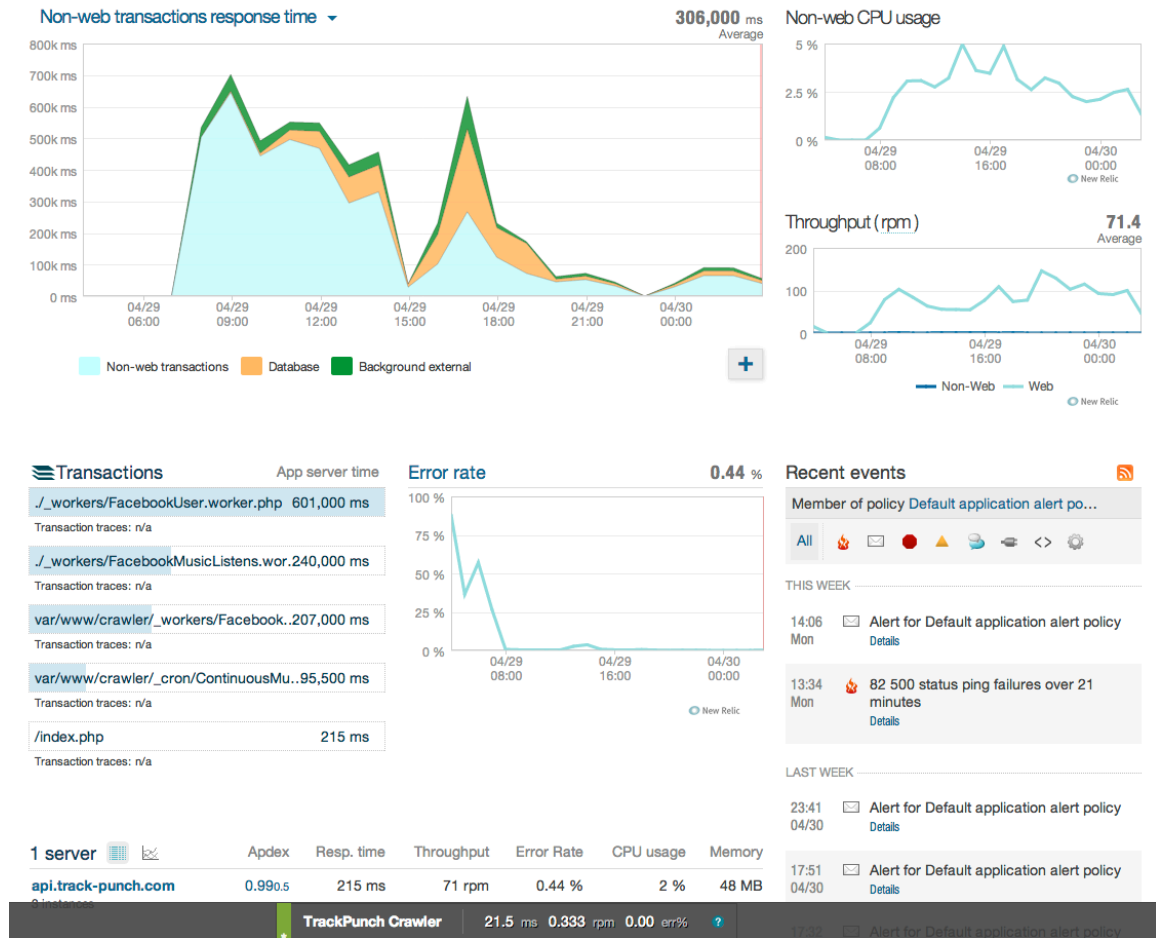
God Must Be A Lonely Man by Ashworth

powered by Soundcloud

TrackPunch

God Must Be A Lonely Man
 Ashworth
 Unknown album

Attachment #2: Screenshot of Crawler Monitoring



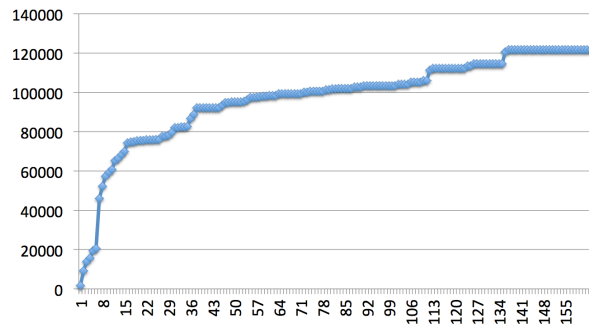
Beanstalk console localhost:11300 All tubes

name	current-jobs-urgent	current-jobs-ready	current-jobs-reserved	current-jobs-delayed	current-jobs-buried	total-jobs
FacebookMusicListens	0	0	0	0	0	242712
FacebookSong	0	0	0	0	0	201928
FacebookToken	0	0	0	0	0	167
FacebookUser	0	0	0	0	0	638786
default	0	0	0	0	0	0

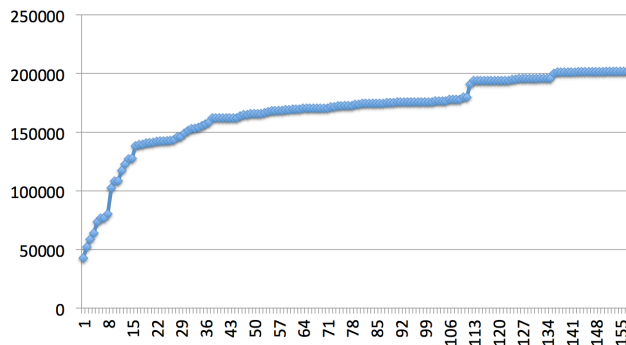
Attachment #3: Data Acquisition Charts

The crawler had to be run at reduced capacity starting at hour 40 due to the large number of database writes becoming a performance issue. *We are currently resolving the issue by batching writes & moving actions from SQL to NOSQL.*

Users Reached & Indexed



Individual Tracks Indexed



Music Listens Actions Indexed

