



Tarea Programada 1

Elizabeth Córdoba

Marianela Cordero

21 de noviembre del 2013

Tabla de Contenidos

Descripción del problema.....	02
Diseño del programa.....	03
Librerías usadas.....	05
Análisis de resultados.....	06
Manual de usuarios.....	07
Conclusion personal.....	13
Bibliografía.....	14

Descripción del problema

La cuarta tarea programada tiene como objetivo crear una aplicación que involucre la información de las bandas existentes en Bandcamp y la divulgación de las mismas por medio de la red social Twitter.

La página Bandcamp no cuenta con el API disponible, por lo que se debe implementar una solución para adquirir la información solicitada.

El usuario puede efectuar búsquedas de las bandas ya sea por el género musical o por la localización de la banda.

En la aplicación diseñada se deben mostrar los primeros diez resultados de las búsquedas realizadas, cada una de las bandas debe contener su nombre, el álbum musical y el Url que lo redireccione a la página web de dicha banda en Bandcamp.

Por otra parte el usuario puede realizar tweeds sobre sus bandas preferidas, debe de incorporar la información de la banda (nombre, álbum, Url). Para este punto se debe de trabajar con el API de Twitter.

Diseño del programa

Se realizó la clase inicializar, para iniciar las variables que almacenaran el URL, el nombre, el álbum y el URL de la imagen de la banda.

La clase buscar se compone de tres métodos. El método de búsqueda es el que va recibir el criterio de búsqueda que ingrese el usuario en la aplicación, el cual se une a <http://www.bandcamp.com/tag/> para formar el url correcto.

Con ayuda de hpricot y open-uri se obtiene el html del url anterior, mediante métodos de búsqueda de hpricot se extrae la banda, el álbum y el url en tres arreglos distintos, los cuales más adelante se recorren y se almacenan las diez primeras en una lista con sus concordancias específicas, este método retorna la lista.

El método obtener datos recibe la lista anterior, la cual es recorrida y con hpricot se obtiene el html de cada banda, para obtener el url de la imagen, para la publicación en la página de resultados.

Los datos de la lista y sus respectivas imágenes se almacenan en una variable llamada datos, la cual llama la clase inicializar, para iniciar esos valores, esta variable es almacenada en la variable global \$información. Este método llama al método pantallas_resultados.

El método anterior recibe la variable global información, es aquí donde se guardan en variables distintas cada información de cada banda, estas variables son globales, las cuales son incluidas en el HTML para mostrarlas en la pantalla resultado.

Para la creación de la aplicación se hicieron dos páginas, la principal en donde el usuario indica el criterio de búsqueda y esta redirecciona a una siguiente, con las diez primeras bandas que corresponden a dicho criterio.

Para realizar tweets se incluye un botón en cada uno de los resultados, el cual se conecta al API de twitter para compartir el tweet.

Librerías Utilizadas

- **Sinatra:** Es una gem de ruby, la cual se define por ser un Lenguaje Específico de Dominio, el cual permite crear aplicaciones Web en Ruby.
- **Hpricot:** Es una gem de ruby que permite extraer y analizar información de páginas de HTML, trabaja junto con open-uri.
- **Open-uri:** Por medio de esta gem se puede abrir (accesar) una Url HTTP.
- **Rubygems:** utilizado en el lenguaje de programación Ruby, con el objetivo de brindar un formato estándar; teniendo como ventaja el poder utilizar programas o librerías.

Análisis de resultados

A continuación se especificaran los objetivos que se cumplieron en el proyecto programado 4:

- Se pudo diseñar la aplicación web para mostrar las consultas, utilizando el framework Sinatra.
- Se logró obtener el contenido de Bandcamp, para extraer la información solicitada para cada banda (nombre, álbum, Url).
- El usuario puede realizar búsquedas ya sea por el nombre de un género musical o por el lugar de ubicación.
- En la página solicitada se pueden observar los primeros diez resultados de las búsquedas de las bandas, con su respectiva imagen, nombre, álbum y Url.
- Se logró conectarse al API de Twitter, para que el usuario pueda realizar tweets sobre una banda elegida.

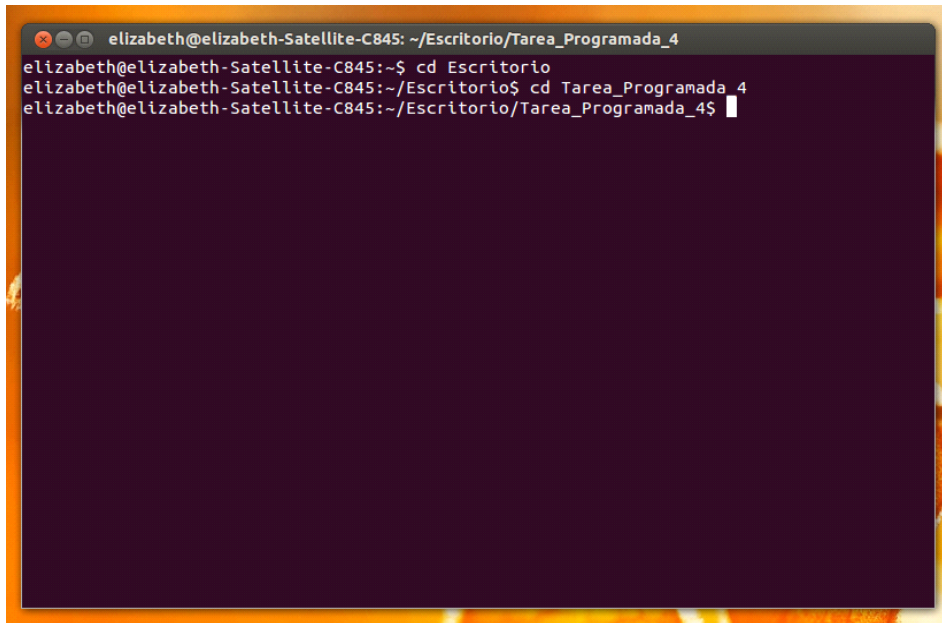
Objetivos que no se lograron:

- Obtener el resultado de que si la banda es gratuita o pagada.
No se pudo cumplir con este objetivo dado que hasta el final nos dimos cuenta que también había que incluir ese apartado. Al estar contra tiempo no lo pudimos incorporar.
Pero sabemos que esa información se obtenía de forma similar a como se extrajo la otra información pedida.

Manual de usuario

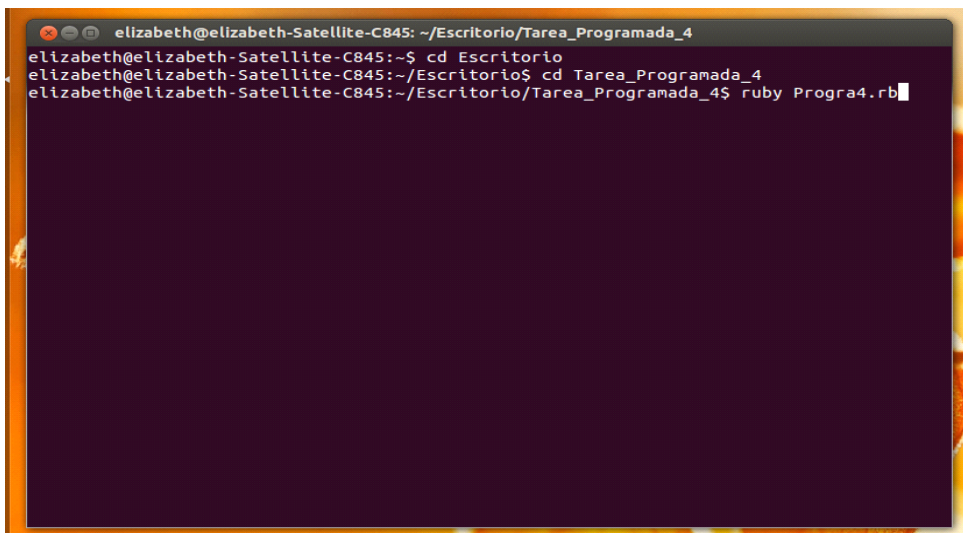
Pasos para utilizar la aplicación:

1. Primeramente se debe abrir la terminal y ubicarse en el directorio donde se encuentra el proyecto.

A terminal window with a dark purple background and orange window frame. The title bar reads 'elizabeth@elizabeth-Satellite-C845: ~/Escritorio/Tarea_Programada_4'. The terminal shows three lines of commands: 'cd Escritorio', 'cd Tarea_Programada_4', and the prompt returns to '~/. The cursor is at the end of the third line.

```
elizabeth@elizabeth-Satellite-C845: ~/Escritorio/Tarea_Programada_4
elizabeth@elizabeth-Satellite-C845:~$ cd Escritorio
elizabeth@elizabeth-Satellite-C845:~/Escritorio$ cd Tarea_Programada_4
elizabeth@elizabeth-Satellite-C845:~/Escritorio/Tarea_Programada_4$
```

2. Luego escribe ruby y el nombre del proyecto .erb.

A terminal window with a dark purple background and orange window frame. The title bar reads 'elizabeth@elizabeth-Satellite-C845: ~/Escritorio/Tarea_Programada_4'. The terminal shows three lines of commands: 'cd Escritorio', 'cd Tarea_Programada_4', and 'ruby Progra4.erb'. The prompt returns to '~/. The cursor is at the end of the third line.

```
elizabeth@elizabeth-Satellite-C845: ~/Escritorio/Tarea_Programada_4
elizabeth@elizabeth-Satellite-C845:~$ cd Escritorio
elizabeth@elizabeth-Satellite-C845:~/Escritorio$ cd Tarea_Programada_4
elizabeth@elizabeth-Satellite-C845:~/Escritorio/Tarea_Programada_4$ ruby Progra4.erb
```


3. Luego al presionar enter se le desplegara una informacion similar a la mostrada a continuación, para el siguiente paso debe tener en cuenta el puerto port.

```
elizabeth@elizabeth-Satellite-C845: ~/Escritorio/Tarea_Programada_4
elizabeth@elizabeth-Satellite-C845:~$ cd Escritorio
elizabeth@elizabeth-Satellite-C845:~/Escritorio$ cd Tarea_Programada_4
elizabeth@elizabeth-Satellite-C845:~/Escritorio/Tarea_Programada_4$ ruby Progra4.rb
Progra4.rb:140: warning: class variable access from toplevel
[2013-11-21 09:44:44] INFO WEBrick 1.3.1
[2013-11-21 09:44:44] INFO ruby 2.0.0 (2013-06-27) [i686-linux]
== Sinatra/1.4.4 has taken the stage on 4567 for development with backup from WEBrick
[2013-11-21 09:44:44] INFO WEBrick::HTTPServer#start: pid=3848 port=4567
```

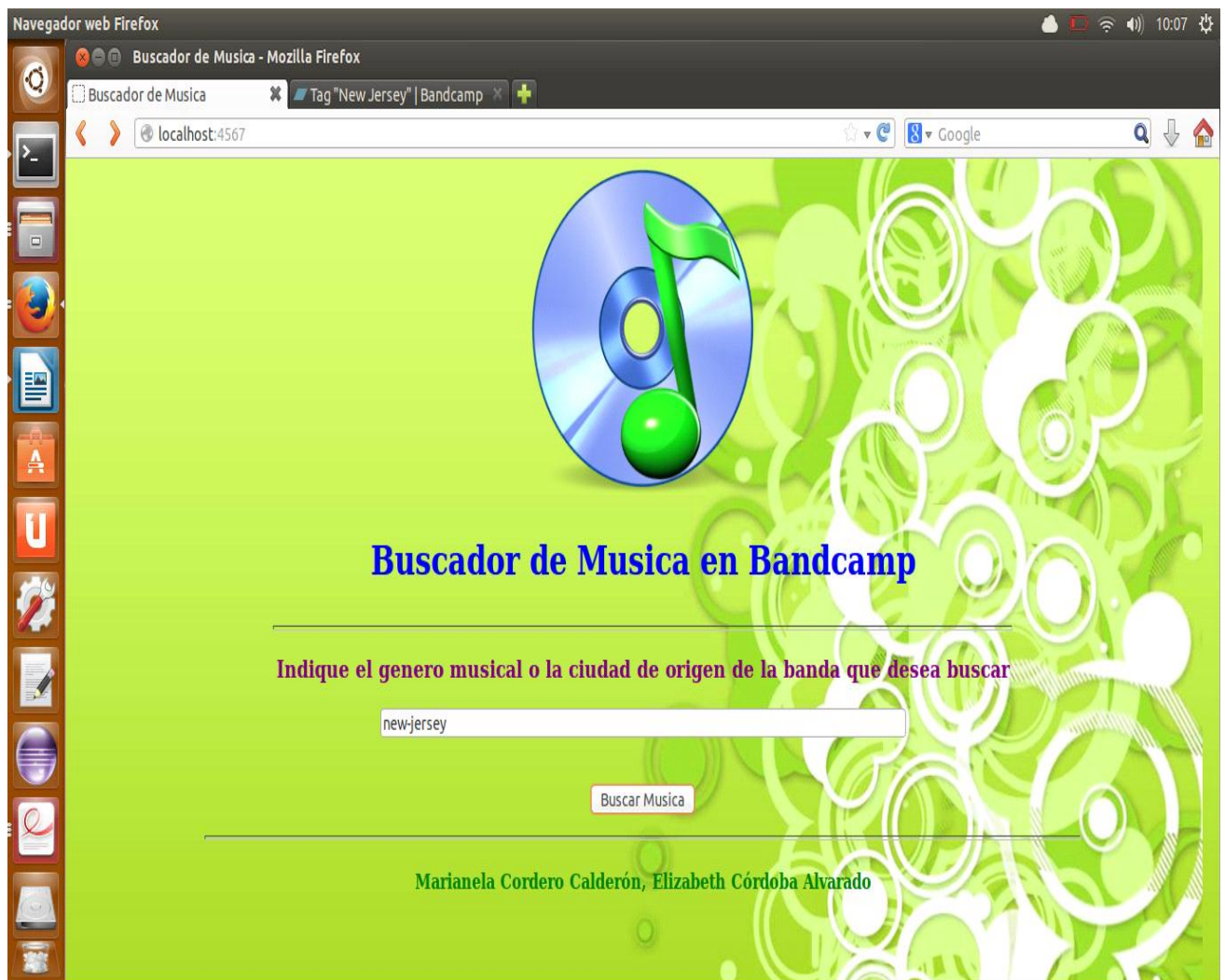
4. Abrir el navegador y escribir localhost: y el port mostrado en el paso anterior. Así se muestra la página inicial de la aplicación para la búsqueda en bandcamp.



5. Para realizar la búsqueda debe ingresar al criterio de búsqueda. Puede hacerlo por género musical, como se muestra a continuación.



También lo puede hacer por localización.






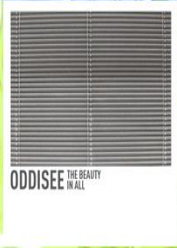




6. Al presionar el botón “Buscar Música”, se redirecciona a la página de resultados con la información de las primeras 10 bandas que cumplen con el criterio de búsqueda.

Cada resultado posee el nombre de la banda, el nombre del álbum, la imagen del álbum y el url de la banda en bandcamp. Además de un botón para realizar tweets.

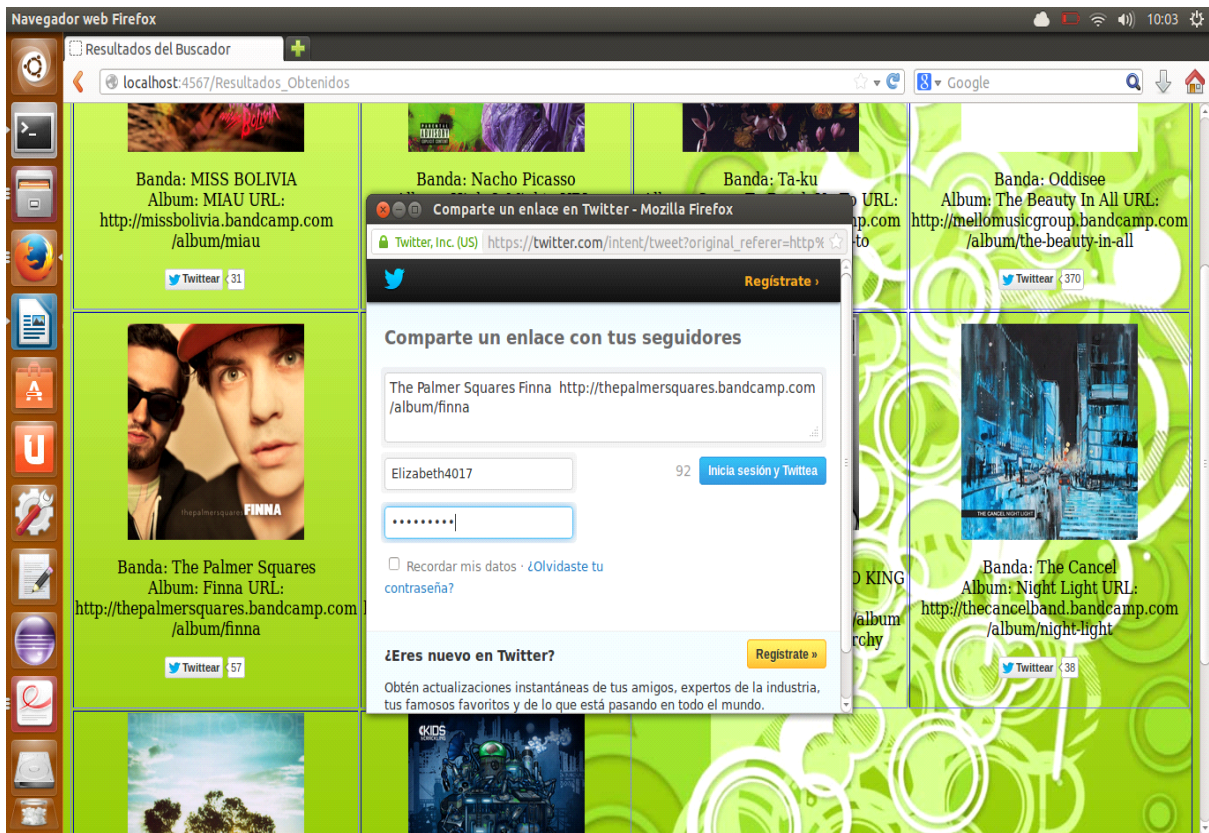
Resultados del Buscador - Mozilla Firefox

localhost:4567/Resultados_Obtendidos

Resultado de la búsqueda

 <p>Banda: MISS BOLIVIA Album: MIAU URL: http://missbolivia.bandcamp.com/album/miau</p> <p>Twitter 31</p>	 <p>Banda: Nacho Picasso Album: High & Mighty URL: http://nachopicasso.bandcamp.com/album/high-mighty</p> <p>Twitter 178</p>	 <p>Banda: Ta-ku Album: Songs To Break Up To URL: http://huhwhatwhere.bandcamp.com/album/songs-to-break-up-to</p> <p>Twitter 527</p>	 <p>Banda: Oddisee Album: The Beauty In All URL: http://mellomusicgroup.bandcamp.com/album/the-beauty-in-all</p> <p>Twitter 370</p>
 <p>Banda: The Palmer Squares Album: Finna URL:</p>	 <p>Banda: Man Of Boom Album: Back To The Boom URL:</p>	 <p>Banda: ANARCHY Album: DGKA (DIRTY GHETTO KING) URL:</p>	 <p>Banda: The Cancel Album: Night Light URL:</p>

7. Si desea twittear la información de alguna de las bandas debe presionar el botón de “Twittear”, el cual lo redirecciona a la página de twitter para realizar tweets. En esta página, en el espacio donde se escribe lo que se desea compartir, ya tiene la información de la banda (nombre, album y url). Por lo que solamente se debe iniciar sesión y realizar el tweet.



8. Al realizar el tweet, la página se cierra automáticamente. Y el tweet ya se encuentra publicado.

Conclusión personal

En la realización de la última tarea programada se tuvo la experiencia de trabajar con conceptos nuevos, por lo que al final del proyecto contamos con un mayor conocimiento.

Algunos de los puntos son: aprendizaje en el lenguaje ruby (utilización de conceptos de Programación Orientación a Objetos), creación de páginas web (Sinatra), utilización de gemas, como conectarse a APIs, realizar tweets, extracción de información de una página por medio del manejo de su HTML.

Al inicio se presentaron varios problemas en cómo obtener la información de Bandcamp teniendo en cuenta que el API de dicha página no está disponible, por lo que tuvimos que dedicar varias horas a la investigación e implementación de los métodos elegidos.

El mismo problema surgió a la hora de querer realizar tweets, dado que al querer poner require twitter y require Oauth nos daba un error que no pudimos arreglar, aunque dedicamos varias horas a ese punto, por lo que decidimos realizar nuevamente búsquedas en la web para ver de qué forma podíamos conectarnos al API de twitter, para poder efectuar los tweets. Después de tanto intentar conseguimos un botón que permite conectarse al API de twitter y por lo tanto realizar los tweets.

En conclusión la tarea fue un gran aprendizaje en muchos de los puntos antes mencionados.

Bibliografía

- Assistance Technique (s. f.). *Analizar una tabla HTML usando Hpricot (Ruby)* Recuperado el 21 de Noviembre del 2013, de <http://es.softuses.com/104995>
- Guía para escribir documentos HTML (s. f.). Recuperado el 21 de Noviembre del 2013, de <http://www.uv.es/jac/guia/indice.htm>
- Hpricot (s. f.). Recuperado el 21 de Noviembre del 2013, de <https://github.com/hpricot/hpricot>
- Ruby (s. f.). *RubyGems* Recuperado el 21 de Noviembre del 2013, de <http://rubygems.org/gems>
- RubyGems.org (s. f.). Recuperado el 21 de Noviembre del 2013, de <http://rubygems.org/gems>
- Sinatra (s. f.). Recuperado el 21 de Noviembre del 2013, de <http://www.sinatrarb.com/intro-es.html>
- Twitter (s. f.). *Tweet Button* Recuperado el 21 de Noviembre del 2013, de <https://dev.twitter.com/docs/tweet-button>
- [General] OT: HTML Parser (2008, 26 de Septiembre). Recuperado el 21 de Noviembre del 2013, de <http://www.softwarelibre.org.bo/pipermail/general/2008-September/024289.html>