



Tarea programada #4

Tecnológico de Costa Rica
Administración de Tecnologías de Información

Profesor: Andrei Fuentes

Estudiantes: Fernanda Fernández

Miguel Ángel González

Yesenia Montiel

Introducción

El objetivo de la tarea es familiarizarse con el desarrollo de aplicaciones web en Ruby, usando el paradigma de Orientación a Objetos. La idea es desarrollar una aplicación de búsqueda de fotos.

Descripción del problema

En los últimos años, se han popularizado varios servicios de publicación de fotos, tales como Flickr o Instagram. Dichos servicios permiten a los usuarios subir sus fotos y categorizarlas de acuerdo a palabras claves.

Se deberá desarrollar una aplicación web de búsqueda de fotos.

La aplicación deberá tener un formulario de búsqueda, en el cual los usuarios podrán especificar un criterio de búsqueda (pueden ser palabras o un texto). Posteriormente, la aplicación deberá usar el API de búsqueda del servicio de fotos utilizado, y obtener las fotos que concuerden con el criterio de búsqueda.

El número de fotos que se obtenga por cada búsqueda podrá ser configurable, ya sea desde la misma aplicación, o desde un archivo de configuración del lado del servidor.

Para cada resultado, deberán mostrar la imagen propiamente, el título o leyenda de la foto (según corresponda), el autor, y el url de la foto original. La información de los resultados la deberán manejar a nivel interno mediante objetos.

El programa deberá ser una aplicación web, para lo cual se usará un framework de creación de desarrollo de web apps llamado Sinatra.

Una vez que se hayan obtenido las fotos correspondientes a la búsqueda, se podrán hacer publicaciones de esos resultados en Twitter. (Opcional)

Diseño del Programa

Descripción de funciones:

La función `initialize()` nos va a guardar una llave, y una llave secreta, las cuales nos servirán para identificar el programa en las cuentas de los sistemas.

La función `ObtFotos(filtro, numero)` es la que va a obtener las fotos, va a recibir el parámetro de búsqueda bajo el nombre de filtro, y la cantidad de imágenes a mostrar bajo el nombre de numero, este se conectara a la base de flirck, que realizara una búsqueda con esos datos dados y devolverá una serie de datos.

La función `IngreInfo()` guradara todos los datos obtenidos en una variable en donde se guardaran todos los url de la búsqueda.

La función `ObtInfo()` es la que toma una foto de la lista álbum y almacenara los datos propios de la imagen en varias variables, como su titulo, su descripción y su url, para su posterior impresión en pantalla.

La función `AsigImagen()` es la que va a asigar la imagen a la interfaz.

La clase `class twitter` es la que permitirá el logueo y el twiteo, la cual incluye la función `initialize()` que pedirá datos para iniciar el logueo, también la función `conexion()`, tomara datos anteriores y realizara la conexión. Por otra parte esta la función `twettear(lista_tweets)` la cual esta predefinida para realizar un tweet de la información que se encuentre en la variable `lista_tweets`, cada ves que se indique que se desea realizar el tweet, esto mediante el encendido del botón de tweet en la aplicación.

Tarea Programada 4: Ruby

Buscador de Imágenes

El desarrollo de la parte web y su respectiva conexión, es complicada de explicar mediante funciones por lo que se mostrara el código y se explicara.

```
<HTML>
  <HEAD>
    <TITLE>Tarea Programada 4 - Flickr-Sinatra-Twitter</TITLE>
  </HEAD>
  <body >
    <IMG SRC = "http://pabloadaraya.org/wp-content/uploads/2012/01/ruby_logo.jpg" >
    <H5><LEFT>Tecnologico de Costa Rica <P>
      Administracion de Tecnologias de Informacion <P>
      TI-3404 Lenguajes de Programacion <P>
      Prof. Andrei Fuentes L.
      II Semestre 2012
    </LEFT></H5>
    <BODY BGCOLOR="FFFFFF" TEXT= "000000">
    <HR><CENTER>
      Bienvenido a nuestro proyecto:<P>
      Por favor digite su busqueda:
      <CENTER>
    <FORM action="/busqueda" method="post"> <P></P>
    <BR><CENTER>
      <input type="text" name="campo1" value="" size="20">
      <BR>
      <P>Numero de muestra:</P>
      <input type="text" name="campo2" value="" size="20">
      <br>
      <INPUT type="submit" value="Realizar busqueda">
    </FORM>
  </BODY>
</HTML>
```

Este código en html, creara la pagina principal en donde el usuario podrá desempeñarse, con funciones básicas html se crean los títulos, los entrys, los botones y labels.

```
<HTML>
  <HEAD>
    <TITLE>Tarea Programada 4 - Flickr-Sinatra-Twitter</TITLE>
  </HEAD>
  <body>
    <IMG SRC = "http://pabloadaraya.org/wp-content/uploads/2012/01/ruby_logo.jpg" >
    <H5><LEFT>Tecnologico de Costa Rica <P>
      Administracion de Tecnologias de Informacion <P>
      TI-3404 Lenguajes de Programacion <P>
      Prof. Andrei Fuentes L.
      II Semestre 2012
    </LEFT></H5>
    <BODY BGCOLOR="FFFFFF" TEXT= "000000">
    <HR><CENTER>
      Bienvenido a nuestro proyecto:<P>
      <CENTER>
    <HR>
    <P>
      <TABLE BORDER>
        <% for foto in $Album do %>
          <% $Photo = foto[1]%>
          <TR>
            <TD><IMG SRC = <% $Photo %> width=400 height=200/></TD>
            <TD> <%foto[0]%> </TD>
          </TR>
          <% end %>
        </TABLE>
      </P>
    </BODY>
  </HTML>
```

Este código creara una nueva pantalla en la cual se recibirá la información de la imagen, pero mantendrá un perfil vacío.

Tarea Programada 4: Ruby

Buscador de Imágenes

```
<HTML>
<HEAD>
  <TITLE>Tarea Programada 4 - Flickr-Sinatra-Twitter</TITLE>
</HEAD>
<body>
  <IMG SRC = "http://pabloaraya.org/wp-content/uploads/2012/01/ruby_logo.jpg" >
  <H5><LEFT>Tecnologico de Costa Rica <P>
    Administracion de Tecnologias de Informacion <P>
    TI-3404 Lenguajes de Programacion <P>
    Prof. Andrei Fuentes L.
    II Semestre 2012
  </LEFT></H5>
  <BODY BGCOLOR="FFFFFF" TEXT= "000000">
  <HR><CENTER>
    Resultado de tu busqueda:<P>
    <CENTER>

  <P>
    <IMG SRC = <%= $Foto %> width=400 height=300/>
  </P>
  <P> El titulo de la imagen es <%= $Titulo %> </P>
  <FORM action="/siguiente" method="post"> <P></P>
  <INPUT type="submit" value="Siguiente">
  <FORM action="/tweet" method="post"> <P></P>
  <INPUT type="submit" value="Tweeatear!">
</BODY>
</HTML>
```

Este código creara una nueva pantalla en la cual se recibirá la información de la imagen, acomodara la misma, y mostrara propiamente la imagen en pantalla.

```
<HTML>
<HEAD>
  <TITLE>Tarea Programada 4 - Flickr-Sinatra-Twitter</TITLE>
</HEAD>
<body background="imagen.jpg">
  <H5><LEFT>Tecnologico de Costa Rica <P>
    Administracion de Tecnologias de Informacion <P>
    TI-3404 Lenguajes de Programacion <P>
    Prof. Andrei Fuentes L.
    II Semestre 2012
  </LEFT></H5>
  <BODY BGCOLOR="FFFFFF" TEXT= "000000">
  <HR><CENTER>
    Gracia por tweeteatear desde nuestra aplicacion
    <CENTER>
  </FORM>
</BODY>
</HTML>
```

Por ultimo este código es el que mostrara un mensaje de agradecimiento por tweeteatear desde la aplicación.

Librerías usadas:

El programa se desarrolla en lenguaje orientado a objetos (Ruby), se tuvo que investigar como se da la conexión con la página Twitter y además como poder recuperar los datos de la página Bandcamp, además de la apertura de URL desde el programa, para ello se utilizaron 3 librerías, que se describen a continuación:

`sudo apt-get install ruby`: instalar ruby

Rubygems: Esta biblioteca se encarga de darnos el soporte para poder descargar más bibliotecas al sistema. Las maneja por medio de “gems”.

`sudo gems install rubygems-update`: actualiza rubygems

Open-Uri: Esta biblioteca es necesaria para poder tener acceso a URL's desde la aplicación, abrirlas y manipularlas con elementos de la biblioteca Hpricot

Sinatra: Base para poder recuperar la información del HTML, se pueden asignar tags para realizar las búsquedas.

Twitter_oauth: es la biblioteca responsable de darnos el token para ingresar a Twitter desde la aplicación. Además de generar el cliente con la respectiva llave y seguridad que solicita el API.

Decisiones Tomadas

1. Se utiliza el sistema operativo Ubuntu y ruby build-essential 1.9.3 para poder ejecutar la aplicación.
2. La aplicación será capaz de realizar infinidad de búsquedas de imagen.
3. Se utilizan bibliotecas que facilitan el parseo de las páginas a buscar, entre ellas se decide usar Flirck por su simplicidad y amplia documentación disponible en diferentes medios.
4. El programa muestra el título, la imagen y una opción para mostrar la siguiente imagen, y una opción para twittear.
5. Se decide mostrar al usuario los resultados obtenidos en una aplicación web la cual es en lenguaje html, por su simplicidad a la hora de desarrollar una pagina web.
7. Se decide desarrollar una pagina web simple para el uso del usuario donde solo aparezca información relevante al proyecto, y dos casillas para ingresar el parámetro de búsqueda y la cantidad de imágenes a buscar.
6. El método de autenticación del usuario en Twitter se realiza en la página, cuando se desea tener acceso a la aplicación, luego de este proceso el token será enviado a la aplicación en señal de confirmación.

Análisis de Resultados

La aplicación desarrollada cumple con los todos los objetivos que se plantean, a continuación los puntos más importantes:

- Se logra la conexión de la aplicación con el API de Twitter y el API de Flickr.

- Dada una palabra de búsqueda, se logra buscar en el HTML de la página Bandcamp y retornar los resultados a la estructura que se planteó en la aplicación.

- Se muestran al usuario los resultados de la búsqueda antes de ser tweet's

Se logra crear la conexión con twitter y con la cuenta del usuario para poder realizar los tweet's.

- Se valida la conexión a internet, la autorización de la aplicación en el API, se muestran cantidades deseadas de resultados.

Tarea Programada 4: Ruby

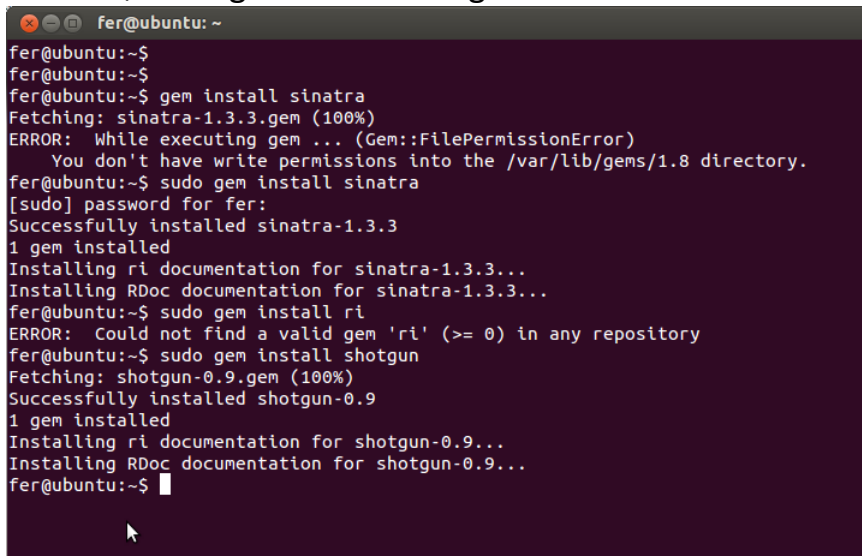
Buscador de Imágenes

Manual de Usuario

Para poner a funcionar la aplicación debe de instalar las librerías antes descritas para el funcionamiento total de la aplicación. Primero debemos abrir la terminal y digitar los siguientes comandos de instalación.

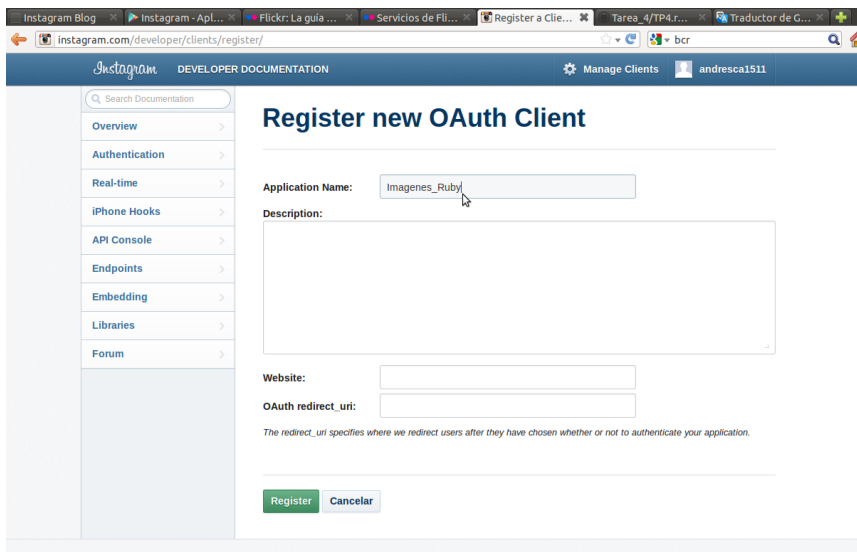
```
~$ sudo gem install Sinatra
```

```
~$ sudo gem install shotgun
```



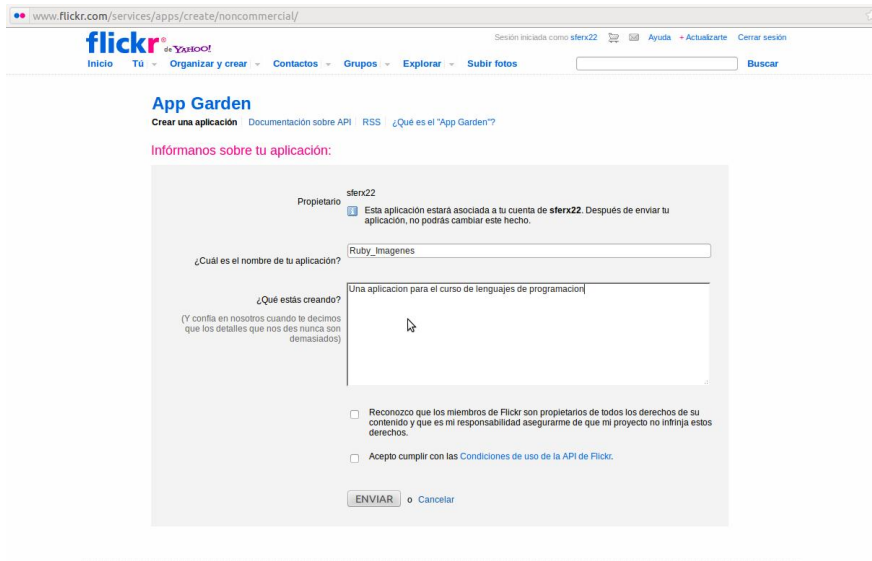
```
fer@ubuntu: ~  
fer@ubuntu:~$  
fer@ubuntu:~$  
fer@ubuntu:~$ gem install sinatra  
Fetching: sinatra-1.3.3.gem (100%)  
ERROR: While executing gem ... (Gem::FilePermissionError)  
You don't have write permissions into the /var/lib/gems/1.8 directory.  
fer@ubuntu:~$ sudo gem install sinatra  
[sudo] password for fer:  
Successfully installed sinatra-1.3.3  
1 gem installed  
Installing ri documentation for sinatra-1.3.3...  
Installing RDoc documentation for sinatra-1.3.3...  
fer@ubuntu:~$ sudo gem install ri  
ERROR: Could not find a valid gem 'ri' (>= 0) in any repository  
fer@ubuntu:~$ sudo gem install shotgun  
Fetching: shotgun-0.9.gem (100%)  
Successfully installed shotgun-0.9  
1 gem installed  
Installing ri documentation for shotgun-0.9...  
Installing RDoc documentation for shotgun-0.9...  
fer@ubuntu:~$
```

Ahora bien debemos de ingresar a las paginas de Instagram y de flirck para poder obtener los apis necesarios de desarrollo, en nuestro caso seleccionamos los apis de flirck ya que son un poco mas comprensibles y sencillos a la hora de realizar la instalación de los mismos.



Tarea Programada 4: Ruby

Buscador de Imágenes



The screenshot shows the Flickr App Garden page for creating a non-commercial application. The page is titled "App Garden" and includes links for "Crear una aplicación", "Documentación sobre API", "RSS", and "¿Qué es el 'App Garden?'". The form is for the user "sferx22" and includes a warning that the application will be associated with their account. The application name is "Ruby_Imagenes" and the description is "Una aplicación para el curso de lenguajes de programación". There are two checkboxes for terms of service, both of which are unchecked. The form has "ENVIAR" and "Cancelar" buttons.

www.flickr.com/services/apps/create/noncommercial/

Sesión iniciada como sferx22 Ayuda Actualizarte Cerrar sesión

flickr by Yahoo!

Inicio Tú Organizar y crear Contactos Grupos Explorar Subir fotos Buscar

App Garden

Crear una aplicación Documentación sobre API RSS ¿Qué es el "App Garden?"

Infórmanos sobre tu aplicación:

Propietario sferx22

Esta aplicación estará asociada a tu cuenta de sferx22. Después de enviar tu aplicación, no podrás cambiar este hecho.

¿Cuál es el nombre de tu aplicación? Ruby_Imagenes

¿Qué estás creando? Una aplicación para el curso de lenguajes de programación

(Y confía en nosotros cuando te decimos que los detalles que nos das nunca son demasiados)

☐ Reconozco que los miembros de Flickr son propietarios de todos los derechos de su contenido y que es mi responsabilidad asegurarme de que mi proyecto no infrinja estos derechos.

☐ Acepto cumplir con las Condiciones de uso de la API de Flickr.

ENVIAR o Cancelar

Luego debe correr la aplicación de la siguiente manera:

~\$ ruby -rubygems Principal.rb

```
fer@ubuntu:~$ cd Escritorio/final
fer@ubuntu:~/Escritorio/final$ ruby -rubygems Principal.rb
[2012-11-26 06:46:52] INFO WEBrick 1.3.1
[2012-11-26 06:46:52] INFO ruby 1.8.7 (2011-06-30) [i686-linux]
== Sinatra/1.3.3 has taken the stage on 4567 for development with backup from WEBrick
[2012-11-26 06:46:57] INFO WEBrick::HTTPServer#start: pid=6118 port=4567
```

Tarea Programada 4: Ruby

Buscador de Imágenes

Habiendo hecho los logueos en las respectivas plataformas, se procede a abrir una pestaña del navegador predeterminado, con la dirección:

Localhost:4567

El cual va a llamar los archivos .erb en los cuales se encuentra la información de la aplicación web a ser mostrada en el buscador.



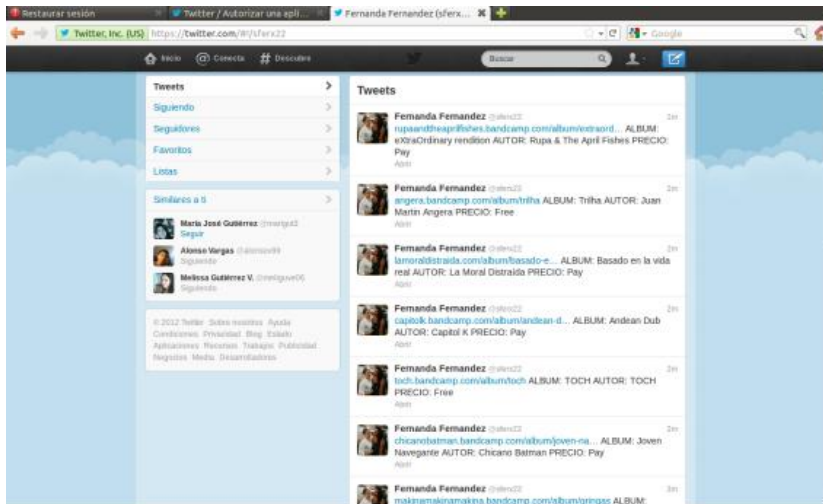
Así se mostrará la ventana principal de la aplicación. En la cual en la primera casilla vacía se introducirá el parámetro de búsqueda, y en la segunda casilla se introducirá la cantidad de imágenes que deseamos que nos muestre la aplicación.



Tarea Programada 4: Ruby

Buscador de Imágenes

Habiendo obtenido una búsqueda deseada, se mostrara la imagen relacionada al tema de búsqueda y su debido titulo, además aparecerán dos botones mas en los cuales el botón de “siguiente” nos llevara a la siguiente búsqueda, o a otra imagen relacionada con el tema, y el botón de “twittear” nos enviara a la pagina de twitter, en la cual debemos loguearnos para realizar un twit con el respectivo url de la imagen deseada.



Conclusiones

Durante el desarrollo de esta tarea programada se desarrollaron diferentes habilidades informáticas.

Como lo son la recapitulación de los requerimientos, el análisis profundo de los mismos, así como la búsqueda de información relativa a los temas, dentro de ellas búsqueda de información sobre el lenguaje de programación Ruby, el sistema Sinatra, y además el como implementar aplicaciones web en html.

Dentro de las complicaciones adquiridas a la hora del desarrollo de la tarea, fue el del traslado de la información del código en ruby, a la aplicación web en html, también el poder hacer los logings necesarios en Flirck y en twitter.

Además como conocimiento adquirido y conclusiones finales, nos gusto la parte de la tarea en la cual se tenia que realizar desde cero la aplicación web, ya que creemos que en un mundo que se comunica altamente por medio de sitios web, es de vital importancia tener conocimientos sobre como se manejan estos sistemas, y el como poder construir uno si fuese una tarea de trabajo profesional.

Desempeño en la Tarea:

Integrantes	Investigación	Desarrollo	Documentación
Fernanda	100%	100%	100%
Miguel Ángel	100%	100%	100%
Yesenia	100%	100%	100%

Bibliografía

- Sinatra. (2012). Recuperado el 12 de Noviembre de 2012, de <http://www.sinatrarb.com/>
- Lenguaje Ruby. (2012). Recuperado el 12 de Noviembre de 2012, de <http://www.ruby-lang.org/es/>
- APIS Flirck. (2012). Recuperado el 15 de Noviembre de 2012, de <http://www.flickr.com/services/api/>
- APIS Twitter. (2012). Recuperado el 22 de Noviembre de 2012 de <https://dev.twitter.com/docs>
- HTML. (2012). Recuperado el 22 de Noviembre de 2012 de <http://www.w3schools.com/html/default.asp>
- Index of Classes & Methods in open-uri: Ruby Standard Library Documentation (Ruby 1.9.3) (s. f.). Recuperado el 6 de Junio del 2012, de <http://www.ruby-doc.org/stdlib-1.9.3/libdoc/openuri/rdoc/>
- Moomerman/twitter_oauth · GitHub (s. f.). Recuperado el 7 de Junio del 2012, de https://github.com/moomerman/twitter_oauth