

ESCUELA SUPERIOR POLITECNICA DEL LITORAL

FACULTAD DE INGENIERIA EN ELECTRICIDAD Y

COMPUTACION

“ DIRECTORIO INTERACTIVO DE LA ESPOL ”

TESINA DE SEMINARIO

PREVIO A LA OBTENCION DEL TITULO DE:

INGENIERO EN CIENCIAS COMPUTACIONALES

ESPECIALIZACION SISTEMAS TECNOLOGICOS

E

INGENIERO EN CIENCIAS COMPUTACIONALES

ESPECIALIZACION SISTEMAS DE INFORMACION

PRESENTADO POR:

GONZALO ANTONIO ORDÓÑEZ RODRÍGUEZ

STALIN ALBERTO ARROYABE MERCHAN

GUAYAQUIL – ECUADOR

2010

AGRADECIMIENTO

Damos sinceras gracias a Dios, nuestro Señor, por darnos la vida y los recursos necesarios para alcanzar ésta meta anhelada.

A nuestras familias, por brindarnos su apoyo y confianza.

A nuestros amigos y compañeros que han sido parte de nuestra vida estudiantil.

A nuestros profesores y Director de Tesis, que han tenido mucha paciencia para enseñarnos y enrumbarnos correctamente.

A todos ellos, sinceras gracias por apoyarnos en la obtención del título de Ingeniería en Ciencias Computacionales.

DEDICATORIA

A Dios, a nuestros padres, hermanos, familiares y amigos por su apoyo y confianza a lo largo de nuestra vida.

A todos ellos, esperamos retribuirles de la mejor manera en el presente y futuro.

TRIBUNAL DE SUSTENTACION

MSc. Pedro Echeverría Briones
PROFESOR DEL SEMINARIO

MSc. Xavier Ochoa Chehab
PROFESOR DELEGADO DEL DECANO

DECLARACIÓN EXPRESA

“La responsabilidad del contenido de este Trabajo de Graduación, nos corresponde exclusivamente; y el patrimonio intelectual de la misma a la **ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL**”.

Stalin Alberto Arroyabe Merchán

Gonzalo Antonio Ordóñez Rodríguez

RESUMEN

En este proyecto se ha elaborado una Aplicación web llamada “Directorio Interactivo de la ESPOL”, que permite a los estudiantes, publicar información personal en su perfil de usuario.

La información es extraída desde las redes sociales más populares en nuestro medio (Facebook, Hi5, Youtube), y desde la bases de datos universitaria, para luego compartirla con los demás usuarios del sistema, formando de esta manera una red social politécnica.

Las redes sociales tomadas en cuenta para este proyecto son:

- Facebook y Hi5.- El “Directorio Interactivo de la ESPOL” permite importar los álbumes de la cuenta de usuario de alguna de estas redes sociales (previa configuración).

- Youtube.- El “Directorio Interactivo de la ESPOL” permite importar los videos, de la cuenta de un usuario (previa configuración).

Para elaborar la funcionalidad detallada anteriormente utilizamos:

- Gdata Library de Google para la comunicación con los servidores de Youtube.
- Servicios Web basados en REST, para la comunicación con los servidores de Hi5.
- Servicios de Web basados en SOAP, para la comunicación con los servidores de la ESPOL
- JAVASCRIPT y FQL para la comunicación con los servidores de Facebook.

Además tomamos en cuenta los estándares y directrices definidos por la W3C para el desarrollo de nuestra aplicación web.

INTRODUCCIÓN

La web 2.0 es una etapa de la web que ha definido nuevos proyectos en Internet y está preocupándose por brindar mejores soluciones para el usuario final.

La expansión de la Web 2.0 está brindando a los usuarios la posibilidad de influir significativamente en el diseño, utilización, elaboración de productos y servicios de los distintos sectores de la sociedad.

La infraestructura de la Web 2.0, posee nuevas tecnologías que han permitido con facilidad la publicación de información, además de poder compartirla con otros sitios Web.

Los sistemas de gestores de contenido (Content Management Systems, CMS) han aportado significativamente a la publicación de información de manera sencilla. Un ejemplo es la facilidad que tiene un usuario al gestionar su propio blog, sin la necesidad conocer de programación Web

La Web 2.0 toma en cuenta la necesidad de comunicación entre aplicaciones, y es por eso que se han desarrollado microformatos estandarizados, para compartir automáticamente información entre sitios Web.

Las Redes Sociales son un ejemplo de la web 2.0. Son formas de interacción social con intercambio dinámico entre personas, grupos e instituciones en contextos de alta complejidad. Son sistemas abiertos y en construcción permanente, que involucran a múltiples conjuntos de usuarios con las mismas necesidades y problemáticas. Los usuarios de las redes sociales son prosumidores, porque producen y consumen recursos.

Las nuevas tecnologías de la información (web 2.0), han revolucionado la economía tradicional, hasta convertirla en una economía con multitudes inteligentes.

La ESPOL cuenta con un sin números de tecnologías que ayudan al funcionamiento estratégico y operacional de la institución, lo que ha permitido cumplir con efectividad y eficiencia las responsabilidades adquiridas con la sociedad. Los estudiantes de la Comunidad Politécnica, hacemos uso diario de la tecnología para cumplir con nuestras labores, por eso tenemos un alto grado de usabilidad en el manejo de tecnologías.

Los estudiantes dedicamos un tiempo muy significativo a la interacción con las siguientes redes sociales: Facebook Hi5 y Youtube. En estas redes sociales se encuentra información personal muy valiosa de acceso público, siempre y cuando se cumpla con las políticas de privacidad declaradas en cada sitio.

La universidad no cuenta con una plataforma tecnológica, que permita el fortalecimiento de la relaciones entre los estudiantes. Es por eso que surge la idea del “Directorio Interactivo de la ESPOL”; él cual permitirá la búsqueda y acceso (no restringido) a la información de los estudiantes. La información es extraída de las redes sociales anteriormente mencionadas.

Las motivaciones para realizar este trabajo de Tesis son:

- No existe una Herramienta o Sistema universitario, que permita unir más estrechamente los lazos de compañerismo y amistad entre los estudiantes.

- La mayoría de los estudiantes le dedican un pequeño, pero significativo tiempo a la interacción con las redes sociales antes mencionadas, colocando información personal a cada instante.

- Está información puede ser accedida mediante APIs y Servicios Web, y por eso que se convierte en una ventaja para “Directorio Interactivo DE LA ESPOL”, dado que un usuario no tendrá que ingresar gran cantidad de información para tener personalizado su perfil.

Los principales objetivos planteados son:

- Ayudar a enlazar más estrechamente el compañerismo y la amistad entre los estudiantes, mediante el acceso libre a la información personal de cualquiera de ellos.
 - Está información solo puede ser visualizada por los estudiantes, debido a que la única forma de ingreso es mediante usuario y contraseña ESPOL.
- Aprender, analizar, aplicar y evaluar la comunicación de aplicaciones web, mediante el uso de estándares, protocolos e interfaces de programación avanzada (APIs).

Con este proyecto de graduación, se espera fortalecer las relaciones entre los miembros de la comunidad, y aportar de manera indirecta al ascenso de posiciones en el ranking mundial de universidades, porque se espera que nuestra aplicación genere mayor tráfico al dominio de la ESPOL.

ÍNDICE GENERAL

RESUMEN.....	I
INTRODUCCIÓN.....	III
ÍNDICE GENERAL	VII
ABREVIATURAS	X
ÍNDICE DE FIGURAS.....	XI
ÍNDICE DE TABLAS.....	XIII
1. Comunicación con los Servicios web de la ESPOL	1
1.1 <i>Servicios Web</i>	1
1.2 <i>Protocolo SOAP</i>	2
1.3 <i>Lenguaje de descripción de servicios web (WSDL)</i>	3
1.4 <i>Uso del Protocolo</i>	4
1.4.1 <i>Requerimientos</i>	4
1.4.1.1 <i>Requerimiento 1:</i>	4
1.4.2.2 <i>Requerimiento 2:</i>	6
2. Comunicación con los Servicios Web de Facebook	10
2.1 <i>La red social y su impacto en la web [6]</i>	10
2.2 <i>REST [7]</i>	11
2.3 <i>Forma de comunicación con la red [8]</i>	13
2.3.1 <i>Interfaz REST</i>	13
2.3.2 <i>Java Script Client Library</i>	13
2.4 <i>Facebook Query Language (FQL) [9]</i>	14
2.5 <i>Lenguaje de Consulta</i>	15
2.6 <i>Tablas FQL [10]</i>	17
2.6.1 <i>Tabla Álbum</i>	19
2.6.2 <i>Tabla Photo</i>	20

3. Comunicación con los Servicios web de Hi5.....	22
3.1 Descripción de la red social [11]	22
3.2 Formas de comunicación [12]	23
3.2.1 Principales espacios de nombres.....	23
3.3 Uso del protocolo	24
3.3.1 Requerimiento.....	24
4. Comunicación con los servicios web de YouTube	27
4.1 Descripción de la Red Social [13]	27
4.2 Redifusión web [14]	28
4.2.1 RSS [15].....	29
4.2.2 ATOM [16]	30
4.2.3 Diferencias entre RSS y ATOM.....	30
4.2.3.1 Especificaciones	30
4.2.3.2 Integración con XML	31
4.2.3.3 Estandarización	32
4.3 Detalle del API [18]	34
4.4 Acerca de esta documentación [18].....	35
4.5 Versión de la API	39
5. Implementación - Pruebas - Manual de usuario	41
5.1 Diagrama de interacción de los módulos.....	41
5.2 Descripción de los módulos.....	42
5.2.1 Módulo ESPOL	42
5.2.1.1 Requisitos del módulo ESPOL.....	42
5.2.1.2 Archivos fuentes que contiene el módulo ESPOL	43
5.2.2 Módulo Facebook.....	44
5.2.2.1 Requisitos del módulo Facebook	44
5.2.2.2 Archivos fuentes que contiene el módulo Facebook	44
5.2.3 Módulo Hi5.....	45
5.2.3.1 Requisitos del módulo Hi5	45
5.2.3.2 Archivos fuentes que contiene el módulo Hi5	46
5.2.4 Módulo YouTube	47
5.2.4.1 Requisitos del módulo YouTube.....	47
5.2.4.2 Archivos fuentes que contiene el módulo Youtube	48
5.3 Pruebas de usabilidad	48
5.3.1 Pruebas sobre la versión 1.....	48
5.3.2 Pruebas sobre la versión 2.....	51
5.4 Manual de usuario	52

5.4.1 Ingresar al “Directorio Interactivo de la ESPOL”	52
5.4.2 Primera pantalla de ingreso al Sistema	53
5.4.3 Importar Álbumes de Facebook	54
5.4.4 Importar Álbumes de Hi5	59
5.4.5 Importar Videos de Youtube	63
5.4.6 Buscar Personas.....	67
Conclusiones y recomendaciones.....	72
<i>Conclusiones.-</i>	72
<i>Recomendaciones.-</i>	74
Referencias Bibliográficas	75

ABREVIATURAS

XML:	Lenguajes de marcas extensibles.
CSS:	Hojas de estilo en cascada.
XHTML:	Lenguaje extensible de marcado de hipertexto.
SOAP:	Protocolo de acceso simple a objetos.
REST:	Transferencia de Estado Representacional.
API:	Interfaz de programación de aplicaciones.
MVC:	Modelo Vista Controlador.
URL:	Localizador uniforme de recurso.
JSON:	Notación de objetos Java Script.

ÍNDICE DE FIGURAS

Figura 1.- Mecanismo de autenticación de un usuario	4
Figura 2.- Mensaje de petición en la autenticación de un usuario	5
Figura 3.- Mensaje de respuesta en la autenticación a un usuario.....	6
Figura 4.- Consulta de la información de un estudiante	6
Figura 5.- Mensaje de petición en la consulta de información de un usuario.....	7
Figura 6.- Mensaje de respuesta en la consulta de la información de usuario	8
Figura 7.- Diagrama de comunicación REST con el servicio web de Hi5.....	25
Figura 8.- Autenticación de un usuario de YouTube	38
Figura 9.- Diagrama de Interacción de módulos	41
Figura 10.- Pantalla de ingreso al sistema.....	53
Figura 11.- Primera pantalla de ingreso al Sistema.....	54
Figura 12.- Importar álbumes de Facebook.....	55
Figura 13.- Conectar a la aplicación de Facebook	56
Figura 14.- Ventana de autenticación con Facebook.....	57
Figura 15.- Álbumes del usuario en Facebook.....	58
Figura 16.- Álbumes de facebook con acceso público, de un usuario	59
Figura 17.- Importar álbumes de Hi5	60
Figura 18.- Conexión con los servidores de Hi5	61
Figura 19.- Álbumes del usuario en Hi5	62
Figura 20.- Álbumes de Hi5 con acceso público, de un usuario.....	63
Figura 21.- Importar videos de YouTube.....	64

Figura 22.- Conexión con los servidores de YouTube	65
Figura 23.- Videos del usuario en YouTube	66
Figura 24.- Videos de Youtube con acceso público, de un usuario	67
Figura 25.- Buscador de personas en nuestra aplicación	68
Figura 26.- Ingreso de parámetros de búsqueda.....	69
Figura 27.- Perfil del usuario encontrado (primera parte).....	70
Figura 28.- Perfil del usuario encontrado (segunda parte)	71

ÍNDICE DE TABLAS

Tabla I.- Librerías utilizadas para la comunicación con los servicios web de la ESPOL	9
Tabla II.- Descripción de la tabla álbum de facebook	20
Tabla III.- Descripción de la tabla foto de facebook.....	21
Tabla IV.- Principales espacios de nombres SOAP	24
Tabla V.- Librerías utilizadas para la comunicación con los servicios web de Hi5...	26
Tabla VI.- Librerías utilizadas para la comunicación con los servicios web de Youtube	40
Tabla VII.- Requisitos que se implementan en el modulo ESPOL	42
Tabla VIII.- Conjunto de archivos fuentes que contiene el módulo ESPOL	43
Tabla IX.- Requisitos que se implementan en el módulo Facebook.....	44
Tabla X.- Conjunto de archivos fuentes que contiene el módulo Facebook.....	45
Tabla XI.- Requisitos que se implementan en el módulo Hi5	46
Tabla XII.- Conjunto de archivos fuentes que contiene el módulo Hi5.....	46
Tabla XIII.- Requisitos que se implementan en el módulo Youtube	47
Tabla XIV.- Conjunto de archivos fuentes que contiene el módulo Youtube	48
Tabla XV.- Prueba de Usabilidad 1 de la versión 1	49
Tabla XVI.- Prueba de Usabilidad 2 de la versión 1	50
Tabla XVII.- Prueba de Usabilidad 3 de la versión 1	51
Tabla XVIII.- Prueba de Usabilidad 1 de la versión 2.....	51
Tabla XIX.- Prueba de Usabilidad 2 de la versión 2.....	52

CAPITULO 1

1. Comunicación con los Servicios web de la ESPOL

1.1 Servicios Web

Son un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones. Las aplicaciones pueden estar desarrolladas en lenguajes de programación diferentes, y ejecutarse sobre cualquier plataforma. Por lo general, el intercambio de datos se lo realiza sobre Internet, pero se lo puede realizar sobre cualquier red.

La interoperabilidad se consigue mediante la adopción de estándares abiertos. Las organizaciones OASIS y W3C son los comités responsables de la arquitectura y reglamentación de los servicios Web. Los dos tipos de comunicación más importantes son REST y SOAP [1].

1.2 Protocolo SOAP

Es un protocolo estándar de intercambio de mensajes en un sólo sentido, sin estado, pero las aplicaciones pueden crear patrones de interacción más complejos (por ejemplo, petición/respuesta, petición/respuestas múltiples, etc.).

SOAP no interfiere en la semántica de los datos que transfiere, ni tampoco en el enrutamiento de los mensajes.

Proporciona un marco de trabajo para que las aplicaciones puedan comunicarse de forma extensible. También proporciona una descripción completa de las acciones que debe realizar un nodo SOAP al recibir un mensaje SOAP [2].

Los mensajes constan de 3 secciones:

- Envelope (envoltura): Es el elemento raíz del mensaje para describir su contenido y la forma de procesarlo.
- Header (encabezado): Es la información de identificación del contenido. Un grupo de reglas de codificación para expresar los tipos de datos definidos por la aplicación.
- Body (cuerpo): Es el contenido del mensaje. Una convención para representar las llamadas y las respuestas a procedimientos remotos.

1.3 Lenguaje de descripción de servicios web (WSDL)

Es un formato XML que permite describir los servicios Web. La versión 1.0 fue la primera recomendación definida por la W3C. La versión 1.1 nunca alcanzó de consolidarse y la versión 2.0 es la recomendada actualmente.

La descripción se basa en la definición de los requisitos del protocolo y el formato de los mensajes. Con estas definiciones se puede interactuar con los servicios listados. Las operaciones y mensajes que soporta se describen en abstracto y para luego implementarlo en el protocolo de red.

Se usa a menudo en combinación con SOAP y XML Schema. Un programa cliente que se conecta a un servicio web puede leer el WSDL para determinar qué funciones están disponibles en el servidor. Los tipos de datos especiales se incluyen en el archivo WSDL en forma de XML Schema. El cliente puede usar SOAP para hacer la llamada a una de las funciones listadas en el WSDL [3].

1.4 Uso del Protocolo

1.4.1 Requerimientos

1.4.1.1 Requerimiento 1:

Verificar que los datos de inicio de sesión (usuario y contraseña) estén asociados algún estudiante politécnico



Figura 1.- Mecanismo de autenticación de un usuario

Solución:

Para la validación de un usuario del sistema se utilizó la operación de “autenticación” que tiene el Servicio web “directorioEspol” [4]

Para acceder a la operación “autenticación” se construyó un mensaje de petición en versión SOAP1.

```
POST /services/directorioEspol.asmx HTTP/1.1
Host: www.academico.espol.edu.ec
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://academico.espol.edu.ec/webservices/autenticacion"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <autenticacion xmlns="http://academico.espol.edu.ec/webservices/">
      <varUser>string</varUser>
      <varContraseña>string</varContraseña>
    </autenticacion>
  </soap:Body>
</soap:Envelope>
```

Figura 2.- Mensaje de petición en la autenticación de un usuario

El servidor envía el siguiente mensaje de respuesta:


```

POST /services/directorioEspol.asmx HTTP/1.1
Host: www.academico.espol.edu.ec
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://academico.espol.edu.ec/webservices/autenticacion"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <autenticacion xmlns="http://academico.espol.edu.ec/webservices/">
      <varUser>string</varUser>
      <varContraseña>string</varContraseña>
    </autenticacion>
  </soap:Body>
</soap:Envelope>

```

Figura 3.- Mensaje de respuesta en la autenticación a un usuario

1.4.2.2 Requerimiento 2:

Consultar la información académica de un miembro de la comunidad universitaria.

ESCENARIO 2: Consulta de la información académica de un miembro de la comunidad universitaria.

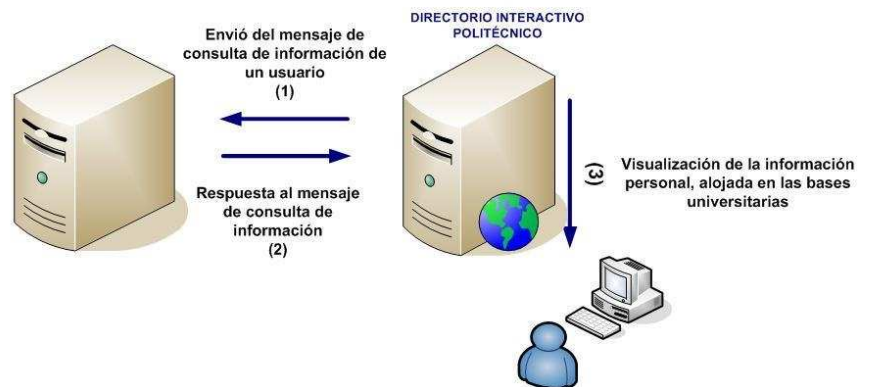


Figura 4.- Consulta de la información de un estudiante

Solución:

Si el resultado de respuesta del mensaje de autenticación es verdadero, se realiza la petición de “consulta de la información de un usuario” al Servicio web “wsSAAC” [5]

```

POST /Services/wsSAAC.asmx HTTP/1.1Host: www.academico.espol.edu.ec
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction:
"http://academico.espol.edu.ec/webservices/InformacionAcademicaEstudianteGet"
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <InformacionAcademicaEstudianteGet
xmlns="http://academico.espol.edu.ec/webservices/">
      <identificacion>string</identificacion>

      <Matricula>string</matricula>
    </InformacionAcademicaEstudianteGet>
  </soap:Body>
</soap:Envelope>

```

Figura 5.- Mensaje de petición en la consulta de información de un usuario

El servidor retorna los siguientes datos, asociados a un usuario:

```

<soap:Envelope><soap:Body><InformacionAcademicaEstudianteGetResponse><InformacionAcademicaEstudianteGetResult>

<xs:schema id="NewDataSet"><xs:element msdata:IsDataSet="true" msdata:UseCurrentLocale="true" name="NewDataSet"><xs:complexType><xs:choice maxOccurs="unbounded" minOccurs="0"><xs:element name="Table"><xs:complexType><xs:sequence>

    <xs:element minOccurs="0" name="APELLIDOS" type="xs:string"/>
    <xs:element minOccurs="0" name="NOMBRES" type="xs:string"/>
    <xs:element minOccurs="0" name="DIRECCION" type="xs:string"/>
    <xs:element minOccurs="0" name="TELEFONO" type="xs:string"/>
    <xs:element minOccurs="0" name="NOMBRE_CARRERA" type="xs:string"/>
    <xs:element minOccurs="0" name="EMAIL" type="xs:string"/>
    <xs:element minOccurs="0" name="CARRERA" type="xs:string"/>

</xs:sequence></xs:complexType></xs:element></xs:choice></xs:complexType></xs:element></xs:schema>

<diffgr:diffgram><NewDataSet><Table diffgr:id="Table1" msdata:rowOrder="0">
    <APELLIDOS>ARROYABE MERCHAN</APELLIDOS>
    <NOMBRES>STALIN ALBERTO</NOMBRES>
    <DIRECCION>GUAYACANES MZ 72 VILLA 45</DIRECCION>
    <TELEFONO>2820706 </TELEFONO>
    <NOMBRE_CARRERA>INGENIERÍA EN CIENCIAS COMPUTACIONALES ESPECIALIZACIÓN SISTEMAS DE INFORMACIÓN</NOMBRE_CARRERA>
    <EMAIL>sarroyab@espoledu.ec </EMAIL>
    <CARRERA>INGENIERÍA EN CIENCIAS COMPUTACIONALES</CARRERA>
</Table></NewDataSet></diffgr:diffgram></InformacionAcademicaEstudianteGetResult></InformacionAcademicaEstudianteGetResponse></soap:Body></soap:Envelope>

```

Figura 6.- Mensaje de respuesta en la consulta de la información de usuario

Para la comunicación con los Servicios web y el procesamiento del mensaje de respuesta se utilizaron las siguientes librerías:

Librería	Versión
Axis	1.0

Saaj	1.2
Axis-ant	1.4
commons-discovery	0.2
Jaxrpc	1.1
log4j-1.2.8	1.2.8
Mail	1.3.3
wsdl4j	1.5.1

Tabla I.- Librerías utilizadas para la comunicación con los servicios web de la ESPOL

CAPITULO 2

2. Comunicación con los Servicios Web de Facebook

2.1 La red social y su impacto en la web [6]

Facebook es una red social creada por Mark Zuckerberg. Originalmente era un sitio para estudiantes de la Universidad de Harvard, pero actualmente está abierto a cualquier persona que tenga una cuenta de correo electrónico. Los usuarios pueden participar en una o más subredes sociales (grupo) dependiendo de su situación académica, su lugar de trabajo o región geográfica.

Ha recibido mucha atención en la blogosfera y en los medios de comunicación por convertirse en una plataforma en donde terceros pueden desarrollar aplicaciones y hacer negocio a partir de la red

social. A pesar de ello, existe la preocupación acerca de su posible modelo de negocio, dado que los resultados en publicidad se han revelado como muy pobres.

A mediados de 2007 lanzó las versiones en francés, alemán y español para impulsar su expansión fuera de Estados Unidos, ya que sus usuarios se concentran en Estados Unidos, Canadá y Gran Bretaña. En enero de 2010, Facebook contaba con 350 millones de miembros, y traducciones a 70 idiomas.

2.2 REST [7]

La Transferencia de Estado Representacional (Representation State Transfer - REST) describe un estilo arquitectónico de sistemas en red.

El término fue utilizado por primera vez en el año 2000 durante una disertación doctoral por Roy Fielding, uno de los principales autores de la especificación HTTP.

REST contiene limitaciones y principios arquitectónicos. Si una aplicación o diseño cumple con esas limitaciones y principios, se considera RESTful.

La solicitud de interacción entre el cliente y el servidor no maneja estado. Por eso debe contener toda la información necesaria para ser comprendida.

El servidor posee recursos que representan el estado y la funcionalidad de la aplicación. Un recurso es un elemento que puede ser accedido por el cliente, algunos ejemplos de recursos son: objetos de aplicaciones, registros de bases de datos, algoritmos, etc. Cada recurso tiene acceso único a través de una URI (Universal Resource Identifier – identificador de recursos universal). Se usan métodos estándares de HTTP como GET, PUT, POST y DELETE. Los recursos se interconectan mediante hipervínculos.

Las limitaciones arquitectónicas REST permiten que la aplicación logre un alto grado de escalabilidad. También se reduce la latencia (suma de retardos temporales dentro de una red) en la interacción entre clientes y servidores.

2.3 Forma de comunicación con la red [8]

2.3.1 Interfaz REST

La API utiliza REST como interfaz. Esto significa que el API de Facebook hace las llamadas a métodos mediante el envío de peticiones HTTP GET o POST a los servicios web basados en REST. Casi cualquier lenguaje de programación puede ser utilizado para comunicarse a través de HTTP con el servidor de REST.

2.3.2 Java Script Client Library

Nos permite hacer llamadas a la API REST de Facebook mediante el método `fql_query(String query, Object onRequestCompleted)` que permite utilizar una interfaz al estilo SQL con mayor facilidad de consulta de datos sobre la Red Social. El cliente puede hacer sus propias consultas a las Tablas FQL.

2.4 Facebook Query Language (FQL) [9]

FQL es una forma de consulta de datos en Facebook. Hay otras formas de acceso mediante métodos del API. Muchas de las llamadas normales de la API son contenedores simples para las consultas de FQL. Una consulta típica se ve algo como esto:

```
SELECT name FROM user WHERE uid=1630691750;
```

Nos da como resultado Gonzalo Antonio Ordóñez Rodríguez. Con este simple ejemplo nos damos cuenta que FQL es muy fácil de utilizar. Las principales ventajas de utilizar FQL son las siguientes:

- XML liviano: reduce los costos de ancho de banda y análisis. En lugar de obtener toda la información disponible acerca de un gran conjunto de elementos, se puede obtener sólo los campos que se desee de acuerdo a la condición específica. Usted puede solicitar el conjunto específico de información mediante la adición de las restricciones a la cláusula WHERE, y sólo lista algunos campos en la cláusula SELECT.
- Mayor cantidad de requerimientos pueden reducir el número de solicitudes necesarias. A menudo, los datos que usted está tratando de obtener depende de los resultados de una llamada al método anterior. Por ejemplo, con la API tradicional, para

obtener los nombres de los amigos de una persona, primero llame `friends.get` y luego pasar el resultado directamente de nuevo a `users.getInfo`. Ahora sólo se puede ejecutar una consulta FQL que utiliza una subconsulta para obtener el conjunto de amigos - lo que reduce un viaje extra de ida y vuelta, y la latencia asociada a ella. Si se llama `fql.multiquery`, puede realizarse consultas aún más poderosas porque utiliza los resultados de una consulta en la convocatoria de otra consulta dentro de la misma llamada.

- Proporciona una única interfaz coherente y unificada, para todos los datos. En lugar de tener que aprender muchos métodos diferentes, puede hacer todas sus peticiones con una función que tiene un tipo de cambio constante. Además, si es necesario llamar a cualquiera de los métodos tradicionales, el retorno de XML es muy similar, por lo que el coste de transferencia es insignificante.

2.5 Lenguaje de Consulta

Si ya sabes SQL, debería ser bastante sencillo. Las consultas son de la forma `SELECT [campos] FROM [tabla] WHERE [condición]`

(también se puede añadir opcionalmente en las cláusulas ORDER BY y LIMIT como lo hacen en algunos gestores de base de datos). A diferencia de SQL, la cláusula FROM en FQL sólo puede contener una sola tabla.

FQL introduce una forma de tratar con las columnas, que pueden ser estructuras o conjuntos (por ejemplo, el "education_history" columna de la tabla "usuario", la cual contiene un arreglo de estructuras "education_info", o el "current_location" columna de la tabla de usuario la cual es una estructura de ubicación. Puede hacer referencia a las estructuras como un todo, o puede filtrar un solo campo dentro de la estructura mediante puntos.

Por ejemplo para obtener estructuras completas, usted puede hacer:

```
“SELECT education_history, current_location Education_history, current_location”
```

O para obtener los campos individuales en esa estructura usted puede hacer:

```
“SELECT education_history.name, current_location.zip Education_history.name,  
current_location.zip”
```

2.6 Tablas FQL [10]

Usted puede utilizar las siguientes tablas de referencia para la construcción de las consultas de FQL.

- album
- application
- comment
- cookies
- connection
- daily_metrics
- developer
- event
- event_member
- family
- friend
- friend_request
- friendlist
- friendlist_member
- group
- group_member
- link
- link_stat

- listing
- mailbox_folder
- message
- metrics
- note
- notification
- page
- page_admin
- page_fan
- permissions
- photo
- photo_tag
- profile
- standard_friend_info
- standard_user_info
- status
- stream
- stream_filter
- thread
- translation
- user
- video

- video_tag

Las tablas que utilizamos para cumplir con los requisitos de nuestra aplicación son: álbum y photo.

2.6.1 Tabla Álbum

La siguiente tabla posee todos los campos relacionados a un álbum en facebook

Indexable	Nombre	Tipo	Descripción
*	aid	<u>string</u>	El ID del álbum que se consulta. La aid no puede ser mayor de 50 caracteres. Nota: Debido a que el aid es una cadena, siempre se debe ajustar entre comillas cuando se hace referencia en una consulta. El aid es única sólo para un usuario determinado.
*	owner	int	El ID de usuario del propietario del álbum que se consulta.
*	cover_pid	<u>string</u>	El ID de la foto utilizada como portada para el álbum que se consulta. El cover_pid no puede ser mayor de 50 caracteres. Nota: Debido a que la cover_pid es una cadena, siempre se debe envolver la cover_pid entre comillas cuando se hace referencia en una consulta.
	name	<u>string</u>	El nombre del álbum que se consulta.
	created	time	La fecha que fue creado el álbum.
	modified	time	La última fecha de de actualización del álbum.
	description	String	La descripción del álbum que se consulta.
	location	string	La ubicación para el álbum que se consulta.
	size	int	El número de fotos en el álbum que se

			consulta.
	link	string	La dirección URL del álbum que se consulta.
	visible	string	Indica quienes pueden ver el álbum.
	modified_major	time	Indica el tiempo de una actualización importante (como la adición de fotos) la última actualización que se hizo en el álbum.
	edit_link	string	La URL de la edición del álbum.
	type	string	El tipo de álbum de fotos. Puede ser una de las siguientes: <ul style="list-style-type: none"> • profile : El álbum que contiene fotos de los perfiles • mobile : El álbum contiene fotos subidas de un móvil. • wall Wall El álbum que contiene fotos publicadas en una pared de usuario • normal Para todos los otros álbumes.
	object_id	int	Utilize Object_id para permitir que los usuarios dejen comentarios con el API

Tabla II.- Descripción de la tabla álbum de facebook

2.6.2 Tabla Photo

Indexable	Nombre	Tipo	Descripción
*	pid	string	El ID de la foto que se consulta. El pid no puede ser mayor de 50 caracteres. Nota: Debido a que el PID es una cadena, siempre se debe envolver el PID entre comillas cuando se hace referencia en una consulta. El PID es única sólo para un usuario determinado.
	aid	string	El ID del álbum que contiene la foto que se consulta. La aid no puede ser mayor de 50 caracteres. Nota: Debido a que el aid es una cadena, siempre se debe ajustar entre comillas cuando se hace referencia en una consulta. El aid es única sólo para un

			usuario determinado.
	owner	Int	El ID de usuario del propietario de la foto que se consulta.
	src_small	string	La dirección URL de la versión en miniatura de la foto que se consulta. La imagen puede tener un ancho máximo de 75px y una altura máxima de 225px. Esta URL puede estar en blanco.
	src_small_height	string	Altura de la versión en miniatura, en px. Este campo puede estar en blanco.
	src_small_width	string	Ancho de la versión en miniatura, en px. Este campo puede estar en blanco.
	src_big	string	La dirección URL de la versión en tamaño completo de la foto que se consulta. La imagen puede tener un ancho máximo o la altura de 604px. Esta URL puede estar en blanco.
	src_big_height	string	Altura de la versión en tamaño completo, en px. This field may be blank. Este campo puede estar en blanco.
	src_big_width	string	Anchura de la versión en tamaño completo, en px.. Este campo puede estar en blanco.
	src	string	La URL a la versión de vista del álbum de la foto que se consulta. La imagen puede tener un ancho máximo o la altura de 130px. Esta URL puede estar en blanco.
	src_height	string	Altura de la versión de vista de álbum, en px. Este campo puede estar en blanco.
	src_width	string	Ancho de la versión de vista de álbum, en px. Este campo puede estar en blanco.
	link	string	La dirección URL a la página que contiene la foto que se consulta.
	caption	string	El título de la foto que se consulta.
	Created	time	La fecha en que la foto fue añadida
	modified	time	La fecha en que se consulta la foto fue modificada por última vez.
	object_id	string	Utilice Object_id para permitir que los usuarios dejen comentarios con el API

Tabla III.- Descripción de la tabla foto de facebook

CAPITULO 3

3. Comunicación con los Servicios web de Hi5

3.1 Descripción de la red social [11]

Hi5 es una red social basada en un sitio web lanzado en 2003 y fundada por Ramun Yalamanchis, actual director general de la empresa hi5 Networks. El número de usuarios ha crecido rápidamente y al finalizar el año 2007 tenía más de 70 millones de usuarios registrados, la mayoría de ellas en América Latina; además, es uno de los 40 sitios web más visitados del mundo.

El sitio hi5 es famoso por su interactividad, pues hace de una simple cuenta de usuarios una especie de tarjeta de presentación virtual. Hi5 se encuentra presente en 23 idiomas.

3.2 Formas de comunicación [12]

Hi5 tiene 2 formas de acceder a sus servidores de servicio:

- SOAP: Se usa a través de un conjunto de WSI Basic Profile compatible con SOAP v1.1, además es compatible con XML-binary Optimized Packaging (XOP) y con el Mecanismo de Optimización de Transmisión de Mensajes para SOAP (MTOM), para la transmisión de datos binarios.
- REST: permite acceder a un conjunto de recursos a través de un conjunto fijo de operaciones.

Para acceder a la información de un usuario solo se utiliza su correo, pero su perfil debe ser público.

3.2.1 Principales espacios de nombres

Espacios de nombres SOAP	Espacios de nombres en REST	Descripción
AuthApiService	profile/lookup/{correo}	Permite autenticarse a un usuario para poder acceder a sus datos
ProfileApiService readProfileUser	/profile/user/{userId}	Retorna la información de perfil de un usuario
ProfileApiService readFoaf	/profile/foaf/{userId}	Retorna la lista de amigos de un usuario

readAlbumListFeed	/feed/album/{userId}	Retorna una los álbumes asociados a un usuario
ProfileApiService readPhotos	/feed/photos/user/{userId}/albumid/{albumId}	Retorna las fotos que tiene un album de un usuario

Tabla IV.- Principales espacios de nombres SOAP

3.3 Uso del protocolo

3.3.1 Requerimiento

Importar los álbumes, las fotos y la descripción, desde la cuenta de un usuario.

Requerimiento: Importar los álbumes, las fotos y el comentario de descripción, desde la cuenta de un usuario.



Figura 7.- Diagrama de comunicación REST con el servicio web de Hi5

Solución:

- La obtención del identificador de un usuario de hi5, dado el correo electrónico:
 - <http://api.hi5.com/rest/profile/lookup?email=correo>
- La obtención de la lista de álbumes, dado el identificador de usuario de hi5:
 - <http://api.hi5.com/rest/feed/albumlist/identificador>
de usuario

- La obtención de las fotos con su respectiva descripción, dado el identificador de usuario y el identificador del álbum:
 - <http://api.hi5.com/rest/feed/photos/user/identificador de usuario /albumid/identificador de álbum>

Para la comunicación con los Servicios web y el procesamiento del mensaje de respuesta, se utilizaron las siguientes librerías:

Librería	Versión
Activation	1.1.1
Jaxws-api	2.0
wSDL4J	1.5.2
commons-logging	1.1
commons-httpclient	3.0
commons-codec	1.3
jdom	1.0
Mail	1.4

Tabla V.- Librerías utilizadas para la comunicación con los servicios web de Hi5

CAPITULO 4

4. Comunicación con los servicios web de YouTube

4.1 Descripción de la Red Social [13]

YouTube es un sitio web de alojamiento de videos en el cual los usuarios pueden subir y compartir videos. Tres antiguos empleados de PayPal crearon YouTube en febrero de 2005. En noviembre de 2006, YouTube, LLC fue comprado por Google Inc. por 1.650 millones de dólares, y ahora opera como una filial de Google.

YouTube usa un reproductor en línea basado en Adobe Flash para servir su contenido. Es muy popular gracias a la posibilidad de alojar videos personales de manera sencilla. YouTube aloja una variedad

de clips de películas, programas de televisión, vídeos musicales, así como contenidos amateur como videoblogs (a pesar de las reglas de YouTube contra subir vídeos con copyright, este material existe en abundancia). Los enlaces a vídeos de YouTube pueden ser también puestos en blogs y sitios web personales usando APIs o incrustando cierto código HTML.

4.2 Redifusión web [14]

La sindicación web se refiere a lo que en español se denomina redifusión web (o redistribución) mediante la cual, parte de la información de una página web se pone a disposición de otros sitios web. Esto puede ser simplemente licenciando el contenido para que puedan usarlo otras personas; Sin embargo, en general, la redifusión web se refiere a ofrecer un contenido informativo desde una fuente web de un sitio, para proporcionar a otras personas la utilización del mismo (por ejemplo, noticias de un periódico, nuevos artículos en una bitácora, los últimos comentarios en un foro, etcétera).

Las fuentes suelen codificarse en XML, aunque el formato puede ser cualquier otro que pueda transportarse mediante HTTP, como son el HTML o el Java Script. Las dos principales familias de formatos de redifusión web son el RSS y el ATOM.

4.2.1 RSS [15]

RSS es una familia de formatos de fuentes web codificados en XML. Se utiliza para suministrar a suscriptores, información actualizada de manera frecuente. El formato permite distribuir contenido sin necesidad de un navegador, utilizando un software diseñado para leer estos contenidos RSS. A pesar de eso, es posible utilizar el mismo navegador para ver los contenidos RSS. Las últimas versiones de los principales navegadores permiten leer los RSS sin necesidad de software adicional. RSS es parte de la familia de los formatos XML desarrollado específicamente para todo tipo de sitios que se actualicen con frecuencia y por medio del cual se puede compartir la información y usarla en otros sitios web o programas. A esto se le conoce como redifusión web.

4.2.2 ATOM [16]

Es un fichero en formato XML similar al RSS, usado para la Redifusión Web. Se lo desarrolló para utilizarlo como estándar en la redifusión de contenidos web y, para que sea un formato de redifusión más flexible. No terminó con el problema, es más, actualmente conviven los tres formatos.

Tanto ATOM como RSS tienen el mismo objetivo que es permitir la distribución de contenidos y noticias de sitios web.

Un documento ATOM puede contener mayor cantidad de información, de forma más compleja y consistente que un documento RSS.

4.2.3 Diferencias entre RSS y ATOM

4.2.3.1 Especificaciones

ATOM tiene mayor facilidad para adaptarse a nuevos tipos de contenidos, gracias a una especificación más amplia.

- RSS 2.0 puede contener tanto texto plano como HTML, mientras que ATOM usa una etiqueta específica para este cometido. Esto hace que la forma que contenga mayor cantidad de tipos (HTML, XHTML, XML, Base64-encoded binary, y referencias a contenido externo como documentos, streams de video y audio, y los que se quieran agregar.
- RSS 2.0 tiene un campo "description" para meter el resumen o contenido. ATOM tiene campos separados.

4.2.3.2 Integración con XML

Según la especificación de ambos formatos, ATOM parece estar más integrado con XML que RSS:

- ATOM está definido dentro de un espacio de nombre de XML, mientras que RSS 2.0 no lo está.

- ATOM especifica el uso de XML intrínseco a `xml:base` para las URIs relativas. RSS 2.0 no entiende la diferencia entre las URIS relativas y las no relativas
- ATOM usa el atributo `xml:lang` intrínseco de XML mientras que RSS utiliza uno propio.
- En ATOM, es obligatorio que cada entrada tenga su propio identificador global único para facilitar las actualizaciones de las entradas, cosa que no pasa en RSS.

4.2.3.3 Estandarización

ATOM se estandarizó más porque:

- ATOM especifica que las fechas deben estar en el formato descrito en la RFC 3339 (un subconjunto de la ISO 8601). El formato de fecha en RSS 2.0 fue poco especificado y acepta cualquier formato utilizado en el medio.

- ATOM 1.0 tiene registrado un tipo MIME en IANA. Los feeds RSS 2.0 se envían como una application/rss+xml, aunque no están registrados como tipo MIME.
- ATOM 1.0 incluye un esquema XML. RSS 2.0 no.
- ATOM es un estándar abierto y evolucionable, que se ha desarrollado a través del proceso de estandarización IETF. RSS 2.0 no está estandarizado por ninguna organización autorizada. Además, de acuerdo con su copyright, no puede ser modificado.
- Los elementos de ATOM 1.0 pueden usarse como una extensión de otros vocabularios XML, incluido RSS 2.0
- ATOM 1.0 describe como las fuentes y entradas pueden ser firmados digitalmente usando la especificación XML Digital Signatures.

4.3 Detalle del API [18]

El API permite que un programa pueda realizar muchas de las operaciones disponibles en el sitio web de YouTube. Es posible buscar videos del usuario, y ver su contenido. Un programa también puede autenticar a un usuario para subir videos, modificar las listas de usuarios, y mucho más.

La API es principalmente para los desarrolladores que están acostumbrados a la programación en los lenguajes de lado del servidor. Es útil para los sitios o las aplicaciones que desean tener una mayor integración con YouTube. Esta integración puede ser una aplicación web que permite a los usuarios subir vídeos a YouTube. También podría ser un dispositivo o aplicación de escritorio que trae la experiencia de YouTube a una nueva plataforma. El API le da acceso a los vídeo e información del usuario almacenados en YouTube. Con esto, usted puede personalizar su sitio web o aplicación con la información existente de los usuarios, así como realizar acciones en su nombre (como comentarios sobre los vídeos y votar.)

Google Data APIs proporcionan un protocolo simple y estándar para la lectura y la escritura de datos en la web. Google Data APIs están basados en el ATOM 1.0 y RSS 2.0 formatos de sindicación, así como el ATOM Publishing Protocol.

4.4 Acerca de esta documentación [18]

Esta documentación se divide en las siguientes secciones:

- La sección de autenticación describe los dos diferentes métodos de autenticación disponibles para asociar las operaciones de la API con una cuenta de usuario. En esta sección también se describen las diferencias entre la autenticación de la API de datos de YouTube y otros servicios de Google Data APIs.

La autenticación permite a un usuario acceder a las funciones de YouTube que vincule el contenido o información de una cuenta de YouTube en particular. Estas características requieren que el usuario o la aplicación provean a YouTube el nombre de usuario y contraseña para esa cuenta.

Para autenticar correctamente, la solicitud deberá incluir el encabezado de autorización y cualquier X-GData-Key o parámetro clave. En el encabezado X-GData-key (o parámetro clave) se especifica la clave de desarrollador, un valor que identifica su solicitud(es). El encabezado de autorización especifica una señal para permitir a cada usuario (utilizando uno de los tres esquemas de autenticación - AuthSub, OAuth o ClientLogin que están disponibles) el acceso a la cuenta con sus características específicas. La siguiente lista te ayudará a seleccionar el esquema de autenticación que es apropiado para su aplicación

- La autenticación AuthSub proxy, permite a las aplicaciones web autenticar a los usuarios con sus cuentas de YouTube sin, tener que ingresar el nombre de usuario y la contraseña. Usted debe utilizar la autenticación de AuthSub si está construyendo una aplicación web que permitirá a los usuarios acceder a videos, comentar, calificar, contactar u otra información a sus propias cuentas de YouTube.
- El protocolo de OAuth proporciona una forma estándar de acceder a datos protegidos en diferentes sitios web. Considerando que los métodos AuthSub y ClientLogin

son específicos de Google, OAuth es un protocolo abierto que puede ser aplicado en otros sitios Web. Al igual que AuthSub, OAuth es útil si usted está construyendo una aplicación web que permitirá a los usuarios enlazar videos, comentar, calificar, contactar, etc. a sus propias cuentas de YouTube. OAuth puede ser particularmente atractiva para usted si su aplicación también se integra con otras API, además de la API de YouTube, y los API también soportan el protocolo OAuth.

- La autenticación de ClientLogin permite asociar todas las acciones de la API de YouTube con una cuenta única. En este caso, la aplicación le preguntará al usuario que introduzca su nombre de usuario y contraseña de YouTube para luego utilizar esos valores y solicitar un token de autenticación ClientLogin.
- El entendimiento de las fuentes de vídeos y la sección de entrada proporciona una muestra de la respuesta de la API y explica cómo extraer información acerca de un vídeo a partir de una lista de vídeos o un conjunto de resultados de búsqueda. También se muestra cómo los elementos XML, en una respuesta de datos de API de YouTube, corresponden a

la información de vídeo que normalmente se muestra en el sitio web de YouTube.

Al recuperar una fuente de vídeo o la lista de resultados de búsqueda, YouTube devuelve un ATOM feed <entry> que contiene una etiqueta para cada vídeo en el conjunto de resultados. La raíz de etiquetas XML en la respuesta será la <feed> etiqueta. El ATOM feed también contiene el número total de resultados en la lista, el índice del primer elemento de la lista, el número de elementos de la lista y otros metadatos acerca del feed.

Requerimiento: Autenticación de un usuario y visualización de sus videos

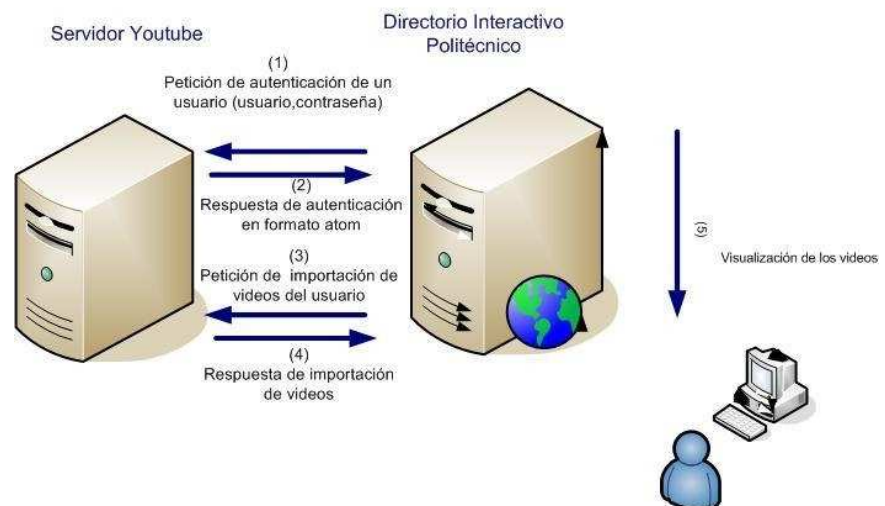


Figura 8.- Autenticación de un usuario de YouTube

4.5 Versión de la API

En la solicitud al API, puede especificarse la versión de la API de YouTube que se utilizará para manejar la petición. Su solicitud puede especificar una versión de la API usando el parámetro `v` o la Versión GData del encabezado de la petición HTTP. Si en la solicitud no se especifica una versión de la API, a continuación, YouTube usará la versión más antigua compatible de la API para manejar esa solicitud.

Tenga en cuenta las siguientes características de los números de versión de la API:

- YouTube puede publicar actualizaciones para una versión específica del API para la cual a la liberación no se le asigna un nuevo número de versión. Estas actualizaciones compatibles con versiones anteriores pueden incluir características opcionales de la API, correcciones de errores o de ambos.

Un incremento del número de versión de la API expresa que contiene cambios que son incompatibles con versiones anteriores de la API.

Para la comunicación con YouTube y el procesamiento de los videos se utilizaron las siguientes librerías:

Librería	Versión
gdata-youtube	2.0
gdata- youtube-meta	2.0
gdata-core	1.0
gdata-media	1.0

Tabla VI.- Librerías utilizadas para la comunicación con los servicios web de Youtube

CAPÍTULO 5

5. Implementación - Pruebas - Manual de usuario

5.1 Diagrama de interacción de los módulos

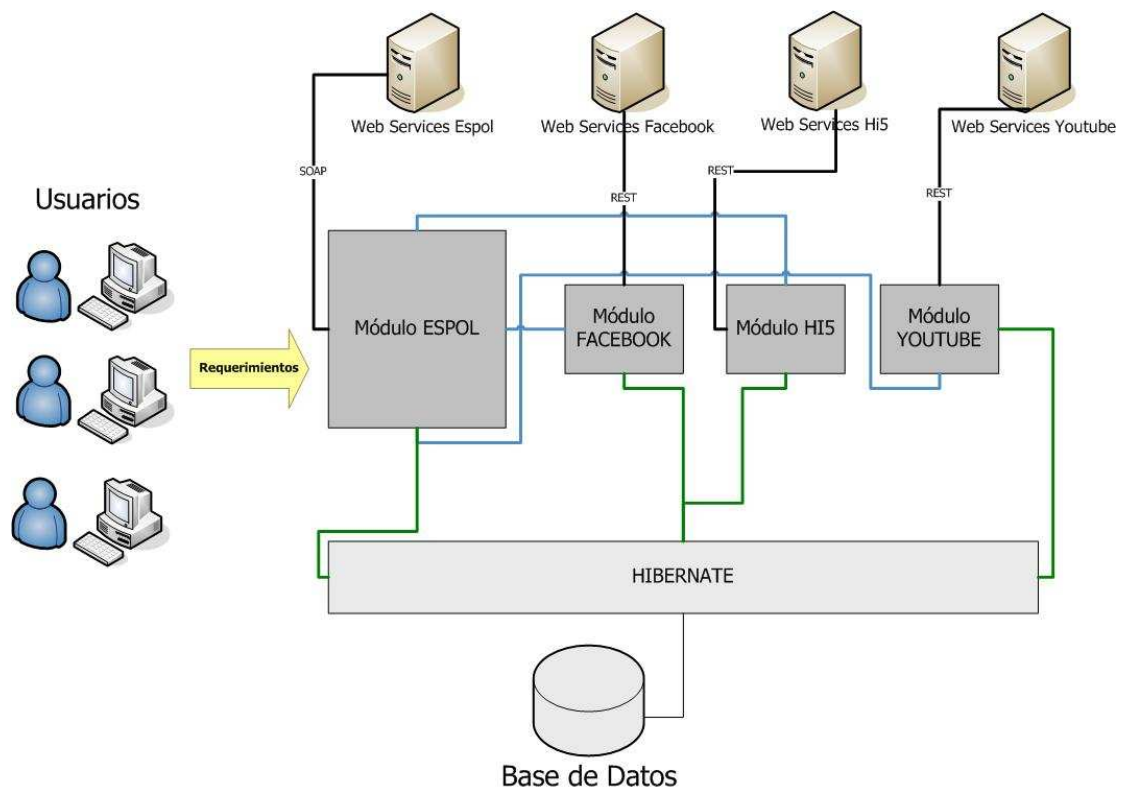


Figura 9.- Diagrama de Interacción de módulos

5.2 Descripción de los módulos

5.2.1 Módulo ESPOL

5.2.1.1 Requisitos del módulo ESPOL

Requisito	Descripción
Ingreso al sistema	Permite validar que solo puedan acceder estudiantes politécnicos a la aplicación. Si es la primera vez que un usuario ingresa con sus datos, se valida en el web services ESPOL, caso contrario se lo hace en la base de datos de nuestra aplicación.
Búsqueda de usuarios	Permite consultar por nombres y apellidos, para luego listar los usuarios y da la opción de visualizar su perfil

Tabla VII.- Requisitos que se implementan en el modulo ESPOL

5.2.1.2 Archivos fuentes que contiene el módulo ESPOL

Vistas	<ul style="list-style-type: none"> ▪ index ▪ principal ▪ buscar_personas ▪ perfil_otro_usuario ▪ fotos_album ▪ fotos_album_usuario_encontrado
Controlador	<ul style="list-style-type: none"> ▪ ServletEspol
Modelos	<ul style="list-style-type: none"> ▪ ConexiónWebServicesSoap ▪ DynamicInvoker
Librerías	<ul style="list-style-type: none"> ▪ Axis 1.0 ▪ Saaj 1.2 ▪ Axis-ant 1.4 ▪ commons-discovery 0.2 ▪ Jaxrpc 1.1 ▪ log4j-1.2.8 1.2.8 ▪ Mail 1.3.3

Tabla VIII.- Conjunto de archivos fuentes que contiene el módulo ESPOL

5.2.2 Módulo Facebook

5.2.2.1 Requisitos del módulo Facebook

Requisito	Descripción
Autenticación con la red social	El usuario ingresa su correo y contraseña para validar si se encuentra registrado en esta red social.
Consulta de todos los álbumes que posee un usuario	Se consulta la foto de perfil de cada álbum, para que el usuario elija solo aquellos que tendrán acceso público para los demás usuarios de la aplicación.
Almacenamiento de solo aquellos álbumes que un usuario desee que tengan acceso público	El almacenamiento no es físico, solo se guarda ciertos valores, tales como: El identificador del usuario, identificador del álbum, identificador de la foto, entre otros.
Consulta de toda la información concerniente a un álbum	Las fotos y los comentarios de un álbum se consultan en tiempo real al servidor de web services de ésta red.

Tabla IX.- Requisitos que se implementan en el módulo Facebook

5.2.2.2 Archivos fuentes que contiene el módulo Facebook

Vistas	<ul style="list-style-type: none"> ▪ configurar_facebook
Controlador	<ul style="list-style-type: none"> ▪ ServletFacebook
Modelos	
Librerías	<ul style="list-style-type: none"> ▪ JavaScript Client Library

Tabla X.- Conjunto de archivos fuentes que contiene el módulo Facebook

5.2.3 Módulo Hi5

5.2.3.1 Requisitos del módulo Hi5

Requisito	Descripción
Autenticación con la red social	El usuario ingresa solamente su correo y el web services de ésta red social valida su existencia siempre
Consulta de todos los álbumes que posee un usuario	Solo se podrán acceder a los álbumes de un usuario que tenga su perfil como público.
Almacenamiento de solo aquellos álbumes que un usuario desee que tengan acceso público	El almacenamiento no es físico, solo se guarda ciertos valores, tales como: El identificador del usuario, identificador del álbum, descripción del

	álbum, entre otros.
Consulta de toda la información concerniente a un álbum	Las fotos y los comentarios de un álbum se consultan en tiempo real al servidor de web services de ésta red.

Tabla XI.- Requisitos que se implementan en el módulo Hi5

5.2.3.2 Archivos fuentes que contiene el módulo Hi5

Vistas	<ul style="list-style-type: none"> ▪ configurar_hi5
Controlador	<ul style="list-style-type: none"> ▪ ServletHi5
Modelos	<ul style="list-style-type: none"> ▪ LeerAlbum ▪ LeerAlbumes ▪ LeerFotosAlbumes ▪ LeerIdHi5
Librerías	<ul style="list-style-type: none"> ▪ Activation 1.1.1 ▪ Jaxws-api 2.0 ▪ wsdl4j 1.5.2 ▪ commons-logging 1.1 ▪ commons-httpclient 1.1

Tabla XII.- Conjunto de archivos fuentes que contiene el módulo Hi5

5.2.4 Módulo YouTube

5.2.4.1 Requisitos del módulo YouTube

Requisito	Descripción
Autenticación con la red social	El usuario ingresa su correo y contraseña que posee como datos de sesión para validar si pertenece o no a ésta red social
Consulta de todos los videos que posee un usuario	Se visualizaran todos los videos que tiene el usuario esta red, para que luego elija aquellos que tendrán acceso público.
Almacenamiento de solo aquellos videos que un usuario desee que tengan acceso público.	El almacenamiento no es físico, solo se guarda ciertos valores, tales como: El identificador del usuario, identificador del video, titulo del video, entre otros.
Consulta de los videos de acceso público	Si un usuario desea visualizar un video, debe darle click en el icono de "play" y automáticamente se reproduce.

Tabla XIII.- Requisitos que se implementan en el módulo Youtube

5.2.4.2 Archivos fuentes que contiene el módulo Youtube

Vistas	<ul style="list-style-type: none"> ▪ configurar_youtube
Controlador	<ul style="list-style-type: none"> ▪ ServletYoutube
Modelos	<ul style="list-style-type: none"> ▪ SimpleCommandLineParser ▪ YouTubeWriteClient
Librerías	<ul style="list-style-type: none"> ▪ gdata-youtube 2.0 ▪ gdata- youtube-meta 2.0 ▪ gdata-core 1.0 ▪ gdata-media 1.0

Tabla XIV.- Conjunto de archivos fuentes que contiene el módulo Youtube

5.3 Pruebas de usabilidad

La aplicación posee 3 versiones. Para las pruebas que se detallan a continuación, se uso las versiones 1 y 2.

5.3.1 Pruebas sobre la versión 1

En esta versión solo tiene implementado los módulos de Hi5 y Youtube

Nombre de Usuario	Homero Tumbaco
Módulo	Hi5
Requisito a evaluar	Consulta de todos los álbumes que posee un usuario
Observaciones	<ul style="list-style-type: none"> ▪ No pide la clave de acceso al perfil

	<ul style="list-style-type: none"> ▪ El perfil debe estar configurado como público ▪ Debe existir alguna diferencia entre los álbumes que en la última configuración los publique, y los que no.
Análisis de observaciones	<p>Con respecto a la 1era observación:</p> <ul style="list-style-type: none"> ▪ La forma utilizada para comunicarnos, no exige la autenticación del usuario para poder extraer su información, pero si obliga al usuario a tener configurado su perfil como "Público". <p>Con respecto a la 2da observación:</p> <ul style="list-style-type: none"> ▪ Debido a que accedemos a la información de los usuarios mediante un usuario externo(no dueño de perfil), la API solo permite consultar la información de aquellos usuarios con perfil público <p>Con respecto a la 3era observación:</p> <ul style="list-style-type: none"> ▪ Es una observación válida que originó un cambio en el desarrollo de la aplicación. Este cambio se visualiza en la versión 2
Cambios implementados después de realizar ésta prueba	1

Tabla XV.- Prueba de Usabilidad 1 de la versión 1

Nombre de Usuario	Pedro Torres
Módulo	Youtube
Requisito a evaluar	Consulta de todos los videos que posee un usuario

Observaciones	<ul style="list-style-type: none"> ▪ La aplicación debe re direccionar automáticamente al perfil del usuario, después de guardar los videos que se configuraron como públicos
Análisis de observaciones	<p>Con respecto a la 1era observación:</p> <ul style="list-style-type: none"> ▪ Es una observación válida que originó un cambio en el desarrollo de la aplicación. Este cambio se visualiza en la versión 2
Cambios implementados después de realizar ésta prueba	1

Tabla XVI.- Prueba de Usabilidad 2 de la versión 1

Nombre de Usuario	Luis Rivas
Módulo	Aplicación en general
Requisito a evaluar	Todos
Observaciones	<ul style="list-style-type: none"> ▪ Debe aparecer el número de fotos que tiene un álbum ▪ No se puede intuir si se puede ampliar una foto en un álbum
Análisis de observaciones	<p>Con respecto a la 1era observación:</p> <ul style="list-style-type: none"> ▪ Es una observación válida que originó un cambio en el desarrollo de la aplicación. Este cambio se visualiza en la versión 2 <p>Con respecto a la 2da observación:</p> <ul style="list-style-type: none"> ▪ Es una observación válida que originó un cambio en el desarrollo de la aplicación. Este cambio se visualiza en la versión 2 y se basa en colocarle un borde y ampliar un poco la

	imagen para que el usuario esté incentivado a dar clic
Cambios implementados después de realizar ésta prueba	2

Tabla XVII.- Prueba de Usabilidad 3 de la versión 1

5.3.2 Pruebas sobre la versión 2

La versión 2 implementa los cambios que surgieron después de realizar las pruebas a la versión 1 de la aplicación. Además, implementa los módulos de “Facebook” y “ESPOL”

Nombre de Usuario	Fernando Loor
Módulo	ESPOL
Requisito a evaluar	Búsqueda de usuarios
Observaciones	<ul style="list-style-type: none"> ▪ Debe existir alguna forma de distinguir el perfil de usuario encontrado, con el perfil del usuario que ingreso con sus datos de cuenta ESPOL. No es suficiente con el mensaje “Perfil de x persona”
Análisis de observaciones	<p>Con respecto a la 1era observación:</p> <ul style="list-style-type: none"> ▪ Es una observación válida que originó un cambio en el desarrollo de la aplicación. Este cambio se visualiza en la versión 3 y se basa en colocarle el color “café” al perfil del usuario encontrado. ▪
Cambios implementados después de realizar ésta prueba	1

Tabla XVIII.- Prueba de Usabilidad 1 de la versión 2

Nombre de Usuario	Arturo Hernández
Módulo	Facebook
Requisito a evaluar	Consulta de todos los álbumes que posee un usuario
Observaciones	<ul style="list-style-type: none"> ▪ ¿ Por qué aparece mensaje “You are logging out of both this site and facebook”?
Análisis de observaciones	<p>Con respecto a la 1era observación:</p> <ul style="list-style-type: none"> ▪ Este mensaje aparece porque se cierra sesión después de consultar los álbumes. El motivo de cerrar sesión es por seguridad del usuario, para que ninguna persona puede alterar algún perfil ajeno, y como consecuencia originar una molestia hacia el uso de nuestra aplicación. ▪
Cambios implementados después de realizar ésta prueba	0

Tabla XIX.- Prueba de Usabilidad 2 de la versión 2

5.4 Manual de usuario

El siguiente manual de usuario toma en cuenta la última versión de la aplicación (Versión 3)

5.4.1 Ingresar al “Directorio Interactivo de la ESPOL”

Para ingresar al Sistema, se digita el usuario y contraseña de la cuenta ESPOL. Luego presionamos el botón “Iniciar Sesión”.

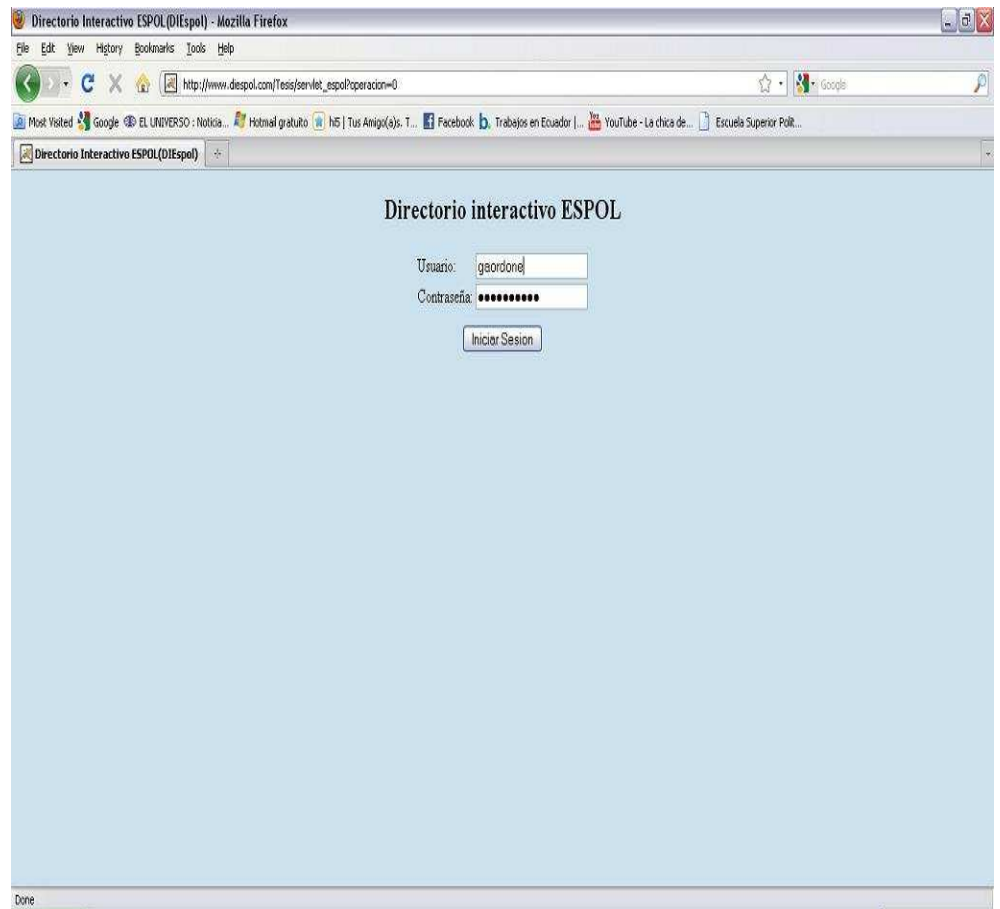


Figura 10.- Pantalla de ingreso al sistema

5.4.2 Primera pantalla de ingreso al Sistema

Si es la primera vez que ingresamos al sistema, solo se mostrará la información que exista en las bases de datos de la ESPOL.

Para importar los álbumes de Facebook y Hi5, además de los videos de Youtube se debe configurar los datos de la cuenta que posee el usuario en éstas redes sociales.

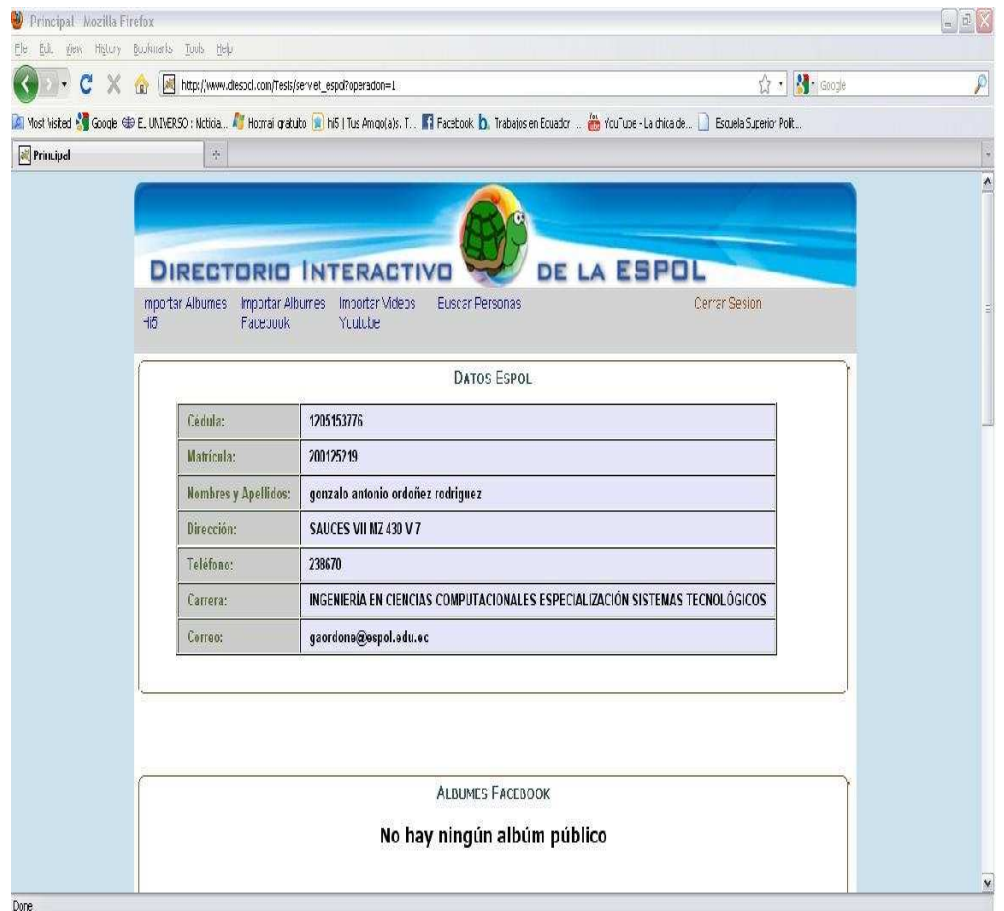


Figura 11.- Primera pantalla de ingreso al Sistema

5.4.3 Importar Álbumes de Facebook

Para importar álbumes de Facebook damos clic en “Importar Álbumes de Facebook”



Figura 12.- Importar álbumes de Facebook

En la siguiente pantalla se debe dar clic en “Connect” e, iniciar sesión con el correo y contraseña de la cuenta virtual de ésta red social.



Figura 13.- Conectar a la aplicación de Facebook

Nos aparecerá una ventana de Facebook que indica que la aplicación “Directorio Interactivo de la ESPOL”, tendrá acceso a nuestros álbumes para importarlos.

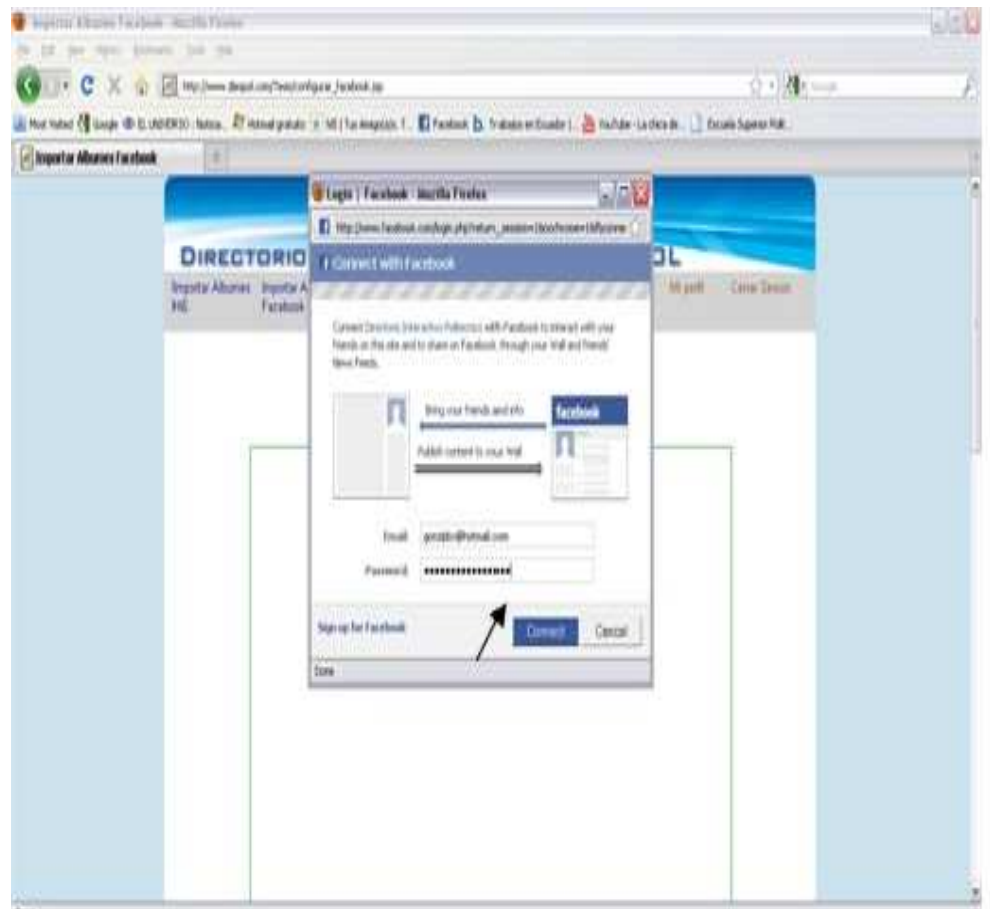


Figura 14.- Ventana de autenticación con Facebook

Una vez conectados, se visualizarán todos los álbumes que tiene el usuario. Procedemos a elegir aquellos álbumes que deseamos que sean públicos para todos los usuarios del “Directorio Interactivo de la ESPOL” y, presionamos el botón Guardar



Figura 15.- Álbumes del usuario en Facebook

Luego nos re-direccionará a la ventana principal, donde se visualizarán los álbumes públicos (seleccionados anteriormente) y que podrán ser vistos por los demás usuarios.

The screenshot shows a Mozilla Firefox browser window displaying the 'DIRECTORIO INTERACTIVO DE LA ESPOL' website. The page features a navigation menu with options like 'Importar Álbumes Hi5', 'Importar Álbumes Facebook', 'Importar Vídeos Youtube', 'Buscar Personas', and 'Cerrar Sesión'. Below the navigation, there is a section titled 'DATOS ESPOL' containing a table with the following information:

Cédula:	1205153776
Matrícula:	200125219
Nombres y Apellidos:	gonzalo antonio ordoñez rodríguez
Dirección:	SAUCES VII MZ 430 V 7
Teléfono:	238670
Carrera:	INGENIERÍA EN CIENCIAS COMPUTACIONALES ESPECIALIZACIÓN SISTEMAS TECNOLÓGICOS
Correo:	gaordone@espol.edu.ec

Below the data table, there is a section titled 'ÁLBUMES FACEBOOK' with navigation links: 'Primero | Anterior | Siguiente | Último'. This section displays four small thumbnail images of people.

Figura 16.- Álbumes de facebook con acceso público, de un usuario

5.4.4 Importar Álbumes de Hi5

Para importar álbumes de Hi5 damos clic en “Importar Álbumes de Hi5”



Figura 17.- Importar álbumes de Hi5

Para importar los álbumes de Hi5 no es necesario iniciar sesión, basta con solo ingresar el correo electrónico con el que tenemos configurada nuestra cuenta, pero si es necesario que el perfil del usuario sea público. Luego presionamos el botón “Importar Álbumes”.



Figura 18.- Conexión con los servidores de Hi5

Una vez conectados, se visualizarán todos los álbumes que tiene el usuario. Procedemos a elegir aquellos álbumes que deseamos que sean públicos para todos los usuarios del “Directorio Interactivo de la ESPOL” y, presionamos el botón Guardar.



Figura 19.- Álbumes del usuario en Hi5

Luego nos re-direccionará a la ventana principal, donde se visualizarán los álbumes públicos (seleccionados anteriormente) y que podrán ser vistos por los demás usuarios.

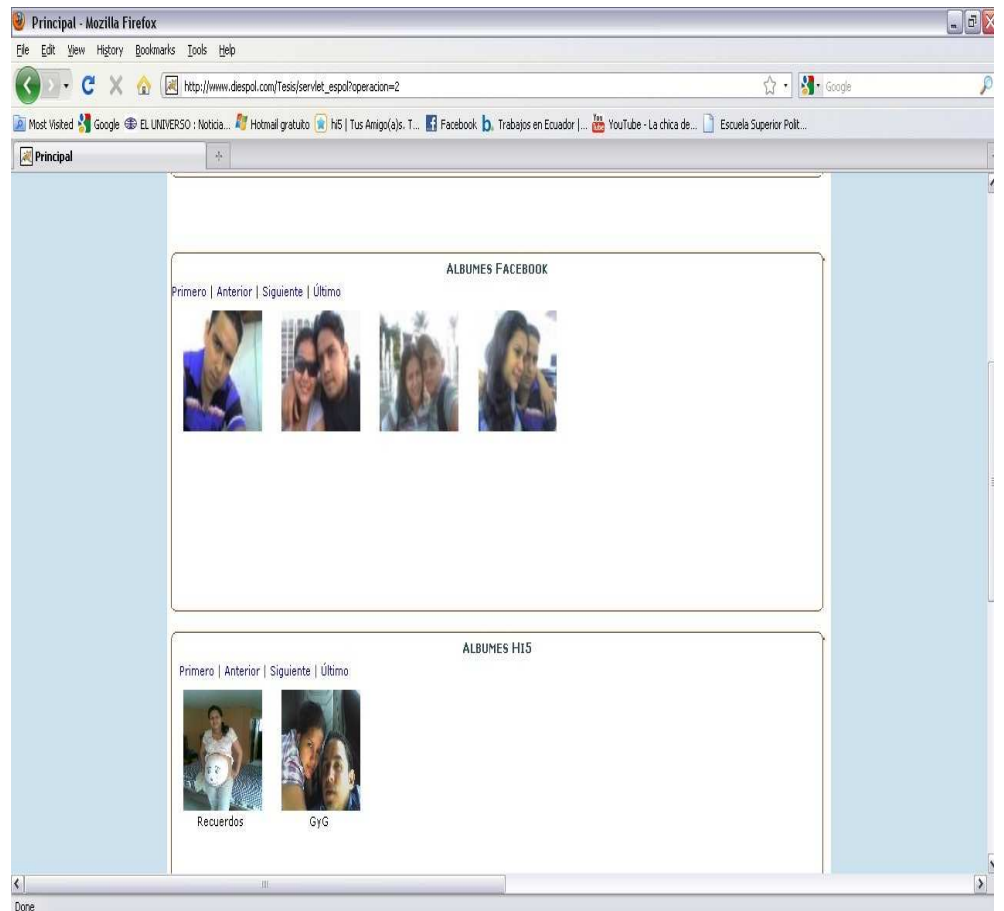


Figura 20.- Álbumes de Hi5 con acceso público, de un usuario

5.4.5 Importar Videos de Youtube

Para importar los videos de Youtube damos clic en “Importar Videos Youtube”.



Figura 21.- Importar videos de YouTube

Iniciamos sesión con el correo electrónico y contraseña, con la que tenemos configurado nuestra cuenta de Youtube.



Figura 22.- Conexión con los servidores de YouTube

Una vez conectados, se visualizarán todos los videos que tiene el usuario. Procedemos a elegir aquellos videos que deseamos que sean públicos para todos los usuarios del "Directorio Interactivo de la ESPOL" y, presionamos el botón Guardar.

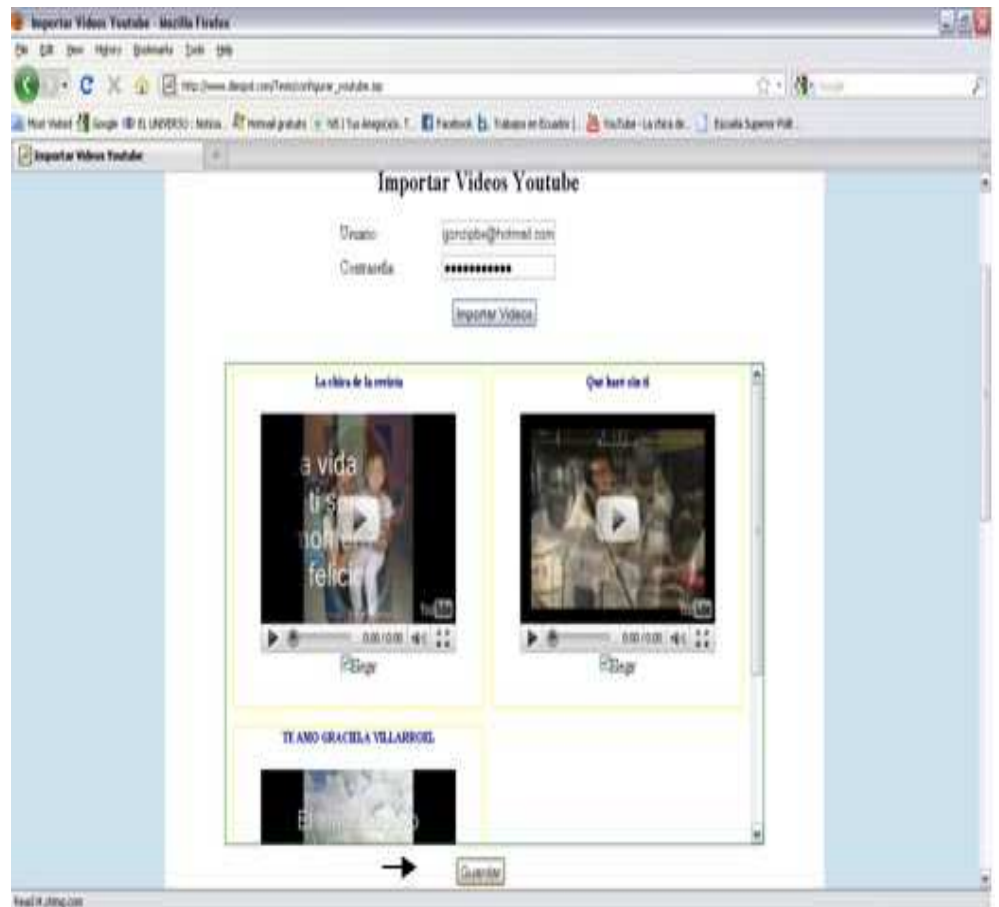


Figura 23.- Videos del usuario en YouTube

Luego nos re-direccionará a la ventana principal, donde se visualizarán los videos públicos (seleccionados anteriormente) y que podrán ser vistos por los demás usuarios.

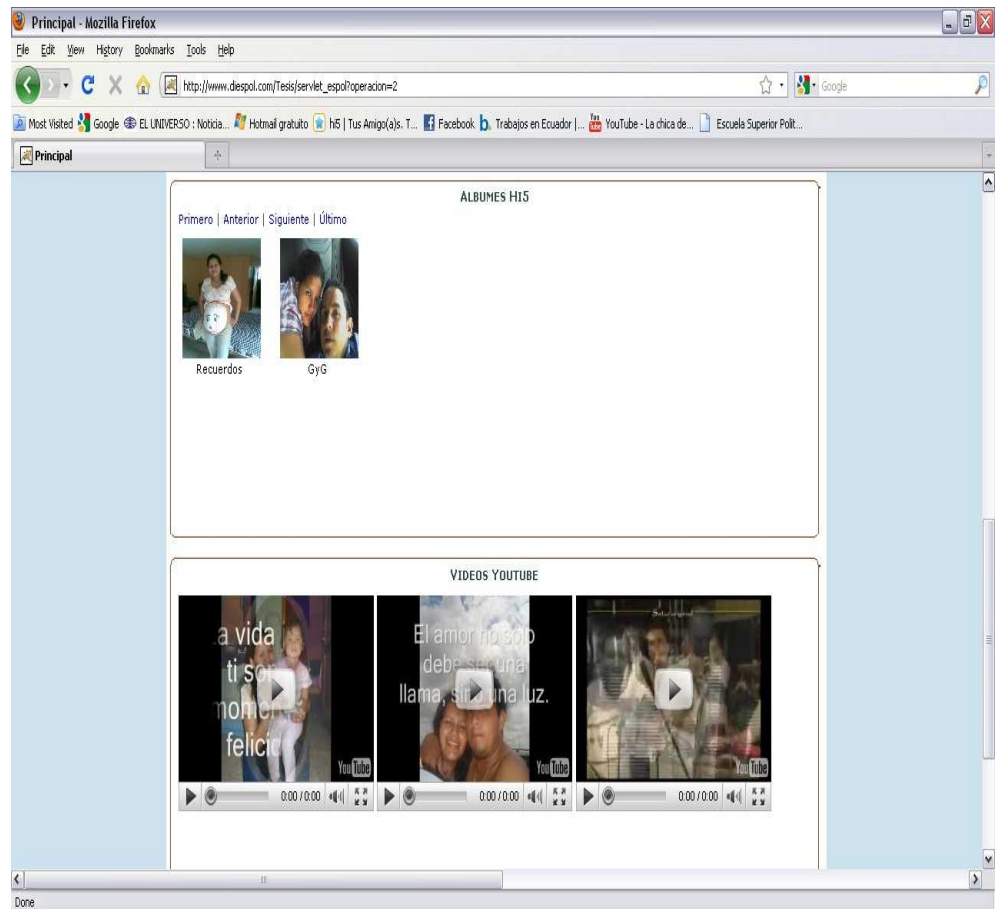


Figura 24.- Videos de Youtube con acceso público, de un usuario

5.4.6 Buscar Personas

Para buscar a los demás usuarios (estudiantes politécnicos), damos clic en “Buscar Personas”.

The screenshot shows a web browser window displaying the 'DIRECTORIO INTERACTIVO DE LA ESPOL' website. The page features a navigation menu with options like 'Ingestar Albumes HC', 'Ingestar Albumes Facebook', 'Ingestar Videos Youtube', and 'Busca Personas'. Below the menu is a table titled 'DATOS ESPOL' containing contact information for Rosca Zucaron. At the bottom, there is a section for 'ALBUMES FACEBOOK' with navigation links and a grid of album thumbnails.

DATOS ESPOL	
Código	120515376
Móvil	200 025218
Redes sociales y perfiles	google+ twitter instagram
Dirección	SAUCES VIMAC CNO V7
Teléfono	238470
Carrera	INGENIERÍA EN CIENCIAS COMPUTACIONALES ESPECIALIZACIÓN SISTEMAS TECNOLÓGICOS
Correo	guzardo@espol.edu.ec

Figura 25.- Buscador de personas en nuestra aplicación

Luego procedemos a ingresar el nombre y el apellido del usuario que se está buscando. También se puede ingresar solamente el nombre o el apellido. Luego se visualizarán a todos los usuarios que coincidan con los parámetros ingresados.



Figura 26.- Ingreso de parámetros de búsqueda

Para ver el perfil del usuario que buscamos, presionamos el enlace “ver perfil”. Se visualizarán todos los álbumes y videos que el usuario dueño de perfil configuró como públicos

La siguiente figura muestra los datos de la ESPOL y los álbumes Facebook que tiene un determinado usuario de nuestra aplicación


Perfil de stalin alberto arroyabe merchan [Volver a Mi perfil](#)

DATOS ESPOL

Cédula:	0923463624
Matrícula:	200405967
Nombres y Apellidos:	stalin alberto
Dirección:	GUAYACANES MZ 72 VILLA 45
Teléfono:	2820706
Carrera:	INGENIERÍA EN CIENCIAS COMPUTACIONALES ESPECIALIZACIÓN SISTEMAS DE INFORMACIÓN
Correo:	sarroyab@espol.edu.ec

ALBUMES FACEBOOK

Primero | Anterior | Siguiente | Último



Viaje a esmeraldas con mi familia y panas

Transferring data from i2.yimg.com...

Figura 27.- Perfil del usuario encontrado (primera parte)

Esta figura nos muestra los álbumes de Hi5 y los videos de Youtube que tiene el mismo usuario.

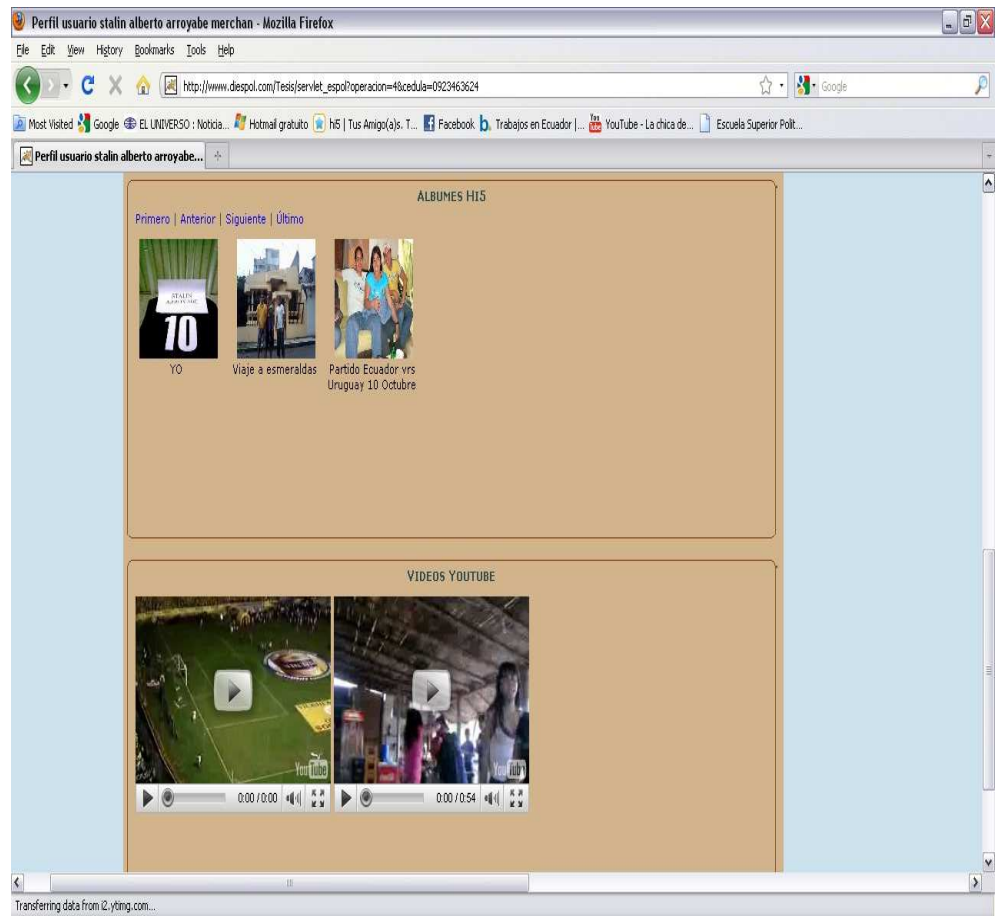


Figura 28.- Perfil del usuario encontrado (segunda parte)

Conclusiones y recomendaciones

Conclusiones.-

1. REST depende directamente del protocolo HTTP (Hypertext Transfer Protocol) en un sentido de petición-respuesta. Utiliza un conjunto bien definido de simples y pequeñas operaciones, tales como: POST, GET, PUT y DELETE.
2. SOAP siempre lleva asociado la definición de su servicio web, que se encuentra en un archivo WSDL.
3. REST está compuesta por una serie de principios arquitectónicos y limitaciones. Si una aplicación o diseño cumple con esos principios y limitaciones, se considera RESTful.
4. FQL es un lenguaje muy parecido a SQL. Debe permitir realizar JOIN, para obtener mejores resultados al momento que se realizan las consultas.
5. Es favorable que se sigan definiendo y que se respeten, los estándares de desarrollo de aplicaciones web, porque facilitan el aprendizaje, programación y comunicación entre distintas aplicaciones.
6. El API de la red social Hi5 es inestable y se debe revisar paulatinamente los cambios en sus espacios de nombres.

7. Facebook lidera el mercado de las redes sociales, porque su aplicación posee un alto grado de usabilidad, además su documentación es clara, y su API puede ser accedida por diversos lenguajes de programación (javascript, php, java, perl, python, c# entre otros).
8. El lenguaje FQL permite expandir el uso del API, debido a que no limita al desarrollador a consumir métodos predefinidos.
9. La finalidad de que una red social publique su API, es para generar un mayor tráfico hacia su dominio, y aprovechar el interés de desarrolladores en agregar valor al sitio, por medio de terceras aplicaciones.
10. La Red Social Facebook debe permitir una mejor forma de autenticación, con la finalidad de no tener que alquilar un hosting. Esto ayudaría a incrementar el uso del API por parte de los desarrolladores.
11. Code Google, nos da la facilidad de interactuar con muchas redes sociales por medio del lenguaje java. Los desarrolladores lo preferimos porque permite descargar librerías, que están bien documentadas y son muy fáciles de usar.

Recomendaciones.-

1. Para el correcto funcionamiento del “Directorio Interactivo de la ESPOL” se recomienda tener un hosting java con excelente características, debido a que se realiza consultas a Servicios Web, llamada a procedimientos remotos mediante APIs, y se estima que la cantidad de usuarios en línea sea muy significativa.
2. Aprender correctamente los distintos protocolos de comunicación (SOAP, REST, RPC, etc.), antes de desarrollar un una aplicación web, con el fin de obtener el mayor beneficio posible
3. Respetar los estándares de desarrollo web, para obtener un mejor posicionamiento, mayor velocidad de renderización, escalabilidad, y mantenimiento.
4. Planificar las actividades a realizar, establecer los recursos y tiempos de entregas en los avances, para evitar incumplimientos.
5. Utilizar mecanismos que permitan que la mayor cantidad de procesamiento recaiga en el cliente para que el servidor pueda atender una mayor cantidad de peticiones.
6. Leer, entender y aprender a interactuar con los APIs de cualquiera red social, con la finalidad de obtener los resultados esperados de acuerdo los requerimientos establecidos.

Referencias Bibliográficas

(1)Wikipedia, Servicios Web, http://es.wikipedia.org/wiki/Servicios_Web, última visita 5 Enero 2010.

(2)W3C, SOAP Versión 1.2, <http://www.w3c.es/Traducciones/es/TR/2003/REC-soap12-part0-20030624/>, última visita 8 Enero 2010.

(3)W3C, Lenguaje descriptor de servicios web WSDL, <http://www.w3.org/TR/wsdl>, última visita 15 Enero 2010.

(4)ESPOL, [webService](https://www.academico.espol.edu.ec/services/directorioEspol.aspx) directorioEspol, <https://www.academico.espol.edu.ec/services/directorioEspol.aspx>, última visita 15 Enero 2010.

(5)ESPOL, [webService](https://www.academico.espol.edu.ec/Services/wsSAAC.aspx) wsSAAC, <https://www.academico.espol.edu.ec/Services/wsSAAC.aspx> , última visita 15 Enero 2010.

(6)Wikipedia, La red social y su impacto en la web, <http://es.wikipedia.org/wiki/Facebook>, última visita 16 Enero 2010

(7) Bruce Sun (IBM), Protocolo de comunicación REST, <http://www.ibm.com/developerworks/ssa/library/wa-aj-multitier/index.html> , última visita 15 Enero 2010.

(8) Wikipedia, Forma de comunicación con la red, http://wiki.developers.facebook.com/index.php/Main_Page, última visita 16 Enero 2010.

(9) Facebook, Facebook Query Language (FQL), <http://wiki.developers.facebook.com/index.php/FQL>, última visita 16 Enero 2010.

(10) Facebook, Tablas FQL, http://wiki.developers.facebook.com/index.php/FQL_Tables, última visita 16 Enero 2010.

(11) Wikipedia, Descripción del sitio Hi5, <http://es.wikipedia.org/wiki/Hi5>, última visita 15 Enero 2010.

(12) Hi5, API del sitio Hi5, <http://api.hi5.com/>, última visita 15 Enero 2010.

(13) Wikipedia, Descripción de la Red Social, <http://es.wikipedia.org/wiki/YouTube> , última visita 30 de octubre del 2009

(14)Wikipedia, Redifusión web, http://es.wikipedia.org/wiki/Redifusi%C3%B3n_web,
última visita 30 de octubre del 2009

(15)Wikipedia, RSS, <http://es.wikipedia.org/wiki/RSS> , última visita 30 de octubre del
2009

(16)Wikipedia, Formato de Redifusión ATOM,
http://es.wikipedia.org/wiki/Atom_%28formato_de_redifusi%C3%B3n%29 , última
visita 30 de octubre del 2009

(17)Google, Detalle del api,
http://code.google.com/intl/es/apis/youtube/2.0/developers_guide_protocol_audience.html, última visita 21 de Junio del 2010

(18)Google, Acerca de esta documentación,
http://code.google.com/intl/es/apis/youtube/2.0/developers_guide_protocol_audience.html, última visita 21 de Junio del 2010