

Prototipo de una Red Social utilizando Micro-formatos y Ontologías para Virtual Learning & Business Solutions

Alejandro Villamarín, Margarita Zambrano, Carlos Prócel

Departamento de Ciencias de la Computación, Universidad de las Fuerzas Armadas "ESPE", Sangolquí, Ecuador,

favm@email.com, mezambrano@espe.edu.ec, ctprocel@espe.edu.ec

RESUMEN

El presente proyecto muestra el desarrollo de un caso de estudio de Web Semántica llamado SemanticKipu. Para llevarlo a cabo se implementó un sitio Web para soluciones de negocios y aprendizaje virtual (VLBS) en que los usuarios registrados puedan disponer de una página de perfil personal la cual en su marcado HTML contenga micro-formatos embebidos tales como RDF y ontologías FOAF. Se siguió las fases de la metodología OOHDM que permitió demostrar el funcionamiento de la Web Semántica y sus agentes. Los resultados muestran que los usuarios pueden obtener RDFs los cuales contienen información personal y red de amigos de cada uno de sus integrantes.

Palabras Clave: Web Semántica, RDF, ontologías, Micro-formatos.

ABSTRACT

This project shows the development of a case study of Semantic Web, called Semantic-Kipu. To carry out this work, it has implemented a Website for business solutions and e-learning (VLBS) in which registered users to have a personal profile page in their HTML which contains embedded micro-formats such as RDF and ontologies FOAF. It used the phases of the OOHDM methodology allowed to demonstrate the operation of the Semantic Web and its agents. The result shows that users can obtain RDFs which contain personal information and network of friends each of its members.

Keywords: Semantic Web, RDF, ontologies, Micro-formatos.

1. INTRODUCCIÓN

El proyecto Web Semántica⁵, es impulsado por la World Wide Web Consortium (W3C), que ha creado un estándar universal para el intercambio de información, dotando de semántica al contenido de los documentos Web, de manera que este contenido sea comprensible por computadores. Mientras que los humanos pueden leer y entender el contenido de una páginas Web, Berners-Lee imaginó una Web compuesta de páginas Web que conteniendo nuevos formatos pudieran ser entendidas, combinadas y analizadas por computadores, con el objetivo de que humanos y computadores pudieran cooperar entre sí, de la misma manera en que humanos cooperan entre sí.

La idea central tras la Web Semántica es la de trasladar la lógica que reside en las aplicaciones a los datos propiamente. La clave para esto es convertir los datos en datos inteligentes. Cabe recalcar que el inventor de la Web Tim Berners - Lee no concuerda con la idea de versionar a la Web (Web 1.0, Web 2.0, Web 3.0), más bien su visión sobre el futuro de la Web es la universalización de la Web Semántica. Es decir la Web está evolucionando hacia una Web Semántica, de manera que cada vez se alinea más con su idea original, la cual fue descrita más de 20 años atrás [1].

Por otra parte, Nova Spivak, un visionario en el tema de la Web y tecnología en general, describió la evolución de la Web desde sus principios hasta la era de la Web Semántica (Web 3.0, para algunos autores), basado en dos aspectos fundamentales: la riqueza en las conexiones de datos y la riqueza en las conexiones sociales. De esta manera se puede evidenciar la evolución de la tecnología partiendo desde la era de la computación personal, a la era de la Web 1.0 (páginas y documentos Web), a la era de la Web 2.0 o era de las redes sociales, hasta la era de la Web 3.0 o era de la Web Semántica. Nova también sostiene que en la era de la Web 4.0, la Web se convertirá en un sistema operativo global el cual constituirá un sistema de datos que abarcará toda la red.

El presente proyecto muestra el desarrollo de un caso de estudio de Web Semántica llamado SemanticKipu. Para llevarlo a cabo se implementó un sitio Web para soluciones de negocios y aprendizaje virtual (VLBS) en que los usuarios registrados puedan disponer de una página de perfil personal la cual en su marcado HTML contenga micro-formatos embebidos tales como RDF y ontologías FOAF. Se siguió las fases de la metodología OOHDM que permitió demostrar el funcionamiento de la Web Semántica y sus agentes. Los resultados muestran que los usuarios pueden obtener RDFs los cuales contienen información personal y red de amigos de cada uno de sus integrantes.

El resto del artículo se ha estructurado de la siguiente manera: En la sección dos se presenta el marco teórico empleado. En la sección tres se puede apreciar el proceso de diseño e implementación. La sección cuatro presenta los resultados obtenidos. La sección cinco muestra los trabajos relacionados. Finalmente la sección seis muestra las conclusiones y trabajo futuros.

⁵ www.w3.org/2001/sw/

2. MARCO TEÓRICO

2.1 Definición de Web Semántica

Antes de definir que es Web Semántica, se exponen a continuación los problemas que esta técnica busca resolver:

- ❖ **Sobrecarga de información:** Google reportó haber indexado al menos 27 billones de páginas Web hasta el 2012 [2], sin embargo los índices de los buscadores Web indican que el número real de páginas Web con contenido significativo es mucho menor al número total de páginas indexadas.
- ❖ **Deficiente agregación y recuperación:** los algoritmos de búsqueda de páginas Web siguen dependiendo de modelos estadísticos independientes del contenido del mismo por lo que las variaciones generales de sintaxis y los errores ortográficos entorpecen la fiabilidad de las estadísticas con respecto a la relevancia de los documentos.
- ❖ **Sistemas unidireccionales:** el flujo de información en los sistemas es unidireccional es decir la información solo viaja hacia las aplicaciones de tal manera que no puede ser intercambiada con aplicaciones u organizaciones que estén fuera del sistema original.

De acuerdo con Hebler en [3] la "La Web Semántica, es la Web que contiene datos que están descritos y relacionados de manera que contexto y semántica puedan ser establecidos al usar construcciones definidas tanto gramaticales como de lenguaje. Este mismo autor amplía: "La Web Semántica añade significado a su contenido a través de conexiones estandarizadas entre información relacionada, esto incluye el etiquetado de datos de manera que estos se conviertan en únicos, y sean fáciles de acceder o encontrar. Cada dato único luego es conectado a un contexto mayor". Finalmente, se puede afirmar que en la Web actual los datos se presentan como *datos aislados*, es decir carecen de relaciones universales o de significado.

Tim Berners-Lee define a la Web Semántica como una extensión de la Web actual en donde la información contiene significado bien definido, de manera que humanos y computadores sean capaces de trabajar en cooperación [4].

2.2 Elementos de la Web Semántica

2.2.1 Datos y metadatos

Los datos son valores, átomos individuales de información, y los metadatos describen la relación entre los átomos de información. Mientras que los datos tienden a cambiar frecuentemente, cambios en los metadatos no son frecuentes. Para permitir a los computadores compartir información de forma automática, datos y metadatos deben ser agrupados, de esta manera la información se convierte en información portable ya que las relaciones entre los valores de los datos permanecen independientes de su almacenaje.

2.2.2 Marco de Trabajo de Descripción de Recursos (RDF)

Fue creado en 1999 por la W3C como un estándar para la codificación de metadatos. Es definido como un estándar publicado por la W3C, que puede ser usado para la representación de información o conocimiento distribuido, de manera que aplicaciones de ordenador puedan usarlo y procesarlo de forma escalable [5]. RDF utiliza un modelo abstracto para la representación del conocimiento acerca del mundo real. Este modelo abstracto consiste en

descomponer la información en piezas pequeñas utilizando reglas semánticas simples; sus componentes principales son:

- ❖ **Declaraciones** [Statement]: es un fragmento de información o conocimiento el cual es expresado como una lista de declaraciones, cada declaración toma la forma de: Sujeto-Predicado-Objeto, y su orden nunca debe ser cambiado. Cada declaración siempre está compuesta de tres componentes por lo que también es llamada triple. Cada declaración o triple representa un hecho individual y una colección de declaraciones o triples representan un fragmento de información o conocimiento; una colección de declaraciones es llamada gráfico RDF. En RDF las entidades que cualquier sujeto u objeto dado representan son también llamadas recursos [resources]. Los dos sujeto y objeto de una declaración son nombres de recursos.
- ❖ **Recursos y sus nombres URI**: el nombre de un recurso debe ser global y debe ser identificado mediante un Uniform Resource Identifier (URI). Un URI puede representar cualquier recurso sin importar si este puede ser devuelto o no electrónicamente.
- ❖ **Predicados y sus nombres URI**: en una declaración RDF el predicado denota la relación entre sujeto y objeto.

2.2.3 Microformats

Son una colección de micro- formatos individuales los cuales son usados para representar un dominio específico (como por ejemplo: personas, eventos, locaciones) que puede ser descrito por el contenido de una página Web. Cada uno de estos micro-formatos provee de un método para agregar anotaciones semánticas a páginas Web de tal manera que la información añadida pueda ser extraída y procesada por aplicaciones de software. Entre los micro-formatos más utilizados están:

- ❖ **hCard**: específica información acerca de personas, compañías y organizaciones, está basado en el estándar vCard RFC 242616⁶.
- ❖ **hCalendar**: formato abierto que permite publicar eventos en la Web, está basado en la representación iCalendar descrita en el estándar RFC 224517⁷.
- ❖ **XOXO**: describe bocetos o resúmenes, está escrito siguiendo el estándar XHTML y es adecuado para embeber en RSS y Atoms. Está definido en:
<http://microformats.org/wiki/xoxo>

2.2.4 Ontologías

Una ontología usa un vocabulario de términos predefinido y reservado para definir conceptos y relaciones entre los mismos términos en un área específica de interés o dominio. Una ontología puede hacer referencia a un vocabulario o a una taxonomía. Usualmente representa un modelo lógico formal el cual es utilizado para describir un dominio de conocimiento. Al utilizar ontologías se puede expresar la semántica que se encuentra escondida tras un vocabulario de términos, sus interacciones y su contexto de uso. Las

⁶ <http://microformats.org/wiki/hcard>

⁷ <http://microformats.org/wiki/rfca2445>

ontologías nos permiten acordar el significado de los términos usados en un dominio definido, sabiendo que muchos de los términos tal vez representen el mismo concepto (sinónimos) y tal vez muchos conceptos describen el mismo término (ambigüedad). Una ontología consiste de una descripción jerárquica de conceptos importantes de un dominio y de una descripción de las propiedades de cada concepto. En el campo de la obtención e inferencia de información una ontología especifica directamente el significado de los conceptos a ser buscados.

- ❖ **RDF Schema (RDFS)**: provee de un vocabulario específico para RDF que puede ser usado para definir taxonomías de clases, propiedades, dominios simples y rangos de especificaciones para propiedades. RDFS no pretende definir un vocabulario que puede ser compartido o utilizado masivamente, más bien provee de un lenguaje con el cual se pueden crear vocabularios personalizados o específicos.

- ❖ **Web Ontology Lenguaje (OWL)**: el lenguaje de ontologías Web extiende el vocabulario RDFS con recursos adicionales los cuales pueden ser utilizados para la creación de ontologías aún más expresivas para la Web. El vocabulario OWL está definido así mismo en el nombre de espacio <http://www.w3.org/2002/07/owl#> y es usualmente referido con el prefijo *owl*. OWL actualmente está en su versión 2. OWL2 extiende el vocabulario original de OWL y reusa los mismos nombres de espacio. OWL obedece el principio de la **Suposición de Mundo Abierto [open world assumption]** [3]. La cual formula que el no conocer si una declaración es explícitamente verdadera no implica que la declaración sea falsa. Bajo esta suposición, puede ser añadida nueva información siempre; puede ser contradictoria, pero no puede remover información previamente afirmada. Ontologías OWL son comúnmente almacenadas como documentos en la Web. Los tres bloques fundamentales de una ontología son: clases, individuos, y propiedades. Una clase es un conjunto de recursos. Un individuo es cualquier recurso que es miembro de al menos una clase. Una propiedad es usada para describir un recurso. Las ontologías en Web Semántica ayudan a compartir el entendimiento común de la estructura de la información entre personas y utilidades de software que posibilitan la reutilización de dominios de conocimiento.

2.2.5 Friend of a Friend (FOAF)

En palabras simples FOAF puede ser definido tanto como un vocabulario u ontología, los cuales incluyen los términos básicos necesarios para describir información personal como por ejemplo: quien eres, que haces, y quienes son tus amigos. Este sirve como estándar para todo aquel que requiera marcar su página Web personal como un documento entendible y procesable por ordenadores. *“Friend Of A Friend es un proyecto que pretende crear una Web de documentos que sean entendibles por ordenadores, los cuales describan a personas, relaciones entre ellas y las cosas que ellas crean y hacen.”*[<http://www.foaf-project.org/>] Se debe recalcar que FOAF como ontología no es un estándar de la W3C; FOAF es manejada siguiendo el estilo de los estándares de proyectos Open Source y de Software Libre, es decir estándares y mantenimiento son proporcionados por una comunidad de desarrolladores. Sin embargo FOAF depende directamente de los estándares de la W3C, por ejemplo de los estándares RDF y OWL. La ontología FOAF está escrita en OWL. Documentos FOAF deben ser documentos RDF bien construidos.

Una ontología FOAF es una colección de términos los cuales están identificados por un

URI predefinido el cual corresponde a la siguiente cadena de caracteres: **http://xmlns.com/foaf/0.1/** Por convención este URI es asociado con el prefijo de nombre de espacio *foaf*: el cual es comúnmente utilizado en documentos RDF/XML. A continuación se describe el término **foaf: Person**.

- ❖ **foaf: Person** es una de las clases que forma parte de las clases núcleo definidas en el vocabulario de FOAF, representa a una persona en el mundo real. Las propiedades definidas por FOAF pueden ser usadas para definir a una persona a un nivel detallado. Por ejemplo, **foaf: firstName** es una propiedad que describe el nombre cristiano de una persona; esta propiedad tienen como dominio a **foaf: Person**, y su rango de valores está definido en: **http://www.w3.org/2000/01/rdf-schema#Literal** De manera similar **foaf: givenName** es una propiedad que describe el nombre dado a una persona, esta propiedad tiene el mismo dominio y rango que **foaf: firstName**. La propiedad **foaf: homepage** está relacionada con la página Web personal de un individuo como un recurso dado, su dominio es **http://www.w3.org/2002/07/owl#Thing**, y su rango es **foaf: Document**. Esta propiedad es una propiedad funcional inversa, es decir un individuo (o un objeto o cosa) puede tener varias páginas personales, sin embargo si dos individuos (objetos o cosas) tienen la misma página personal esto significa que los dos individuos (objetos o cosas) son el mismo. La propiedad **foaf: mbox_sha1sum** tiene la misma funcionalidad de **foaf: mbox** pero mientras que **foaf: mbox** es una representación textual simple de una dirección de correo electrónico, **foaf: mbox_sha1sum** toma una dirección de correo electrónico y aplica el algoritmo SHA1, lo cual genera un cadena de caracteres encriptado que contiene la dirección de correo electrónico de la persona que se desea describir. La propiedad **foaf: knows** es utilizada para describir la relación entre una persona y otra. La más importante característica de **foaf: knows** es que nos permite conectar archivos FOAF unos con otros. La propiedad **foaf: knows** está ligada directamente con la propiedad **rdfs: seeAlso**, si las dos propiedades son proveídas juntas al mismo tiempo, dos documentos RDF diferentes pueden ser conectados o relacionados directamente. La propiedad **rdfs: seeAlso** está definida en el nombre de espacio del esquema RDF, e indica que existe información adicional sobre el recurso que la propiedad está describiendo. En la comunidad FOAF la propiedad **rdfs: seeAlso** es considerada como un hipervínculo entre documentos FOAF. Es a través de esta propiedad que una Web de metadatos entendibles por ordenadores puede ser construida. La propiedad **foaf: depicts** es la propiedad inversa; expresa la relación entre una imagen y algo que la imagen representa. Para concluir podemos establecer los siguientes hechos:

- ❖ Cuando hablo acerca de mí mismo en un documento FOAF, estoy utilizando un lenguaje que es entendible por computadores.
- ❖ Para un computador, mi documento FOAF se convierte en mi nueva página Web personal, la cual que para ojos humanos puede resultar confusa, para un ordenador es perfectamente entendible.
- ❖ Si se asume que todos mis amigos han creado sus propios documentos FOAF, yo puedo apuntar a estos desde mi documento FOAF mediante el uso de las propiedades **foaf: knows** y **rdfs: seeAlso**.

3. IMPLEMENTACIÓN

Siendo Semantic Kipu un prototipo de aplicativo Web, su diseño, desarrollo e implementación se lo hizo siguiendo la metodología OOHD, cuyo proceso constó de cuatro etapas: diseño conceptual, diseño Navegacional, diseño de interfaces abstractas e implementación. SemanticKipu consiste en un esquema simple de scripts PHP los cuales se detallan a continuación:

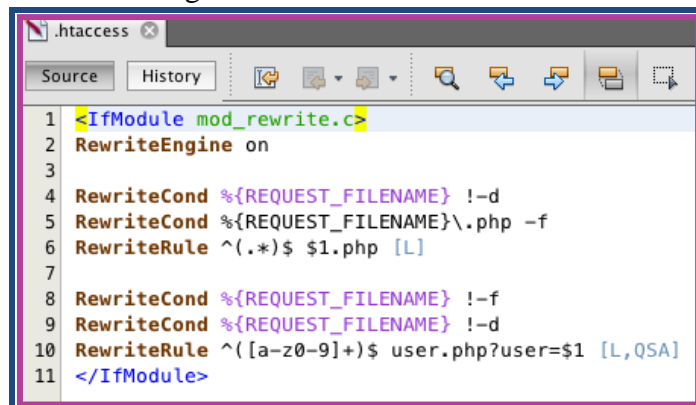
- ❖ **user.php**.- esta página presenta la información de perfil del usuario y su red de contactos.
- ❖ **index.php**.-esta es la página por defecto de Semantic Kipu la cual mostrará el formulario de registro y de ingreso o un mensaje de bienvenida para el usuario que haya accedido.
- ❖ **list.php**.- muestra una lista de los usuarios registrados con enlaces a sus páginas de perfil.
- ❖ **profile.php** - en esta página un usuario puede editar su perfil.
- ❖ **login.php**.- esta página procesa el ingreso del usuario al sistema, inicia una sesión y luego re direcciona al usuario a la página index.php.
- ❖ **logout.php**.- esta página finaliza la sesión de un usuario y re direcciona a la página index.php.
- ❖ **add-friend.php**.- esta página agrega amigos a la red de contactos de un usuario.
- ❖ **register.php**.- esta página agrega a un nuevo usuario a la red social.
- ❖ **update-user.php**.- esta página actualizada los datos de un usuario registrado.
- ❖ **connection.php**.- aquí se contienen los parámetros necesarios para conectar la aplicación con una base de datos.
- ❖ **functions.php**.- contiene funciones comunes necesarias para validaciones de datos ingresados por el usuario.

Aunque estos son implementados como paginas PHP se utilizará el modo de re-escritura de apache server (**mod rewrite**) para ocultar las extensiones de los archivos de usuarios, así por ejemplo los URIs como: <http://www.vlbs.net/SemanticKipu/list> será el URI de la página: <http://www.vlbs.net/SemanticKipu/list.php>. Toda página que no sea una página especial va a ser una página de detalle del usuario. Por ejemplo: <http://www.vlbs.net/SemanticKipu/alejandro> que en realidad es generado por la URI: <http://www.vlbs.net/SemanticKipu/user.php?user=alejandro> va a ser la página del usuario Alejandro.

3.1 .htaccess & RewriteRule

Se utiliza re-escritura condicional (**RewriteCond**) para comprobar si la URI solicita un archivo (f) o un directorio (d) que ya existe. Si el archivo PHP existe, se redirige al mismo y se detiene el procesamiento (bandera [L]). Se debe recordar que SemanticKipu aceptará URIs de la forma [http://url/\[nombre de usuario\]](http://url/[nombre de usuario]), lo cual re-direccionará a user.php, y si se pasa el nombre de un archivo sin la extensión .php, se re-direccionará al archivo PHP adecuado. Por

lo que la página `index.php` será accesible como: `http://url/index` sin la extensión del archivo. En la Figura 1 se muestra la configuración del archivo `.htaccess`:



```
1 <IfModule mod_rewrite.c>
2 RewriteEngine on
3
4 RewriteCond %{REQUEST_FILENAME} !-d
5 RewriteCond %{REQUEST_FILENAME}\.php -f
6 RewriteRule ^(.*)$ $1.php [L]
7
8 RewriteCond %{REQUEST_FILENAME} !-f
9 RewriteCond %{REQUEST_FILENAME} !-d
10 RewriteRule ^([a-z0-9]+)$ user.php?user=$1 [L,QSA]
11 </IfModule>
```

Figura 1: Configuración archivo `.htaccess`

Para lograr URIs de la forma `http://url/[nombre de usuario]` mencionada se utilizará la regla: `RewriteRule ^([a-z0-9]+)$ user.php?user=$1 [L,QSA]` Las cuatro partes que constituyen una regla típica de re-escritura (`RewriteRule`) son: La primera parte es la directiva en sí mismo; luego se tiene una expresión regular (el patrón) para que coincida con la URI, seguido de una URI para asignarlo a la sustitución. La variable `$1` en la regla corresponde a lo que coincide con `([a-z0-9]+)` lo que significa cualquier secuencia alfanumérica. El cuarto parámetro es una bandera de tipo opcional. Se pueden agregar las banderas que se requieran separándolas por comas. En este caso la bandera **L** detiene el procesamiento del conjunto de reglas y la bandera **QSA** indica que las sentencias podrán ser combinadas (patrón y sustitución).

3.2 hCard

Una característica de RDF es que los sustantivos en los triples son usualmente referidos por URIs. En SemanticKipu, la página de inicio del usuario será la cual lo representará. Para que esta representación sea útil se implementará el micro formato hCard, de tal manera que cuando la página sea vista se podrán fácilmente exportar los detalles del contacto a una vCard. hCard es una manera de marcar HTML estándar para indicar nombres de los campos como se especifican en el formato electrónico de tarjetas de negocios RFC 2422 vCard. vCard es comprendido por los clientes de email más populares. Se pueden obtener una vCard de cualquier usuario de SemanticKipu mediante el uso de **Operator**⁸ (Add-on para Firefox). hCard es una representación de una vCard en HTML.

En la Figura 2 se muestra la codificación vCard como hCard en HTML. Cabe recalcar que el marcado hCard agregado en el marcado HTML no tiene ningún efecto visible en la página cuando se cargue (a menos que se agregue una regla en el CSS para la clase usada para marcar propiedades hCard). A pesar de que en SemanticKipu, las clases hCard son mayormente aplicadas a los elementos `span` y `div`, estas pueden ser aplicadas a cualquier elemento HTML.

Cualquier fecha que se requiera representar dentro de un micro-formato debe seguir el

⁸ <https://addons.mozilla.org/es/firefox/addon/operator>

formato ISO8601⁹ utilizando "Date Time Design Pattern"¹⁰. La fecha legible para humanos estará encerrada dentro de un elemento *abbr* y el valor actual en formato ISO8601 es puesto en el atributo *title*, como se observa en la Figura 3.

Tabla 1: Campos BD vs. Propiedades hCard

Campo base de datos	Propiedad hCard
Username	fn, n, nickname
user_email	Email
user_birthday	Bday
user_form	Adr
user_avatar	Photo
user_Website	url
user_occ	Role

En la Tabla 1 se muestra la relación entre los atributos de la base de datos y las propiedades de hCard utilizadas en el marcado HTML de la página de perfil de usuario.

```

1 <div class="vcard">
2 <a class="url fn" href="http://www.exert-tech.com/">Alejandro</a>
3 (<span class="title"> Desarrollador Senior</span>)
4 <div>Email:
5 <span class="email">
6 <span class="type">internet</span>
7 <span class="value">favm@email.com</span>
8 </span>
9 </div>
10 <div class="adr">
11 <span class="locality">Quito</span>,
12 <span class="country-name">Ecuador</span>
13 </div>
14 </div>
15
    
```

Figura 2: Codificación vCard como hCard en HTML.

```

<div>
  Cumpleaños:
  <abbr class="bday" title="1985-12-14"> 14 diciembre </abbr>
</div>
    
```

Figura 3: Ejemplo uso Date Time Design Pattern

3.3 RDF a partir de hCard utilizando GRDDL

GRDDL permite obtener RDF de documentos existentes en la Web, incluyendo aquellos marcados con propiedades hCard. Para permitir que un agente GRDDL obtenga automáticamente un RDF a partir de un hCard se necesita agregar un atributo al elemento head del script **user.php** (véase la Figura 4).

⁹ <http://www.w3.org/TR/NOTE-datetime>

¹⁰ <http://microformats.org/wiki/datetime-design-pattern>

```

<head profile="http://www.w3.org/2003/g/data-view http://www.w3.org/2006/03/hcard">
  <title>alejandro</title>
  <link href="css/reset.css" type="text/css" rel="stylesheet">
  <link href="css/style.css" type="text/css" rel="stylesheet">
  <link href="http://www.w3.org/2006/vcard/hcard2rdf.xsl" rel="transformation">
</head>

```

Figura 4: Implementación GRDDL en marcado HTML

Se deben agregar dos URIs uno en el atributo *profile* y otro en el atributo *link*, con esto se especifica al agente GRDDL que tipo de documento debe esperar. Luego se debe indicar la transformación adecuada que se debe realizar para extraer la información en RDF del documento, para lograr esto se agrega en el elemento *link* el atributo *rel* (atributo de transformación) en la cabecera del documento. La W3C ha desarrollado un agente capaz de realizar la transformación de hCard a RDF el cual está disponible en: <http://www.w3.org/2007/08/grddl/>, para utilizar este agente solo se requiere copiar el URI de un perfil de un usuario de SemanticKipu y el agente devolverá el archivo RDF del perfil de usuario.

2.6 Red social utilizando FOAF

Se representará FOAF en XHTML utilizando RDFa, que es una extensión para XHTML que permite embeber información RDF. RDFa se basa en XML y a su vez XHTML es básicamente HTML expresado como XML válido. RDFa extiende XHTML implementando atributos extras y hace un uso más formal de algunos atributos ya existentes. Los atributos claves existentes de XHTML son: *rel*, *rev*, *name*, *content*, *href*, *src*, *resource*, *datatype*, *typeof*. Para utilizar atributos RDFa se debe utilizar un DTD (definición de tipo de documento) específico y namespaces a continuación mostrados (véase Figura 5).

```

1 <?xml version="1.0" encoding="UTF-8" ?>\n
2 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML+RDFa 1.0//EN"
3   "http://www.w3.org/Markup/DTD/xhtml-rdfa-1.dtd">
4 <html xmlns="http://www.w3.org/1999/xhtml"
5   xmlns:foaf="http://xmlns.com/foaf/0.1/">
6

```

Figura 5: Encabezado para XHTML+RDFa

Los atributos FOAF serán embebidos en los elementos estándar de XHTML (Ver Figura 6).

```

<div typeof="foaf:Person">
  <a property="foaf:name" rel="foaf:homepage"
    href="http://exert-tech.com">Alejandro</a>
  <div class="social-network" about="#me" rel="foaf:knows">
    <ul>
      <li typeof="foaf:Person">
        <a property="foaf:name" rel="foaf:homepage"
          href="http://www.vlbs.net/SemanticKipu/joel">Joel</a>
      </li>
    </ul>
  </div>
</div>

```

Figura 6: FOAF en RDFa

Los atributos RDFa mostrados en el ejemplo anterior se detallan en la siguiente lista:

- ❖ *typeof*.- Es equivalente a "rdf: type" en RDF este atributo indica la clase de cosa representada por el objeto contenido en él.
- ❖ *property*.- Define una relación entre un ente (que lo contiene) y un "bit" literal de texto.
- ❖ *rel*.- Define la relación entre un ente y otro ente.
- ❖ *about*.- Equivalente a "rdf: subjet" en RDF. Especifica el tema de los descendientes de un elemento.
- ❖ *href*.- El atributo *href* junto con *rel* apuntan a un objeto real del cual *rel* define una relación. Es equivalente a *rdf:subject* en RDF.

Con el fin de comprobar que el marcado RDFa está escrito de manera correcta, y de obtener el RDF de un usuario registrado en SemanticKipu se puede utilizar el W3C RDFa Distiller, disponible en: <http://www.w3.org/2007/08/pyRdfa/>

4. RESULTADOS Y PRUEBAS DE FUNCIONALIDAD

2.1 Pruebas de funcionalidad

El usuario puede visualizar la lista de contactos registrados en SemanticKipu seleccionando la opción Lista de Contactos, cuando el usuario selecciona a cualquiera de los miembros de la lista puede visualizar la página de perfil de dicho contacto; si se dispone del add-on Operator en el navegador Firefox, se puede visualizar que el mismo se activa. Operator permite: exportar el contacto, guardar el vCard del contacto, visualizar la posición geográfica del contacto en GoogleMaps, etc. (véase Figura 7).

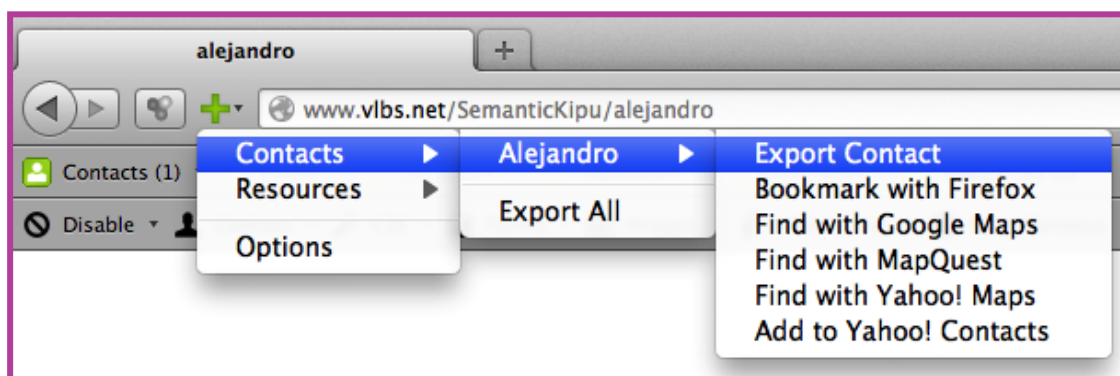
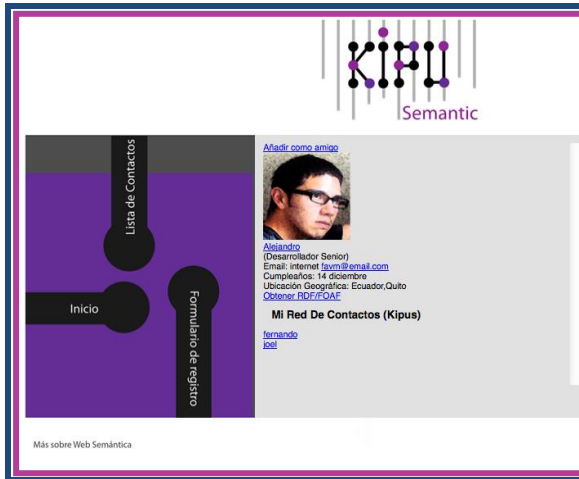


Figura 1 : Operator en Firefox

Los usuarios pueden registrarse en SemanticKipu, editar su información personal, añadir a otros usuarios registrados a su red de contactos. Toda la información de un usuario registrado se muestra en su página de perfil. Las Figuras 8 y 9 muestran lo señalado:



```

- <rdf:RDF>
- <rdf:Description rdf:about="http://www.vlbs.net/SemanticKipu/alejandro#me">
- <foaf:knows>
- <foaf:Person>
  <foaf:name>fernando</foaf:name>
  <foaf:homepage rdf:resource="http://www.vlbs.net/SemanticKipu/fernando"/>
</foaf:Person>
</foaf:knows>
- <foaf:knows>
- <foaf:Person>
  <foaf:name>joel</foaf:name>
  <foaf:homepage rdf:resource="http://www.vlbs.net/SemanticKipu/joel"/>
</foaf:Person>
</foaf:knows>
</rdf:Description>
- <foaf:Person>
  <foaf:name>Alejandro</foaf:name>
  <foaf:homepage rdf:resource="http://www.exert-tech.com"/>
</foaf:Person>
</rdf:RDF>
    
```

Figura 8: Página de perfil de un usuario.

Figura9: RDF de un contacto registrado

Por requerimiento de VLBS, SemanticKipu debe admitir 10 usuarios concurrentes, se realizaron pruebas de carga utilizando JMeter¹¹ (véase Figura 10).

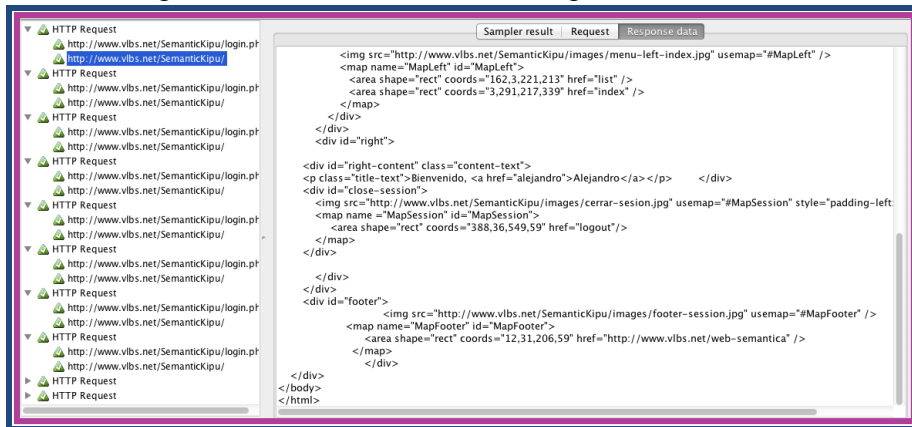


Figura 10: Prueba de carga sobre SemanticKipu

2.2 Resultados

- ❖ Se puede obtener un archivo RDF\FOAF de cualquiera de los usuarios registrados en SemanticKipu.
- ❖ El prototipo SemanticKipu demuestra el funcionamiento de la Web Semántica y sus agentes.
- ❖ La página de perfil de cada usuario contiene embebida información que puede ser exportada utilizando un operador semántico como por ejemplo Operator para Firefox.
- ❖ Los hCard embebidos en la página de perfil de cada usuario son reconocidos por clientes email como por ejemplo: Thunderbird, Contacts (OSX), etc.

2.3 Discusión

Los micro-formatos están diseñados para trabajar con etiquetas HTML ya existentes y

¹¹ <http://jmeter.apache.org>

agregan valores semánticos para casos específicos. Si se tienen datos valiosos en un sitio Web, son la opción ideal. En este contexto, RDF es un formato estructurado y tiene una relación claramente definida hacia todas las tecnologías de Web Semántica. Por tanto, al crear una solución basada en Web Semántica el lema debe ser "siempre buscar fuentes semánticas existentes (ontologías o vocabularios) y añadir personalizaciones a las mismas". El crear fuentes de información desde cero debe ser considerado como último recurso. Finalmente, sobre la base de las consideraciones anteriores, se considera un mito el creer que Web Semántica constituye o es inteligencia artificial, el añadir semántica a través de relaciones y lógica no constituye inteligencia artificial. La Web Semántica ofrece un avance útil para aprovechar al máximo la información.

5. CONCLUSIONES

El presente proyecto desarrolló de un caso de estudio de Web Semántica denominado SemanticKipu. Para lograrlo se implementó un sitio Web para soluciones de negocios y aprendizaje virtual para que los usuarios registrados puedan disponer de una página de perfil personal la cual en su marcado HTML contenga micro-formatos embebidos tales como RDF y ontologías FOAF. Se siguió las fases de la metodología OOHDM que permitió demostrar el funcionamiento de la Web Semántica y sus agentes. Los resultados muestran que los usuarios pueden obtener RDFs los cuales contienen información personal y red de amigos de cada uno de sus integrantes.

REFERENCIAS BIBLIOGRÁFICAS

- [1] Pollock, J. T.. Semantic Web for Dummies. Hoboken: Wiley Publishing, Inc. (2009)
- [2] Sugumaran, V. Applied Semantic Web Technologies. Boca Raton: CRC Press. (2012).
- [3] Hebel, J. Semantic Web Programming. Indianapolis: Wiley Publishing, Inc., (2009).
- [4] Berners-Lee, T., Hendler, J. & Lassila, O. The Semantic Web. Scientific American, pp. 34-43. (May, 2001)
- [5] Yu, L. A Developer's Guide to the Semantic Web. Atlanta, USA: Springer, Berlin. (2011)
- [6] Akerkar, R.. Foundations of the Semantic Web. Oxford: Alpha Science Intl Ltd. (2009)
- [7] Cardoso, J., Hepp, M., & Lytras, M.. The Semantic Web - Real World Applications from Industry. New York: Springer. (2008)
- [8] Fensel, D., Facca, F. M., Simperl, E., & Toma, I. Semantic Web Services. Berlin: Springer. (2011)
- [9] Gruber, T. R.. A Translation Approach to Portable Ontology Specifications. In Knowledge Acquisition: pp. 1999-220, (1993)
- [10] IEEE. (22 de October de 1998). IEEE Xplore - IEEE STD 830-1998. Retrieved 1 de April de 2012 from IEEE Xplore Digital Library:
<http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=5841>
- [11] Pressman, R. Software Engineering a Practitioner's Approach. New York: McGraw-Hill Higher Education, (2002)
- [12] Rossi, G., Pastor, O., Schwabe, D., & Olsina, L. Web Engineering: Modelling and Implementing Web Application. London: Springer. (2008).