



**Centro de Informatización Universitaria  
Facultad 1**

***Trabajo de Diploma para optar por el título de Ingeniero en  
Ciencias Informáticas***

**Título:** Servicio: perfil de usuario de la Red Social para la  
Universidad de las Ciencias Informáticas.

**Autora:** Yisel Gómez Herrera

**Tutores:** Ing. Yunier Saborit Ramírez

Ing. Cesar González Hernández

La Habana, Junio 2012  
“Año 54 de la Revolución”

Declaración de autoría

Declaro que soy el único autor de este trabajo y autorizo al Centro de Informatización Universitaria de la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

Para que así conste firmo la presente a los \_\_\_\_\_ días del mes de \_\_\_\_\_ del año \_\_\_\_\_.

\_\_\_\_\_  
Yisel Gómez Herrera

\_\_\_\_\_  
Ing. Yunier Saborit Ramírez

\_\_\_\_\_  
Ing. Cesar González Hernández

## Síntesis de los tutores

Yunier Saborit Ramírez:

Graduado de Ingeniero en Informática en el 2004. Realizó su primera actividad profesional en proyectos ese mismo año en el apoyo informático a la misión milagro. A partir de 2005 se incorpora a la Dirección de Informatización como especialista, cumpliendo el rol de arquitecto de la dirección. En el año 2006 pasa a ser Director de Informatización, cargo que ocupa actualmente. Ha tutelado varias tesis desde su graduación. Resumen de Tesis que ha tutelado: Sistema de Apoyo Logístico a la Misión Milagro (2004-2005), Análisis y Diseño del Catálogo de Servicios UDDI para la Informatización de la UCI (2006-2007), Propuesta de Arquitectura para la Migración a Software Libre del Sistema de Gestión Académica, Akademos (2006-2007). Sistema para la informatización del Convenio Cuba-Venezuela. Módulo ficha mixta (2006-2007). Correo electrónico: [yony@uci.cu](mailto:yony@uci.cu).

César González Hernández:

Graduado de Ingeniero Informático en la Universidad de Holguín (UHo) en el año 2006. Posee la categoría docente Instructor. Pertenece al Dpto. Universidad Digital donde se desempeña como Jefe de Dpto. Correo electrónico: [cgonzalezh@uci.cu](mailto:cgonzalezh@uci.cu).

## Agradecimientos

*A mis padres por confiar en mí y brindarme su apoyo incondicionalmente durante estos 5 años de carrera, por darme ánimos aún cuando están lejos de mí físicamente, pero espiritualmente siempre los llevo conmigo, por sacrificarse por mí, les agradezco todo lo que soy hasta este momento y lo que puedo llegar a ser. A mi mamá por brindarme siempre su amor y cariño y estar ahí siempre que la necesite, por darme la fuerza de seguir adelante. A mi papá, por ser un ejemplo en mi carrera, espero que estés orgulloso de mí. Los quiero infinitamente.*

*A mi familia por apoyarme en mi carrera y confiar en mí en especial, a mis tías y tíos por darme su apoyo y confiar en mí. A mis abuelos por darme los mejores padres del mundo.*

*A mi hermana por su cariño, a mi hermano por ser tan especial y siempre estar a mi lado siempre que lo necesito, gracias por tus consejos y apoyo.*

*A José Miguel por ser la persona que ha estado a mi lado en las buenas y malas, por apoyarme en todo momento, por confiar siempre en mí, nunca me voy a cansar de agradecerte.*

*A mis amigos y amigas que durante todos estos años he compartido. A Annerys por brindarme su mano amiga desde que ingresé a la universidad, sabes que siempre puedes contar conmigo.*

*A Osmel por su cariño y amistad incondicional, por su confianza, por creer siempre en mí, mil gracias, nunca olvidaré todo lo que has hecho por mí. A Daymí por brindarme su cálido hogar por hacerme sentir parte de la familia.*

*A todas las personas que contribuyeron en mi formación, especialmente a mis profesores.*

*A mis tutores, especialmente al profesor Cesar, por dedicarle parte de su tiempo a este trabajo de diploma que fue bien difícil sacar tiempo de donde no había. Al oponente Miguel Jaeger por ser un buen guía en el proyecto y brindar su mano amiga incondicionalmente, al tribunal por contribuir al perfeccionamiento de esta investigación.*

## Resumen

En la Universidad de las Ciencias Informáticas (UCI), tanto los estudiantes como los profesores hacen uso de redes sociales foráneas suministrando grandes volúmenes de información. Para contrarrestar esta tendencia y poder aprovechar los conocimientos acumulados, la dirección de la institución se da la tarea de desarrollar una Red Social Universitaria (RSU) como parte de la informatización de los procesos sustantivos en el centro. El desarrollo del presente trabajo se enfoca en la realización del servicio: perfil de usuario para la RSU que permita la gestión de la información brindada por ellos en dicha red. Para ello se realiza el análisis del estado del arte de los perfiles de las redes sociales en el mundo, las ontologías para describir personas y las herramientas para la construcción de este tipo de sistemas. También se identificaron las funcionalidades que se desarrollan en la propuesta y las tecnologías a utilizar para el mismo. Se comprueba el correcto funcionamiento del sistema a través de: pruebas funcionales, pruebas de carga y estrés y pruebas de integración con otros servicios de la red social.

**Palabras clave:** ontologías, perfil de usuario, redes sociales.

Índice de contenido

<b>INTRODUCCIÓN</b> .....	<b>1</b>
<b>1 CAPÍTULO I: FUNDAMENTACIÓN TEÓRICA</b> .....	<b>5</b>
1.1 INTRODUCCIÓN .....	5
1.2 REDES SOCIALES .....	5
1.3 PERFILES DE USUARIO.....	8
1.4 REDES SOCIALES Y WEB SEMÁNTICA .....	12
1.5 PLATAFORMAS PARA DESARROLLAR REDES SOCIALES.....	16
1.6 TECNOLOGÍAS PARA EL DESARROLLO .....	22
1.7 CONCLUSIONES PARCIALES .....	27
<b>2 CAPÍTULO II PROPUESTA DE SOLUCIÓN</b> .....	<b>28</b>
2.1 INTRODUCCIÓN .....	28
2.2 DIAGNÓSTICO DEL CAMPO DE ACCIÓN .....	28
2.3 DESCRIPCIÓN DE LA PROPUESTA.....	28
2.4 MODELO DE DOMINIO.....	33
2.5 ESPECIFICACIÓN DE REQUISITOS.....	34
2.6 PATRONES DE ARQUITECTURA .....	41
2.7 PATRONES DE DISEÑO .....	42
2.8 MODELO DE DATOS .....	44
2.9 CONCLUSIONES PARCIALES .....	45
<b>3 CAPÍTULO III IMPLEMENTACIÓN Y VALIDACIÓN DE LA PROPUESTA</b> .....	<b>46</b>
3.1 INTRODUCCIÓN .....	46
3.2 DESARROLLO DE LA PROPUESTA DE SOLUCIÓN .....	46
3.3 ESTÁNDARES DE CODIFICACIÓN .....	47
3.4 DIAGRAMA DE DESPLIEGUE DEL SISTEMA.....	51
3.5 VALIDACIÓN .....	52
3.6 CONCLUSIONES PARCIALES .....	60
<b>CONCLUSIONES GENERALES</b> .....	<b>61</b>
<b>RECOMENDACIONES</b> .....	<b>62</b>
<b>BIBLIOGRAFÍA REFERENCIADA</b> .....	<b>63</b>
<b>BIBLIOGRAFÍA CONSULTADA</b> .....	<b>66</b>
<b>ANEXOS. DESCRIPCIÓN DE REQUISITOS</b> .....	<b>68</b>
<b>ANEXOS. DESCRIPCIÓN DE CASOS DE PRUEBAS</b> .....	<b>83</b>

Índice de tablas

TABLA 1.1 RESULTADOS DE LA COMPARACIÓN ENTRE REDES SOCIALES. ....	11
TABLA 2.1 ESPECIFICACIÓN DE REQUISITO. REALIZAR SOLICITUD DE AMISTAD. ....	36
TABLA 2.2 ESPECIFICACIÓN DE REQUISITOS. CONFIRMAR SOLICITUD DE AMISTAD. ....	37
TABLA 2.3 ESPECIFICACIÓN DE REQUISITOS. RECHAZAR SOLICITUD DE AMISTAD. ....	38
TABLA 3.1 DESCRIPCIÓN DE CASO DE PRUEBA. REALIZAR SOLICITUD DE AMISTAD. ....	54
TABLA 3.2 DESCRIPCIÓN DE CASO DE PRUEBA. ACEPTAR SOLICITUD DE AMISTAD. ....	54
TABLA 3.3 DESCRIPCIÓN DE CASO DE PRUEBA. REALIZAR SOLICITUD DE AMISTAD. ....	55
TABLA 3.4 PÁGINA DE INICIO. ....	57
TABLA 3.5 PÁGINA PERFIL. ....	57
TABLA 3.6 PÁGINA MIEMBROS. ....	57
TABLA 3.7 PRUEBA DE INTEGRACIÓN INT_1. NÚCLEO DE LA RED SOCIAL. ....	59
TABLA 3.8 PRUEBA DE INTEGRACIÓN INT_2. NÚCLEO DE LA RED SOCIAL. ....	59
TABLA 3.9 PRUEBA DE INTEGRACIÓN INT_3. API DE SERVICIOS TELEMÁTICOS. ....	60
TABLA 0.1 ESPECIFICACIÓN DE REQUISITOS. AUTENTICAR USUARIO. ....	68
TABLA 0.2 ESPECIFICACIÓN DE REQUISITOS. ADICIONAR ATRIBUTO AL PERFIL. ....	69
TABLA 0.3 ESPECIFICACIÓN DE REQUISITOS. EDITAR ATRIBUTOS DEL PERFIL. ....	70
TABLA 0.4 ESPECIFICACIÓN DE REQUISITOS. ELIMINAR ATRIBUTOS DEL PERFIL. ....	72
TABLA 0.5 ESPECIFICACIÓN DE REQUISITOS. CREAR TIPO DE PERFIL. ....	72
TABLA 0.6 ESPECIFICACIÓN DE REQUISITOS. EDITAR TIPO DE PERFIL. ....	73
TABLA 0.7 ESPECIFICACIÓN DE REQUISITOS. ELIMINAR TIPO DE PERFIL. ....	74
TABLA 0.8 ESPECIFICACIÓN DE REQUISITOS. BUSCAR AMIGOS. ....	75
TABLA 0.9 ESPECIFICACIÓN DE REQUISITOS. MOSTRAR PERFIL. ....	76
TABLA 0.10 ESPECIFICACIÓN DE REQUISITOS. MOSTRAR ATRIBUTOS DEL PERFIL. ....	76
TABLA 0.11 ESPECIFICACIÓN DE REQUISITOS. MOSTRAR TIPO DE PERFIL. ....	77
TABLA 0.12 ESPECIFICACIÓN DE REQUISITOS. MOSTRAR SOLICITUD DE AMISTAD. ....	78
TABLA 0.13 ESPECIFICACIÓN DE REQUISITOS. MOSTRAR SUGERENCIAS DE AMIGOS. ....	78
TABLA 0.14 ESPECIFICACIÓN DE REQUISITOS. MOSTRAR INFORMACIÓN DE SERVICIOS TELEMÁTICOS. ....	79
TABLA 0.15 ESPECIFICACIÓN DE REQUISITOS. ENVIAR MENSAJES. ....	79
TABLA 0.16 ESPECIFICACIÓN DE REQUISITOS. CERRAR SESIÓN. ....	81
TABLA 0.1 DISEÑO DE CASOS DE PRUEBA. AUTENTICAR USUARIO. ....	83
TABLA 0.2 DISEÑO DE CASOS DE PRUEBA. ADICIONAR ATRIBUTOS AL PERFIL 1. ....	83
TABLA 0.3 DISEÑO DE CASOS DE PRUEBA. ADICIONAR ATRIBUTOS AL PERFIL 2. ....	85
TABLA 0.4 DISEÑO DE CASOS DE PRUEBA. EDITAR ATRIBUTOS DEL PERFIL 1. ....	86
TABLA 0.5 DISEÑO DE CASOS DE PRUEBA. EDITAR ATRIBUTO DEL PERFIL 2. ....	88
TABLA 0.6 DISEÑO DE CASOS DE PRUEBA. ELIMINAR ATRIBUTO DEL PERFIL. ....	89
TABLA 0.7 DISEÑO DE CASOS DE PRUEBA. CREAR TIPO DE PERFIL. ....	89
TABLA 0.8 DISEÑO DE CASOS DE PRUEBA. EDITAR TIPO DE PERFIL. ....	91
TABLA 0.9 DISEÑO DE CASOS DE PRUEBA. ELIMINAR TIPO DE PERFIL. ....	93

Índice de figuras

FIGURA 1.1 CLASES PRINCIPALES DEL VOCABULARIO BIOGRÁFICO (DAVIS Y GALBRAITH, 2011).	15
FIGURA 1.2 ESTRUCTURA DE MAHARA	17
FIGURA 1.3 MODELO DE DATOS DEL ELGG (ELGG, 2011)	19
FIGURA 2.1 ARQUITECTURA DE LA RED SOCIAL	29
FIGURA 2.2 PROPUESTA DEL PERFIL DE USUARIO	30
FIGURA 2.3 MODELO DE DOMINIO	34
FIGURA 2.4 PROTOTIPO DE INTERFAZ. REALIZAR SOLICITUD DE AMISTAD 1	36
FIGURA 2.5 PROTOTIPO DE INTERFAZ. REALIZAR SOLICITUD DE AMISTAD 2	36
FIGURA 2.6 PROTOTIPO DE INTERFAZ. REALIZAR SOLICITUD DE AMISTAD 3	37
FIGURA 2.7 PROTOTIPO DE INTERFAZ. CONFIRMAR SOLICITUD DE AMISTAD	38
FIGURA 2.8 PROTOTIPO DE INTERFAZ. RECHAZAR SOLICITUD DE AMISTAD	38
FIGURA 2.9 MODELO DE DATOS	45
FIGURA 3.1 DIAGRAMA DE DESPLIEGUE DEL SISTEMA	52
FIGURA 3.2 CANTIDAD DE NO CONFORMIDADES POR ITERACIÓN	55
FIGURA 3.3 INTEGRACIÓN CON SERVICIOS DE LA RSU	58
FIGURA 0.1 PROTOTIPO DE INTERFAZ. AUTENTICAR USUARIO	68
FIGURA 0.2 PROTOTIPO DE INTERFAZ. ADICIONAR ATRIBUTO AL PERFIL	69
FIGURA 0.3 PROTOTIPO DE INTERFAZ. EDITAR ATRIBUTO DEL PERFIL	71
FIGURA 0.4 PROTOTIPO DE INTERFAZ. ELIMINAR ATRIBUTO DEL PERFIL	72
FIGURA 0.5 PROTOTIPO DE INTERFAZ. CREAR TIPO DE PERFIL	73
FIGURA 0.6 PROTOTIPO DE INTERFAZ. EDITAR TIPO DE PERFIL	74
FIGURA 0.7 PROTOTIPO DE INTERFAZ. ELIMINAR TIPO DE PERFIL	75
FIGURA 0.8 PROTOTIPO DE INTERFAZ. BUSCAR AMIGOS 1	75
FIGURA 0.9 PROTOTIPO DE INTERFAZ. BUSCAR AMIGOS 2	75
FIGURA 0.10 PROTOTIPO DE INTERFAZ. MOSTRAR PERFIL	76
FIGURA 0.11 PROTOTIPO DE INTERFAZ. MOSTRAR ATRIBUTOS DEL PERFIL	77
FIGURA 0.12 PROTOTIPO DE INTERFAZ. MOSTRAR TIPO DE PERFIL	77
FIGURA 0.13 PROTOTIPO DE INTERFAZ. MOSTRAR SOLICITUD DE AMISTAD 1	78
FIGURA 0.14 PROTOTIPO DE INTERFAZ. MOSTRAR SOLICITUD DE AMISTAD 2	78
FIGURA 0.15 PROTOTIPO DE INTERFAZ. MOSTRAR SUGERENCIAS DE AMIGOS	79
FIGURA 0.16 PROTOTIPO DE INTERFAZ. MOSTRAR INFORMACIÓN DE SERVICIOS TELEMÁTICOS	79
FIGURA 0.17 PROTOTIPO DE INTERFAZ. ENVIAR MENSAJE AL CORREO	80
FIGURA 0.18 PROTOTIPO DE INTERFAZ. ENVIAR MENSAJE AL JABBER	80
FIGURA 0.19 PROTOTIPO DE INTERFAZ. ENVIAR MENSAJE INTERNO	81
FIGURA 0.20 PROTOTIPO DE INTERFAZ. ENVIAR MENSAJE A TODOS	81
FIGURA 0.21 PROTOTIPO DE INTERFAZ. CERRAR SESIÓN 1	82
FIGURA 0.22 PROTOTIPO DE INTERFAZ. CERRAR SESIÓN 2	82

## Introducción

Las redes sociales se han convertido en todo un fenómeno social revolucionando la forma en que se comunican e interactúan las personas. Estas ofrecen valiosos beneficios en diversos ámbitos de la vida. Su atractivo esencial radica en la participación e interacción que proporcionan a los internautas.

“Las redes sociales rompen el aislamiento de muchas personas, posibilitan la combinación entre pluralidad y comunidad, al anonimato le da popularidad, a la discriminación, integración y a la diferencia; espacios de igualdad. Permite el establecimiento de lazos y relaciones con personas que comparten los mismos intereses, preocupaciones y necesidades, así como la actualización permanente de los contactos a diferencia de un servicio de correo electrónico. Con respecto al sector académico y laboral no se han quedado atrás, diversas experiencias innovadoras se reflejan en estas redes. Los empresarios que hacen uso de las redes sociales han demostrado un nivel de eficiencia y un acertado trabajo en equipo, consolidando proyectos de gestión del conocimiento” (AURELIO, 2012).

La Universidad de las Ciencias Informáticas (UCI), desde sus inicios ha estado encaminada a la formación de profesionales informáticos, la producción de aplicaciones y servicios así como servir de soporte a la industria cubana del software.

Tanto estudiantes como profesores, principales actores del proceso, hacen uso de las redes sociales de Internet, proveen una gran suma de información en sus perfiles, que se encuentra disgregada en redes foráneas y fuera del alcance de la institución, desaprovechándose un conocimiento de un alto valor agregado. En estas redes sociales se crean grupos en los que interactúan en su mayoría usuarios de la misma universidad. Estos perfiles recogen cierta cantidad de información sobre el usuario de manera que se puede ir enriqueciendo con la propia actividad de los miembros de la red en cada uno de sus servicios. A la vez que el usuario comparte con las personas que tiene alguna relación se forman lazos digitales mediante los cuales se puede caracterizar a una persona y conocer información de manera inferida gracias a los datos almacenados y las relaciones que existen entre los recursos. Además los perfiles de las redes sociales pueden ser un elemento de gran valor para los negocios y las empresas que pueden conocer las tendencias y preferencias de los usuarios de manera general para poder alcanzar mejores resultados.

Una red social puede ser un factor clave que podría traer crecidos beneficios a la universidad si se emplea como una herramienta de apoyo a los procesos sustantivos, de esta forma se podría potenciar el desarrollo de las investigaciones y el intercambio de conocimientos dentro de la comunidad.

También, fomentaría el trabajo colaborativo y sería una herramienta de gran utilidad como complemento al modelo de formación. Una red social universitaria constituiría un gran paso para el proceso de formación de los estudiantes y profesores, enriqueciendo el intercambio de conocimiento entre ellos y facilitando la creación de herramientas de aprendizaje más eficientes.

Tomando en cuenta la utilidad de una red social en nuestra universidad, se ha desarrollado un conjunto de servicios que se identificaron como componentes de software de la Red Social Universitaria (RSU), ejemplo de ellos son el servicio de mensajería instantánea, correo electrónico, Biblioteca Virtual, Entorno Virtual de Aprendizaje, Microblogging<sup>1</sup>, la plataforma de bitácoras temáticas y personales (*blogs*), el servicio de sindicación RSS<sup>2</sup>, los portales de las comunidades de desarrollo que potencia la FEU de la institución y otros, que toman en cuenta características de las redes sociales. La mayor parte de estos servicios se basan en la interacción entre sus usuarios, ya sea a través de comentarios, votos, o para compartir conocimientos y herramientas, siendo de gran utilidad para la comunidad universitaria y el punto de partida para muchas soluciones a problemas internos del centro de estudios.

La existencia de estos servicios viene aparejada de una gran cantidad de información que los usuarios aportan en su interacción con los mismos, estos datos se encuentran de manera dispersa y sin ninguna interrelación. También se plantea que la red de la universidad no cuenta con un espacio digital en el que los usuarios puedan publicar información referente a sus preferencias de manera general. Los espacios creados para publicar opiniones y comentarios se encuentran disgregados y sin ninguna interacción entre ellos. Los servicios desarrollados carecen de una estructura común en la forma en que almacenan los datos de los usuarios de forma que puedan ser reutilizados por otros servicios. No existe una única fuente de datos donde referirse a la información de los usuarios de la universidad que sean recogidos por la propia institución.

Luego de exponer los argumentos anteriores, se plantea el siguiente **problema a resolver**: ¿Cómo gestionar la información de los miembros de la red social en la Universidad de las Ciencias Informáticas?

El **objeto de estudio** está centrado en la gestión de usuarios en las redes sociales, y se especifica como **campo de acción** los perfiles de usuario para la red social de la Universidad de las Ciencias Informáticas.

---

<sup>1</sup> Es un servicio que permite a sus usuarios enviar y publicar mensajes breves (alrededor de 140 caracteres).

<sup>2</sup> RSS (Really Simple Syndication), es un formato XML para syndicar o compartir contenido en la web.

Con el propósito de resolver el problema propuesto se define como **objetivo general**: Desarrollar un servicio para la red social de la Universidad de las Ciencias Informáticas que permita gestionar toda la información asociada a los usuarios posibilitando la comunicación entre ellos y el resto de los servicios.

El objetivo general es desglosado en los siguientes **objetivos específicos**:

- ✓ Estudiar aspectos teóricos conceptuales sobre los servicios de perfiles de usuario, las redes sociales, las ontologías para describir personas así como las herramientas a utilizar.
- ✓ Proponer una solución para la gestión de información de los usuarios en la Red Social Universitaria.
- ✓ Implementar el servicio: perfil de usuario para la RSU haciendo uso de las herramientas previamente seleccionadas.
- ✓ Probar que el sistema funcione correctamente de acuerdo a los requisitos planteados.

Los **métodos científicos** que se utilizan para darle solución a los objetivos propuestos son:

Métodos Teóricos:

- ✓ Método Histórico-Lógico: este método permitió caracterizar, definir conceptos, términos y vocabularios de las redes sociales. Además posibilitó enfocar la investigación a las herramientas para el desarrollo de redes sociales que existen en el mundo, permitiendo arribar a conclusiones para la investigación.
- ✓ Método Analítico-Sintético: el empleo de este método permitió realizar un estudio y análisis de toda la bibliografía consultada referente al tema de investigación a tratar, donde se extrajeron los elementos más importantes relacionados con el objeto de estudio.
- ✓ Inductivo-Deductivo: se pone de manifiesto en la toma de decisiones que se generaron a partir del estudio y análisis de las redes sociales ya existentes en el mundo.

Métodos Empíricos:

- ✓ Método Observación: se utilizó para la realización del estado del arte de los perfiles de usuario de algunas redes sociales existentes en el mundo, así como las herramientas utilizadas para el desarrollo de las mismas.

**Justificación de la investigación:**

El desarrollo del servicio: perfil de usuario para la Red Social Universitaria tiene como principal objetivo garantizar la gestión de toda la información brindada por los usuarios en los perfiles de la red social. Se cuenta además con una herramienta capaz de establecer comunicación entre los miembros de la RSU, al tiempo que brinda sugerencias de posibles relaciones entre los usuarios y servicios a partir de la información que estos ingresan.

Finalmente se expone que el presente documento se encuentra estructurado de la siguiente forma:

**Capítulo I: Fundamentación teórica:** En este capítulo se estudian temas referentes a redes sociales y sus conceptos. Además se realiza un estudio homólogo de las herramientas para construir redes sociales y perfiles de usuario en redes sociales universitarias, donde se arriba a conclusiones que permiten dar respuesta a la problemática de la investigación.

**Capítulo II: Descripción de la solución propuesta:** Se realiza una valoración sobre el diagnóstico del campo de acción y de la propuesta solución. Se definen los principales requisitos a implementar en el servicio: perfil de usuario, así como la arquitectura y los patrones de diseño a utilizar. Contiene la construcción del servicio a partir de la realización de las tareas de ingeniería para darle cumplimiento a los requisitos planteados.

**Capítulo III: Implementación y validación de la propuesta:** Contiene la descripción de los elementos establecidos por el proceso de desarrollo empleado y las pruebas realizadas para validar el correcto funcionamiento de la aplicación.

### 1 Capítulo I: Fundamentación teórica

#### 1.1 Introducción

En la actualidad las redes sociales están siendo utilizadas por millones de personas en todo el mundo. Su crecimiento ha sido exponencial e integran comunidades de internautas prácticamente sobre cualquier tema.

En este primer apartado se estudian las temáticas referentes a las redes sociales generales y universitarias, así como los perfiles de usuario de estas. Además se analizan las definiciones sobre las diferentes herramientas para construir redes sociales y las ontologías para la descripción de personas.

#### 1.2 Redes sociales

Las redes sociales están cobrando gran importancia en la sociedad actual, reflejando la necesidad del ser humano de expresión y reconocimiento, bien lo dijo Aristóteles “El Hombre es un ser sociable por naturaleza, pues para satisfacer sus exigencias físicas y espirituales necesita vivir en sociedad, ya que al ser una criatura racional e individual, no es autosuficiente y requiere la ayuda y protección de los demás de su especie, por ello forma comunidades”.

Gustavo Arugete define las redes sociales como: “Formas de interacción social, definida como un intercambio dinámico entre personas, grupos e instituciones en contextos de complejidad. Un sistema abierto y en construcción permanente que involucra a conjuntos que se identifican en las mismas necesidades y problemáticas y que se organizan para potenciar sus recursos” (ARUGUETE, 2001).

Boyd y Ellison definen los sitios de redes sociales, como: “webs basadas en servicios que permiten a los individuos construir un perfil público o semi-público dentro de un sistema delimitado, gestionar una lista de otros usuarios con quienes comparten una conexión, y ver y recorrer su lista de conexiones y las hechas por otros usuarios dentro del sistema. La naturaleza y la nomenclatura de las mismas pueden variar de un sitio de red social a otro” (BOYD y ELLISON, 2007).

“Una red social, a nivel general, es una estructura social formada por nodos, habitualmente individuos u organizaciones que están vinculados por uno o más tipos de interdependencia, tales como valores, puntos de vista, ideas, intercambio financiero, amistad, parentesco, aversión, conflicto, comercio,

enlaces web, relaciones sexuales, Internet con su estructura, transmisión de enfermedades o rutas aéreas. Las estructuras resultantes a menudo son muy complejas” (GONZÁLEZ, 2008).

Luego de analizar las definiciones anteriores se determina que una red social es una estructura social donde cada persona tiene un espacio donde compartir conocimientos, intercambiar ideas sin importar sexo, país o religión. También ofrecen la posibilidad a los usuarios de la red de establecer relaciones de amistad por algún criterio en común ya sea parentesco, afición, intereses o preferencias.

---

### 1.2.1 Clasificación de redes sociales

Según Pablo Fernández Burgueño las redes se clasifican en dos tipos (BURGUEÑO, 2009):

- ✓ Analógicas o Redes sociales Off-Line: son aquellas en las que las relaciones sociales, con independencia de su origen, se desarrollan sin mediación de aparatos o sistemas electrónicos.
- ✓ Digitales o Redes sociales On-Line: son aquellas que tienen su origen y se desarrollan a través de medios electrónicos.

Y estas a su vez se clasifican:

Por su público, objetivo y temática:

- ✓ Horizontales: son aquellas dirigidas a todo tipo de usuario y sin una temática definida. Se basan en una estructura de celdillas permitiendo la entrada y participación libre y genérica sin un fin definido. Los ejemplos más representativos del sector son Facebook, Orkut, Twitter.
- ✓ Verticales: están concebidas sobre la base de un eje temático agregado. Su objetivo es el de congregar en torno a una temática definida a un colectivo concreto. En función de su especialización, pueden clasificarse a su vez en:
  - ❖ Profesionales: están dirigidas a generar relaciones profesionales entre los usuarios. Los ejemplos más representativos son Viadeo, Xing y Linked In.
  - ❖ De Ocio: su objetivo es congregar a colectivos que desarrollan actividades de ocio, deporte, usuarios de videojuegos, fans, etc. Los ejemplos más representativos son Wipley, MinubeDogster, Last.FM y Moterus.

- ❖ Mixtas: ofrecen a usuarios y empresas un entorno específico para desarrollar actividades tanto profesionales como personales en torno a sus perfiles: Yuglo, Unience, PideCita.

Por el sujeto principal de la relación:

- ✓ Humanas: son aquellas que centran su atención en fomentar las relaciones entre personas uniendo individuos según su perfil social y en función de sus gustos, aficiones, lugares de trabajo, viajes y actividades. Ejemplos de este tipo de redes los encontramos en Koornk, Dopplr, Youare y Tuenti.
- ✓ De Contenidos: las relaciones se desarrollan uniendo perfiles a través de contenido publicado, los objetos que posee el usuario o los archivos que se encuentran en su ordenador. Los ejemplos más significativos son Scribd, Flickr, Bebo, Friendster, Dipity, StumbleUpon y FileRide.
- ✓ De Objetos: conforman un sector novedoso entre las redes sociales. Su objeto es unir marcas, automóviles y lugares. Entre estas redes sociales destacan las de “difuntos”, siendo éstos los sujetos principales de la red. El ejemplo más llamativo es Respectance.

Por su localización geográfica:

- ✓ Sedentarias: este tipo de red social muta en función de las relaciones entre personas, los contenidos compartidos o los eventos creados. Ejemplos de este tipo de redes son: Blogger, Kwippy, Plaxo, Bitacoras.com, Plurk.
- ✓ Nómadas: a las características propias de las redes sociales sedentarias se le suma un nuevo factor de mutación o desarrollo basado en la localización geográfica del sujeto. Este tipo de redes se componen y recomponen a tenor de los sujetos que se hallen geográficamente cerca del lugar en el que se encuentra el usuario, los lugares que haya visitado o aquellos a los que tenga previsto acudir. Los ejemplos más destacados son: Foursquare, Gowalla, Latitude, Brighkite, FireEagle y Skout.

Por su plataforma:

- ✓ MMORPG y metaversos: normalmente construidos sobre una base técnica Cliente-Servidor (WOW, SecondLife, Lineage).

- ✓ Web: su plataforma de desarrollo se basa en una estructura típica de web. Algunos ejemplos representativos son MySpace, Friendfeed y Hi5.

### 1.2.2 Redes sociales universitarias

Las redes sociales han supuesto una revolución, un cambio profundo e irreversible en muchos de los ámbitos de nuestras vidas. Cambios que abarcan, desde el más mínimo detalle de nuestra vida personal, como las relaciones personales de carácter privado, hasta la forma como desarrollamos nuestras actividades profesionales. Pasando por supuesto por el cambio que el uso de las redes sociales ha comportado en la enseñanza, en los modelos educativos, y por supuesto en las universidades (ÁLVAREZ, 2011).

Las redes sociales se han enmarcado en diferentes esferas de la sociedad hasta llegar a aplicarse en la educación. La enseñanza superior ha tomado a las redes sociales como un elemento de apoyo en el desarrollo de las funciones primarias. La formación e investigación se ven potenciadas con la creación de redes sociales universitarias. Estas, proveen una red de estudio-aprendizaje con el objetivo de brindar oportunidades de enseñanza en un entorno colaborativo, un punto clave en el aprendizaje en red es que los usuarios de la comunidad universitaria puedan establecer conexiones entre ellos mismos de forma tal que con el intercambio de ideas, opiniones, informaciones, enlaces significativos, se enriquezcan de todo ese conocimiento aportado por todos los miembros de la comunidad.

Es necesario insertar las redes sociales en las universidades y al mismo tiempo proporcionar a los usuarios de la mismas beneficios tales como; disponer de *blogs* personales, compartir conocimientos adquiridos con amistades de la comunidad universitaria, posibilita la integración de servicios como correo electrónico, mensajería instantánea, organizar, divulgar eventos de manera que potencie la gestión de conocimiento, entendido esto como la transferencia de conocimiento y experiencia que pueda ser utilizada como un recurso disponible para otras organizaciones.

### 1.3 Perfiles de usuario

La dirección de la Universidad de las Ciencias Informáticas se ha propuesto el desarrollo de una red social que permita la comunicación e intercambio de información entre los miembros de la comunidad, los cuales se relacionan en dependencia de sus gustos, intereses o preferencias. Teniendo en cuenta

esto se hace urgente la implementación de un servicio de perfil de usuario para dicha red, que permita la gestión de la información asociada a los usuarios de la misma, estableciendo además relaciones de corte profesional o académica, de forma tal que se fomente el desarrollo de las investigaciones y el intercambio de conocimientos dentro de la comunidad. Por tal motivo se realiza un estudio detallado de los perfiles de usuario en las redes sociales.

Perfil de usuario (SALAZAR, 1993):

El perfil del usuario puede ser definido como el conjunto de rasgos distintivos que caracterizan al usuario. También constituye el elemento básico para diseñar servicios de información. Para precisar los perfiles de usuarios se debe seguir un proceso denominado “Estudios de Usuarios”, donde se enfatiza la importancia de los perfiles como fundamento para planear cualquier servicio, unidad o sistema de información.

Estudios de Usuarios:

Los estudios de usuarios son un conjunto de técnicas de investigación social que sirven para determinar el comportamiento de los usuarios de información. El objetivo práctico general de un estudio de usuarios es suministrar las bases para crear, planear o transformar un servicio, unidad o sistema de información.

Así como en el área psicológica, los perfiles sirven para establecer el diagnóstico de una persona, en el área de la bibliotecología sirven para describir a un usuario, y en ambas disciplinas son utilizadas como base para planear cualquier procedimiento a seguir.

Los datos que conforman un perfil de usuario son:

- ✓ Disciplina de interés (necesidades de información),
- ✓ nivel de escolaridad,
- ✓ función o actividad principal: ejemplo administrador, científico, investigador y alumno,
- ✓ recursos de información utilizados o requeridos: fuentes primarias o secundarias,
- ✓ métodos para localizar la información: ejemplo citas, uso de la biblioteca, colegio invisible y documentación personal,
- ✓ comportamiento en la búsqueda de información: búsqueda personal o delegada,

- ✓ manejo del lenguaje.

### 1.3.1 Estado del arte de los perfiles de usuario en redes sociales

Con el fin de estudiar como se comportan los perfiles de usuario en las redes sociales, se analizó una muestra de las 12 redes sociales más populares en el mundo (ver Tabla 1.1), tomando como base los datos que predominan en los perfiles de usuario de cada una de estas y realizando una comparación.

Las características a tener en cuenta para realizar la comparación son las siguientes:

- ✓ Información básica (IB),
- ✓ imagen del perfil (IP),
- ✓ formación académica (FA),
- ✓ información laboral (IL),
- ✓ información de contacto (IC),
- ✓ intereses (I),
- ✓ nivel de acceso (NA),
- ✓ nivel de usabilidad (NU),
- ✓ servicios adicionales (SA),
- ✓ referencias a otras redes sociales (RRS).

Las redes sociales seleccionadas para el estudio son:

- ✓ Facebook
- ✓ MySpace
- ✓ Google +
- ✓ Twitter
- ✓ LinkedIn
- ✓ Tagged
- ✓ Twoo
- ✓ Netlog
- ✓ Badoo
- ✓ ResearchGate

- ✓ Campusbug
- ✓ Universidad de Vigo.

Tabla 1.1 Resultados de la comparación entre redes sociales.

Características / Redes sociales		IB (sí/no)	IP (sí/no)	FA (sí/no)	IL (sí/no)	IC (sí/no)	I (sí/no)	NA (sí/no)	RRS (sí/no)
<b>Redes sociales generales</b>	Facebook	sí	sí	sí	sí	sí	sí	sí	sí
	MySpace	sí	sí	sí	sí	no	sí	sí	sí
	Google+	sí	sí	sí	sí	sí	sí	sí	sí
	Twitter	sí	sí	no	no	no	no	no	no
	Linkedin	sí	sí	sí	sí	sí	sí	sí	sí
	Tagged	sí	sí	sí	no	no	sí	sí	no
	Twoo	sí	sí	sí	no	no	sí	no	sí
	Netlog	sí	sí	no	sí	sí	sí	sí	no
<b>Redes sociales universitarias</b>	Badoo	sí	sí	sí	sí	no	sí	sí	sí
	ResearchGate	sí	sí	sí	no	sí	sí	sí	sí
	Campusbug	sí	sí	sí	sí	no	sí	no	sí
	Universidad de Vigo	sí	sí	sí	sí	sí	sí	sí	no

A partir de la tabla anterior se definen cuáles son las características más relevantes (una característica se considera relevante cuando está presente en más de la mitad de las redes sociales estudiadas) en los perfiles de las diferentes redes sociales que se analizan. Como resultados del estudio las características más relevantes son: la información básica (Nombre, edad, sexo, fecha de nacimiento, etc.), la imagen de perfil, los intereses, la formación académica (curso, año de matrícula, grupo, facultad, etc.) y el nivel de acceso a la información (privado, amigos, usuarios logueado, todos); y las otras características que no resultaron relevantes en el estudio son: la información laboral (especialidad, categoría académica, cargo que ocupa, etc.), la información de contacto (teléfono, correo electrónico) y la referencia a otras redes sociales.

La selección de los atributos que se incluyen en el servicio: perfil de usuario de la RSU, se enfoca en la extracción de las características que son relevantes a partir del estudio de las redes sociales realizado anteriormente y otros criterios que aunque no tienen un nivel de significancia alto son de importancia para la universidad. Las características que se incluyen en el perfil de usuario son: la información básica, la imagen de perfil, la formación académica, intereses, la información laboral, el nivel de acceso a la información y la información de contacto.

#### 1.4 Redes sociales y web semántica

Tomando la idea de Berners-Lee donde señala que: “Deberíamos ser capaces no sólo de encontrar cualquier tipo de documento en la Web, sino también de crear cualquier clase de documento fácilmente. Deberíamos no sólo poder interactuar con otras personas, sino crear con otras personas. La intercreatividad es el proceso de hacer cosas o resolver problemas juntos” (BERNERS-LEE, 2000), estas son las ideas fundamentales acerca de la web social, término que fue acuñado por (REILLY, 2005), principal promotor de esta ideología.

“La aplicación de la Web Semántica a la Web Social se está transformando en la Web Semántica Social, creando una red de conocimientos entrelazados y rica de semántica. Esta visión de la Web constará de documentos interrelacionados, datos, e incluso aplicaciones creadas por los propios usuarios finales como resultado de diversas interacciones sociales, y se describe utilizando formatos legibles por las máquinas, para que pueda ser usado para los propósitos que en el actual estado de la Web Social no se puede lograr sin dificultad.” (DOMINGUE *et al.*, 2011).

Según Berners-Lee “la Web Semántica es una extensión de la Web actual en la que la información tiene un significado bien definido que permite a personas y ordenadores trabajar mejor de forma conjunta.” (CORCHUELO, 2007).

Hendler y Golbeck plantean que “las tecnologías de la Web Semántica pueden ser usadas para crear puentes entre la información social de los usuarios. De esta manera, cuanto más personas interactúan entre sí, más conocimiento queda interconectado a escala mundial.” (DOMINGUE *et al.*, 2011).

La dirección de la Universidad de las Ciencias Informáticas, se ha propuesto el desarrollo de una Red Social Universitaria basándose en una estrategia de integración semántica, con el objetivo de aprovechar sus ventajas. Es por esto que, se hace necesario realizar un estudio de los estándares

para publicar la información de los perfiles de manera semántica. Por tal motivo es que se analizan ontologías para la descripción de personas que existen en el mundo y de estas seleccionar la que más se adecúe a las necesidades de la universidad.

Gruber menciona que una “Ontología es una especificación formal y explícita de una conceptualización compartida”. (GRUBER, 1993), se dice que debe ser formal por que debe ser entendible por un computador, explícita porque, se refiere a su descripción en términos de conceptos, atributos y relaciones, y conceptualización debido a que es un modelo abstracto de un fenómeno del mundo que se desea representar, además Gruber menciona conceptos (ideas básicas que se intentan formalizar, describen y representan conceptos de un dominio específico), relaciones (representan la interacción y enlace entre los conceptos del dominio), atributos (expresan cualidades de un concepto), funciones (son relaciones donde se identifica un elemento mediante el cálculo de una función que considera varios elementos de la ontología.), instancias (utilizadas para representar objetos determinados de un concepto) y axiomas (teoremas que se declaran sobre relaciones que deben cumplir los elementos de la ontología) como los componentes de una ontología para representar el conocimiento (GRUBER, 1993). Además una ontología se compone de una taxonomía<sup>3</sup> y un conjunto de reglas de inferencia para un dominio específico de conocimiento. Una taxonomía define clases de objetos y relaciones entre ellos y una regla de inferencia es una condición sobre las relaciones entre los objetos (ROMERO, 2011).

El concepto de ontología se basa en la descripción del mundo real, similar a la programación orientada a objeto, por lo que nos permite representarlo por medio de clases, subclasses, propiedades y además nos permite construir relaciones entre ellos además del uso de reglas a través del cual puedan interactuar y funcionar (HUAMANI, 2011).

Algunas de las ontologías que se utilizan para la descripción de personas son:

- ✓ FOAF
- ✓ BIO

---

<sup>3</sup> Es la ciencia en la que se clasifican los organismos y se establecen parámetros de diferencia, creando familias, ramas y conjuntos de razas.

#### 1.4.1 FOAF (Friend of a Friend)

FOAF (*Friend of a Friend*) es un proyecto dentro del entorno de la Web Semántica para describir personas, vínculos entre ellas, y cosas que hacen o crean. Se trata de un vocabulario que permite disponer de información personal de forma sencilla y simplificada para ser procesada, compartida y reutilizada. Es un lenguaje de definición basado en RDF<sup>4</sup>, utilizado para la definición de datos sobre redes sociales. De esta forma se pueden definir los datos de una persona ejemplo: nombre, apellidos, dirección, correo, intereses, personas que conoce y páginas que le gustan (RUIZ, 2008).

Es una tecnología simple que hace que sea más fácil de compartir y utilizar información sobre las personas y sus actividades (por ejemplo: fotos, calendarios, blogs), para transferir información entre los sitios Web, y para extender de forma automática, combinar y reutilizar en línea.

La idea básica detrás de FOAF es simple: es realizar las conexiones entre las cosas en la red. FOAF ofrece algún mecanismo básico que nos ayude a decirle a la Web sobre las conexiones entre las cosas que nos importan. Miles de personas ya lo hacen en la Web mediante la descripción de sí mismos y sus vidas en su página principal. Utilizando FOAF, puede ayudar a las máquinas entender tu página de inicio, aprender acerca de las relaciones que conectan a personas, lugares y cosas que se describen en la Web. FOAF utiliza la tecnología del W3C<sup>5</sup> RDF para integrar la información de su página de inicio con el de tus amigos, y los amigos de tus amigos, y sus amigos (CARVALHO, 2010).

Dentro de las facilidades que brinda FOAF se encuentra la compartición de información sobre personas, también proporciona el acceso y recuperación de información sobre las mismas de forma automatizada mediante búsquedas. Gracias al uso de metadatos<sup>6</sup> y XML<sup>7</sup> se pueden crear directorios de personas, listas de correo, entre otras cosas (RUIZ, 2008).

---

<sup>4</sup> El *Resource Description Framework* (RDF) es un marco de trabajo para representar la información en la Web. KLYNE, G. y CARROLL, J. J. *Resource Description Framework (RDF)* [Consultado el: 09 de febrero de 2012]. Disponible en: <http://www.w3.org/TR/rdf-concepts/>.

<sup>5</sup> Consorcio de la W3C

<sup>6</sup> Los metadatos se definen como datos sobre datos, es decir, descripciones de características y propiedades sobre datos. Estas descripciones son de gran ayuda en la catalogación y recuperación de información.

<sup>7</sup> XML (en español lenguaje extensible de marcado), originalmente diseñado para satisfacer los desafíos de la gran escala de la publicación electrónica.



incluyendo obviamente el nacimiento, matrimonio y la muerte, pero también la coronación, el desarrollo e incluso el asesinato. Estos eventos no tienen la intención de ser exhaustivo, pero son representativos de los tipos de eventos relacionados con el material biográfico (DAVIS y GALBRAITH, 2011) .

Se determina la selección del estándar FOAF por ser uno de los más aplicados en la integración de las redes sociales, es además la ontología para describir personas más usadas a nivel mundial, por su flexibilidad y adaptabilidad a las necesidades del servicio: perfil de usuario de la RSU.

## 1.5 Plataformas para desarrollar redes sociales

Se analizan algunas herramientas para la construcción de redes sociales, en busca de una que se ajuste a las necesidades para el desarrollo de un servicio: perfil de usuario para la red social de la universidad.

### 1.5.1 Mahara

Mahara es un sistema autónomo que se puede integrar en un marco general de aprendizaje virtual, se encuentra bajo la Licencia Pública General de GNU/Linux. La arquitectura de Mahara está basada en la arquitectura modular y extensible de Moodle, es una herramienta para crear redes sociales de aprendizaje en línea.

Estructura ((TIC), 2009). (Ver Figura 1.2):

- ✓ Artefactos: Son los distintos elementos que el usuario tiene para trabajar con la misma.
- ✓ Vistas: Una forma de mostrar los artefactos de la manera que los usuarios elijan, a las personas que ellos deseen.
- ✓ Grupos: Grupos sociales entorno a inquietudes comunes.

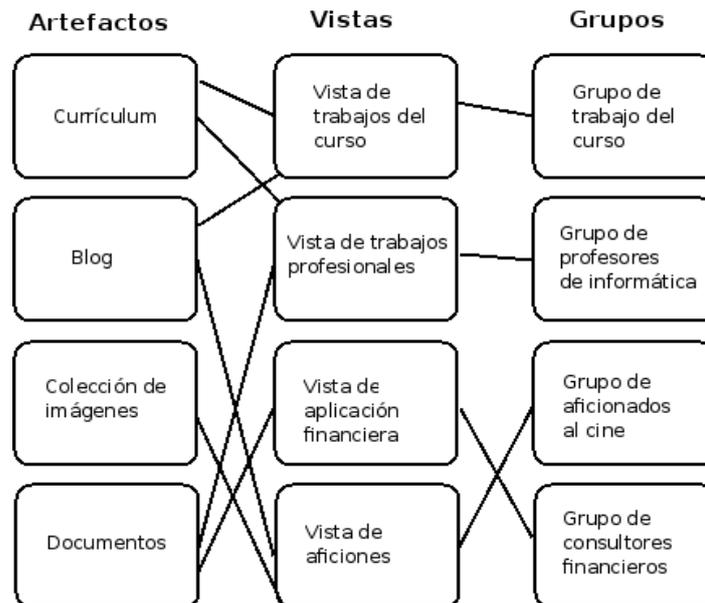


Figura 1.2 Estructura de Mahara

**Características:**

- ✓ Repositorio de ficheros: esta característica le permite a los usuarios estructurar los ficheros en carpetas, subir múltiples ficheros, además puede descomprimir ficheros comprimidos en formatos .zip .tar.gz y .tar.bz2.
- ✓ Bitácoras: Mahara incluye una herramienta para creación de bitácoras personales por parte de los usuarios.
- ✓ Red Social: los usuarios pueden crear listas de amigos y opcionalmente integrarse al portafolio electrónico.
- ✓ Constructor de currículos: esta funcionalidad permite la creación de currículos digitales de los usuarios, que contenga información de contacto, su historial educacional y de trabajo, las certificaciones, premios, acreditaciones, habilidades académicas y ocupaciones laborales, publicaciones, libros y asociaciones profesionales.
- ✓ Información de perfil: permite a los usuarios publicar información en su perfil relacionada con sus datos personales, cuentas de servicios de redes sociales, información de contacto y otros.

- ✓ Personalización: el sistema Mahara permite a los administradores configurar el lenguaje, las plantillas de diseño, los métodos de autenticación, administrar las instituciones, los menús del sistema y las opciones de las cuentas de los usuarios.

### 1.5.2 Elgg

Elgg es un motor de código abierto que proporciona un marco sólido para construir todo tipo de entornos sociales. Se distribuye bajo la Licencia Pública General de GNU, es gratis y su código fuente está disponible gratuitamente a través de las versiones estables. Una vez que se instale y el usuario se registre se pueden crear blogs, hacer comunidades, recolectar noticias mediante agregación RSS y compartir ficheros. Es una plataforma flexible y fácil de usar, cuenta con *plugins*<sup>8</sup> que sirven para extender sus capacidades sociales, además de contar con un API lo que técnicamente es una ventaja importante, ya que se puede a través de esta desarrollar aplicaciones que se adapten a necesidades específicas. Fue creada con la idea de ser un sistema que administre un ambiente social en un colegio, una universidad o incluso un ambiente colaborativo interno dentro de las empresas.

Algunos de los sitios que usan Elgg como plataforma social son:

- ✓ <http://www.blogotronix.com/>
- ✓ <http://www.budokin.com/>
- ✓ <http://www.thehobbyspace.com/>
- ✓ <http://www.scispace.com/>

Características (ELGG, 2011):

- ✓ Modelo de datos potente: Elgg proporciona un potente modelo de datos permitiendo la creación de entidades diferentes sencillas, pero flexibles (ver Figura 1.3).
- ✓ Flujos de actividad: la API de flujo de actividad granular asegura que sus *plugins* pueden empujar el contenido necesario para los usuarios.
- ✓ Extensible: El uso de este *plugin* permite crear y añadir características requeridas.

---

<sup>8</sup> En computación, un plug-in (o plugin) es un conjunto de componentes de software que añade capacidades específicas para una aplicación de software más grande. Los plugins permiten la personalización de la funcionalidad en una aplicación.

- ✓ Gestión de usuarios: Elgg se encarga de la administración de usuarios y los requisitos de la relación entre ellos.
- ✓ Control de acceso: todos los objetos de Elgg pueden tener un nivel de control de acceso, que da los permisos de acceso granular posible.
- ✓ API de servicios web: su objetivo es exponer funcionalidad de la API a través del estilo arquitectónico REST<sup>9</sup>.
- ✓ Permite importar y exportar amigos con la ontología FOAF.

Modelo de datos:

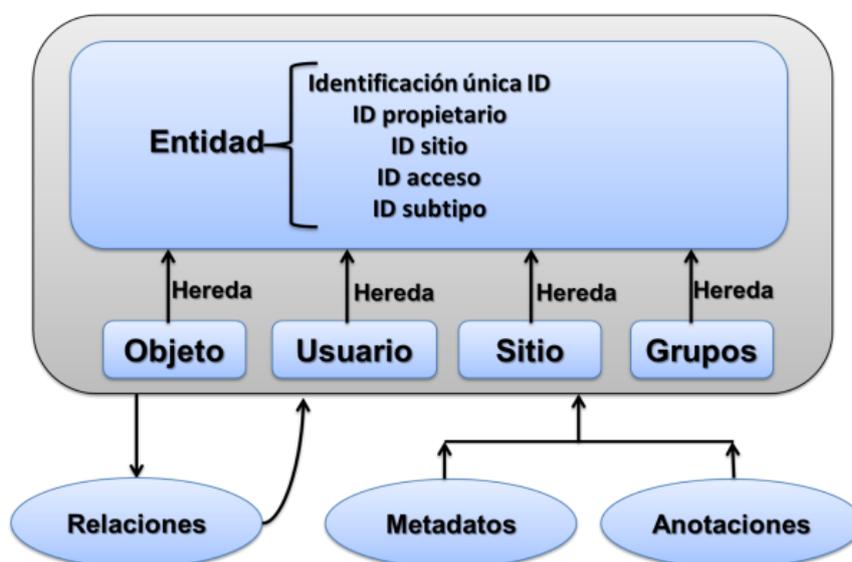


Figura 1.3 Modelo de datos del Elgg (ELGG, 2011)

A continuación se describen las clases contenidas en el modelo de datos.

- ✓ Entidad: representa cualquier elemento de la plataforma, además de controlar permisos y propiedad.
- ✓ Objeto: en la red social cualquier otro tipo de contenido que se genera es mediante un objeto. Ejemplo de objetos son blogs, archivos y marcadores.
- ✓ Usuarios: representa cada usuario del sistema.

<sup>9</sup> La Transferencia de Estado Representacional o REST (Representational State Transfer) es un estilo de arquitectura de software para sistemas hipermédias distribuidos tales como la Web.

- ✓ Sitio: es cada sitio web creado en una instalación Elgg.
- ✓ Grupos: es una entidad que contiene/agrupa otras entidades. Un grupo está compuesto, por miembros que son los usuarios. Puede contener diferentes entidades, como ser blogs, archivos, foros, que son creados dentro del grupo.

Estas entidades se pueden extender para agregar información a las mismas de 2 maneras:

- ✓ Metadatos: son propiedades que se pueden asignar a una entidad. Añadiendo información adicional a la misma. Por ejemplo, en el caso de un usuario, puede ser fecha de nacimiento o nacionalidad.
- ✓ Anotaciones: son utilizadas para realizar un *feedback*<sup>10</sup> sobre una entidad. Por ejemplo, si hablamos de una entrada de blog, los comentarios y *ratings* al blog, son anotaciones.
- ✓ Relaciones: las entidades también pueden relacionarse entre sí y las relaciones son direccionales. Para crear una relación requiere un nombre de relación y dos entidades.

---

### 1.5.3 Socialengine

SocialEngine es una plataforma basada en PHP, es privativo, proporciona aspectos similares a la de una red social en la página web de un usuario. Las características principales incluyen la administración de pequeñas y medianas redes sociales a escala, algunas capacidades de personalización sin encriptar el código, y compatibilidad con *widjets*. Hay una amplia gama de plantillas y *add-ons* disponibles para ampliar las características básicas que ya están incluidos en el núcleo.

Características (SOCIAENGINE, 2012):

- ✓ Perfiles de los miembros: la página de perfiles de los miembros proporciona a cada integrante un hogar para su contenido, actividad e información personal. Además de que brinda la posibilidad de tener perfiles de múltiples partes, tipos de perfiles, opciones de privacidad, mensajes, así como la personalización del diseño. Estos contienen varias secciones de contenidos que muestran la información de perfil, junto con fotos de un miembro, entradas de blog, comentarios, grupos, amigos, estadísticas, y otros artículos que usted decida.

---

<sup>10</sup> (En español retroalimentación) es el proceso mediante el cual se realiza un intercambio de datos con informaciones, hipótesis o teorías entre dos puntos diferentes.

- ✓ Estructura de la red: SocialEngine brinda muchas opciones para decidir como las personas se unirán, como van a interactuar y de qué forma utilizarán su comunidad. También provee niveles de privacidad personalizada, cuenta con una proceso de registro personalizable, el ingreso es por invitación de los miembros o por invitación de administración.
- ✓ Gestión de contenidos: brinda la posibilidad de diseñar el sitio a su gusto, el estilo y contenido pueden ser totalmente personalizado, la organización de las áreas se pueden reorganizar sin tener que escribir ningún código. Permite la creación de páginas nuevas muy fácil, selecciona cuales áreas de su comunidad están disponibles para el público y cuales requieren una cuenta. Presenta una opción de archivos y administrador de multimedia que permite cargar y almacenar los archivos para su uso en el diseño, en los anuncios o cualquier otro contenido. Posee URL amigables y metadatos personalizados que hace más fácil la optimización de búsqueda, además ofrece un soporte multi-idioma que importa paquetes de idiomas que no están en inglés o crea sus propias traducciones.(SOCIALENGINE, 2012)

---

#### 1.5.4 Aroundme

AROUNDme provee todo lo necesario para crear una red social. Es una plataforma libre de código abierto que se puede instalar libremente en un servidor y se montan redes sociales del tipo MySpace.

Características (BLOCLY, 2008):

- ✓ Está implementado en PHP.
- ✓ Soporta MySQL, PostgreSQL, Interbase, Firebird, Oracle, Informix, Access, ADO y FrontBase, SQLite.
- ✓ Es multilinguaje, para publicación y administración.
- ✓ Se pueden formar grupos en función de gustos y especialidades.
- ✓ Cada persona que se registra puede tener su propio blog.
- ✓ Se puede tener una zona donde dejar documentos, manuales, guías, etc.
- ✓ Permite organizar eventos y actividades e invitar a tus amigos.
- ✓ Se basa en 3 herramientas: Identidad personal (tu propia página), Identidad del servidor (conexión entre las identidades personales) y servidor de colaboración (compartir entre varios, visibilidad).

El uso del servidor de colaboración AROUNDme permite crear grupos de colaboración múltiple, espacios web, comunidad o sitios web de redes sociales. Cada grupo puede crear un sitio web colaborativo de varias páginas. Reciben herramientas sociales como un libro de visitas (una pared), un blog de grupo, un foro y una wiki que pueden caer en las páginas web. Cada grupo es totalmente personalizable utilizando XHTML, CSS, Javascript y PHP. Los grupos pueden ser públicos o privados.

Luego del estudio realizado se selecciona la herramienta Elgg, ya que proporciona un modelo de datos potente, el API de servicios web, su flexibilidad y fácil manejo, posibilita el trabajo con la web semántica entre otras que también la hacen escogerla a partir del análisis de las otras herramientas.

## 1.6 Tecnologías para el desarrollo

Para el desarrollo del servicio: perfil de usuario se definen las siguientes tecnologías:

### 1.6.1 Proceso para el desarrollo de software con un enfoque ágil y CMMI nivel 2

#### **Proceso de desarrollo de software**

Un proceso de desarrollo de software tiene como propósito la producción eficaz y eficiente de un producto software que reúna los requisitos del cliente (LETELIER, 2003).

Pressman establece un marco común del proceso definiendo un pequeño número de actividades del marco de trabajo que son aplicables a todos los proyectos del software, con independencia de su tamaño o complejidad (PRESSMAN, 2005).

#### **Enfoque ágil:**

El desarrollo ágil de software es un marco de trabajo conceptual de la ingeniería de software que promueve iteraciones en el desarrollo a lo largo de todo el ciclo de vida del proyecto. Existen muchos métodos de desarrollo ágil; la mayoría minimiza riesgos desarrollando software en cortos plazos de tiempo. El software desarrollado en una unidad de tiempo es llamado una iteración, la cual debe durar de una a cuatro semanas. Cada iteración del ciclo de vida incluye: planificación, análisis de requerimientos, diseño, codificación, revisión y documentación. Al final de cada iteración el equipo vuelve a evaluar las prioridades del proyecto.

**CMMI (*Capability Maturity Model Integration*)**

El SEI (*Software Engineering Institute*) es el instituto que creó y mantiene el modelo de calidad CMM – CMMI, el CMM - CMMI es un modelo de calidad del software que clasifica las empresas en niveles de madurez. Estos niveles sirven para conocer la madurez de los procesos que se realizan para producir software (PALACIO, 2005).

Integración de modelos de madurez de capacidades es un enfoque de mejora de procesos que proporciona a las organizaciones los elementos esenciales de procesos efectivos, lo que mejora su rendimiento. El proceso basado en CMMI, incluye la identificación de las fortalezas y debilidades de los procesos de la organización y realiza cambios en el proceso para convertir debilidades en fortalezas.

Este modelo cuenta con cinco niveles (GRACIA, 2005):

- ✓ Inicial o Nivel 1 CMM - CMMI. Este es el nivel en donde están todas las empresas que no tienen procesos. No hay control sobre el estado del proyecto, el desarrollo del mismo es completamente opaco, no sabes lo que pasa en él.
- ✓ Repetible o Nivel 2 CMM – CMMI: la principal diferencia entre este nivel y el anterior es que el proyecto es gestionado y controlado durante el desarrollo del mismo. El desarrollo no es opaco y se puede saber el estado del proyecto en todo momento.
- ✓ Definido o Nivel 3 CMM – CMMI: alcanzar este nivel significa que la forma de desarrollar proyectos (gestión e ingeniería) está definida, quiere decir que está establecida, documentada y que existen métricas (obtención de datos objetivos) para la consecución de objetivos concretos.
- ✓ Cuantitativamente Gestionado o Nivel 4 CMM – CMMI: los proyectos usan objetivos medibles para alcanzar las necesidades de los clientes y la organización. Se usan métricas para gestionar la organización.
- ✓ Optimizado o Nivel 5 CMM – CMMI: los procesos de los proyectos y de la organización están orientados a la mejora de las actividades. Mejoras incrementales e innovadoras de los procesos que mediante métricas son identificadas, evaluadas y puestas en práctica.

La UCI actualmente se halla inmersa en un proceso de mejora de los procesos que se desarrollan como parte de la construcción del software basado en el modelo de CMMI. Este proceso de mejora tiene como objetivo que la universidad alcance el nivel dos del modelo CMMI, lo que la convierte en la

primera institución del país en alcanzar este nivel. El objetivo es ayudar la UCI a establecer las bases y fundamentos para seguir mejorando sus procesos y fortalecer su cultura de calidad en el desarrollo de software. Alinear los procesos de desarrollo de software con los principios y requisitos del modelo CMMI, estableciendo planes de mejora con los que la organización oriente sus procesos hacia la consecución de sus metas (CALISOFT, 2009).

---

### 1.6.2 ELGG

Luego de seleccionar la herramienta Elgg para el desarrollo de la aplicación se requiere de las siguientes tecnologías para su instalación (ELGG, 2011):

- ✓ lenguaje de programación PHP en la versión 5.2 o superior
- ✓ servidor de base de datos MySQL en la versión 5 o superior.

Seguidamente se analizan las características que se incluyen para el desarrollo del sistema, además de las requeridas por la plataforma.

---

### 1.6.3 PHP 5.3.10

PHP (acrónimo de *PHP: Hypertext Preprocessor*) es un lenguaje de código abierto muy popular especialmente adecuado para desarrollo web y que puede ser incrustado en HTML. Lenguaje de programación interpretado, diseñado originalmente para la creación de página web dinámicas. Es usado principalmente en interpretación del lado del servidor, pero actualmente puede ser utilizado desde una interfaz de línea de comandos o en la creación de otros tipos de programas incluyendo aplicaciones con interfaz gráfica. Generalmente se ejecuta en un servidor web, tomando el código en PHP como su entrada y creando páginas web como salida (PHP, 2012).

---

### 1.6.4 HTML 4.01

El HTML (*Hyper Text Markup Language*) es el lenguaje con el que se escriben las páginas web. Es un lenguaje de hipertexto, es decir, un lenguaje que permite escribir texto de forma estructurada, y que está compuesto por etiquetas, que marcan el inicio y el fin de cada elemento del documento. Un documento hipertexto no sólo se compone de texto, puede contener imagen, sonido, video, etc., por lo que el resultado puede considerarse como un documento multimedia. Los navegadores se encargan

de interpretar el código HTML de los documentos, y de mostrar a los usuarios las páginas web resultantes del código interpretado (W3C, 2012b).

---

#### 1.6.5 JavaScript

JavaScript es un lenguaje de programación interpretado, dialecto del estándar *ECMAScript*. JavaScript es un lenguaje de script multiplataforma orientado a objetos. Es un lenguaje pequeño y ligero; no es útil como un lenguaje independiente, más bien está diseñado para una fácil incrustación en otros productos y aplicaciones, tales como los navegadores web.

JavaScript es un lenguaje en un formato mucho más liberado. No tiene que declarar todas las variables, clases y métodos. No debe preocuparse si sus métodos son públicos, privados o protegidos y no tiene que implementar sus interfaces. Los tipos de variables, parámetros y funciones de retorno no son explícitamente definidos.

El núcleo de JavaScript contiene un conjunto central de objetos, tales como *Array* (arreglos), *Date* (fechas) y *Math* (objetos matemáticos), además de un conjunto central de elementos del lenguaje tales como los operadores, estructuras de control y sentencias (MOZILLA, 2012).

---

#### 1.6.6 CSS 2.1

Hojas de Estilo en Cascada (*Cascading Style Sheets*), es un mecanismo simple que describe cómo se va a mostrar un documento en la pantalla, o cómo se va a imprimir, o incluso cómo va a ser pronunciada la información presente en ese documento a través de un dispositivo de lectura. Esta forma de descripción de estilos ofrece a los desarrolladores el control total sobre estilo y formato de sus documentos. CSS se utiliza para dar estilo a documentos HTML y XML, separando el contenido de la presentación. Los Estilos definen la forma de mostrar los elementos HTML y XML. CSS permite a los desarrolladores web controlar el estilo y el formato de múltiples páginas web al mismo tiempo. CSS es independiente de HTML y se puede utilizar con cualquier lenguaje de marcado basado en XML (W3C, 2012a).

---

### 1.6.7 Servidor de base de datos MySQL 5.1

Es un sistema de gestión de bases de datos relacional<sup>11</sup>. Una base de datos relacional es un conjunto de datos que están almacenados en tablas entre las cuales se establecen unas relaciones para manejar los datos de una forma eficiente y segura. Este gestor de bases de datos es, probablemente, el gestor más usado en el mundo del software libre, debido a su gran rapidez y facilidad de uso. Esta gran aceptación es debida, en parte, a que existen infinidad de librerías y otras herramientas que permiten su uso a través de gran cantidad de lenguajes de programación, además de su fácil instalación y configuración. Para usar y gestionar una base de datos relacional se usa el lenguaje estándar de programación SQL. Además provee altas características de seguridad, contiene un sistema de autenticación capaz de aceptar o denegar las conexiones, también una herramienta para administración de los privilegios y métodos de encriptación (MYSQL, 2012).

---

### 1.6.8 Servidor web Apache 2.2

Apache es el servidor web hecho por excelencia, su configurabilidad, robustez y estabilidad hacen que cada vez millones de servidores reiteren su confianza en este programa. Corre en una multitud de Sistemas Operativos, lo que lo hace prácticamente universal, es una tecnología gratuita, de código abierto, es un servidor altamente configurable de diseño modular. Es muy sencillo ampliar las capacidades del servidor web Apache, cualquiera que posea una experiencia decente en la programación de C o Perl puede escribir un módulo para realizar una función determinada. Tiene una alta configurabilidad en la creación y gestión de logs. Apache permite la creación de ficheros de log a medida del administrador, de este modo puedes tener un mayor control sobre lo que sucede en el servidor (APACHE, 2012).

---

### 1.6.9 Visual Paradigm 8.0

Visual Paradigm es una herramienta CASE que utiliza UML como lenguaje de modelado. Presenta la posibilidad de generar código a partir de los diagramas, para plataformas como .Net, Java y PHP, así como permite la obtención de diagramas a partir de código. Visual Paradigm para UML es

---

<sup>11</sup> Una base de datos relacional archiva datos en tablas separadas en vez de colocar todos los datos en un gran archivo. Esto permite velocidad y flexibilidad. Las tablas están conectadas por relaciones definidas que hacen posible combinar datos de diferentes tablas sobre pedido.

multiplataforma, lo cual le permite al usuario utilizar esta herramienta en varios sistemas operativos como Windows, Linux, Unix y otros; además se encuentra disponible en distintas versiones: Enterprise, Professional, Standard, Modeler, Personal y Community (PARADIGM, 2011).

---

#### 1.6.10 Netbeans 7.0.1

Es un proyecto de código abierto de gran éxito, con una gran comunidad en constante crecimiento. La plataforma Netbeans permite que las aplicaciones sean desarrolladas a partir de un conjunto de componentes de software llamados módulos por lo que pueden ser extendidas fácilmente por otros desarrolladores de software. Netbeans IDE es una aplicación diseñada para el desarrollo de aplicaciones fácilmente portables entre las distintas plataformas. Es una herramienta para que los programadores puedan escribir, compilar, depurar y ejecutar programas. Aunque está escrita en Java puede servir para cualquier otro lenguaje de programación (NETBEANS, 2012).

### 1.7 Conclusiones parciales

El estudio de los perfiles de redes sociales permitió seleccionar las características más relevantes del servicio: perfil de usuario de la RSU. El análisis de las diferentes ontologías para describir personas permitió definir la ontología FOAF para la descripción del perfil de usuario. Además se estableció la herramienta Elgg para el desarrollo del sistema, a raíz del estudio de las plataformas para construir redes sociales.

## 2 Capítulo II: Propuesta de solución

### 2.1 Introducción

En el presente capítulo se abarcan temas relacionados con la propuesta de solución del servicio: perfil de usuario de la RSU para la Universidad de las Ciencias Informáticas y se presentan los principales requerimientos funcionales y no funcionales para el desarrollo del servicio.

### 2.2 Diagnóstico del campo de acción

Los servicios tales como: Biblioteca Virtual, el Entorno Virtual de Aprendizaje (EVA), Inter-nos, Correo Electrónico, el servicio de Mensajería Instantánea, los diferentes portales de las comunidades de desarrollo que potencia la FEU de la institución están destinados a incrementar el desarrollo de las investigaciones y el intercambio de ideas y conocimientos dentro de la universidad. Estos servicios presentan características comunes entre ellos de las cuales se puede mencionar que, la cantidad de información que los usuarios depositan en su interacción con los mismos se encuentra sin ninguna interrelación y en su mayoría de manera disgregada. Además los servicios desarrollados carecen de una estructura común en la forma en que almacenan los datos de los usuarios de manera tal que puedan ser reutilizados por otros servicios, es decir, los sistemas no almacenan los datos de manera estándar.

Es de gran utilidad para la institución contar con la información que aportan los usuarios de la universidad en los sitios de redes sociales en Internet, lo que posibilita identificar sus características y competencias.

Por todo lo antes expuesto se concibe el desarrollo del servicio: perfil de usuario para la RSU de la Universidad de las Ciencias Informáticas que permita gestionar toda la información brindada por los usuarios en los perfiles de la red social. Además brinde sugerencias de posibles relaciones entre los usuarios y servicios a partir de la información que estos ingresan.

### 2.3 Descripción de la propuesta

El desarrollo del servicio: perfil de usuario de la RSU, posibilita el cómo recuperar la información relacionada con los usuarios así como la comunicación entre ellos y el resto de los servicios. Este

servicio provee a los usuarios una serie de funcionalidades tales como, la gestión de los atributos de perfil, y la autenticación en la red social a través del dominio UCI. Esto evita que el usuario tenga que crearse una cuenta antes de poder acceder. También permite establecer relaciones de amistad entre ellos e inferir relaciones de amistad basadas en la información del perfil a través de la integración con la API del núcleo. Por otra parte brinda opciones tales como, enviar mensajes al jabber y el correo mediante la integración con API<sup>12</sup> de los servicios telemáticos además cuenta con la posibilidad de mostrar información sobre la cuota de navegación, obtener la cantidad de correos sin leer, trabajo con el calendario y tareas.

### 2.3.1 Descripción de la red social

En la Figura 2.1 se muestra la forma en la que los servicios de la red social universitaria se integran a través de la API del núcleo, empleando el estilo arquitectónico REST.

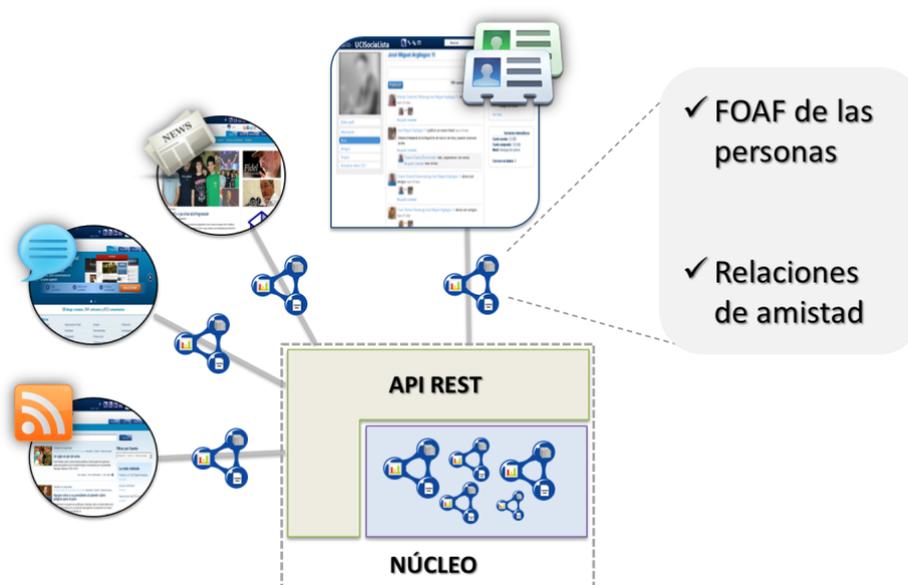


Figura 2.1 Arquitectura de la red social

El servicio: perfil de usuario se comunica con la API del núcleo a través de un componente que se desarrolla con el objetivo de registrar la descripción del perfil de los usuarios que acceden al sistema

<sup>12</sup>Interfaz de programación de aplicaciones o API (*Application Programming Interface*) es el conjunto de funciones y procedimientos que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.

mediante de la ontología FOAF, además muestra las sugerencias de amistad, así como registrar y eliminar las relaciones de amistad.

### 2.3.2 Descripción del perfil de usuario

Del estado del arte realizado a los diferentes perfiles de redes sociales se extraen las características que resultan relevantes en el estudio y que además se incluyen al servicio que se quiere desarrollar. Los atributos que se tienen son: la información básica, la imagen de perfil, la formación académica, intereses y los servicios adicionales y otras que aunque no fueron relevantes pero que tienen gran importancia en la RSU como son: la información laboral y el nivel de acceso a la información.

No solo es suficiente analizar las características de los perfiles de usuarios en las redes sociales en Internet, sino también es importante estudiar los usuarios de la universidad en busca de elementos que sean relevantes para la descripción de la solución.

El servicio: perfil de usuario que se propone se integra por cuatro tipos de perfil: el perfil usuario genérico, el perfil estudiante, perfil estudiante en el ciclo profesional y el perfil profesor (Ver Figura 2.2). A continuación se describen cada uno de estos tipos de perfiles anteriormente expuesto.



Figura 2.2 Propuesta del perfil de usuario

El tipo perfil de usuario contiene los atributos base para el resto de los tipos de perfiles los cuales son:

- ✓ Nombre y apellidos,
- ✓ fecha de nacimiento,
- ✓ número de carne de identidad,
- ✓ dirección particular,
- ✓ provincia,
- ✓ municipio,
- ✓ estado civil,
- ✓ organizaciones políticas,
- ✓ sexo,
- ✓ solapín<sup>13</sup>,
- ✓ intereses,
- ✓ imagen de perfil,
- ✓ correo de la cuenta UCI,
- ✓ nivel de acceso a la información,
- ✓ información de contacto.

En el caso del estudiante además posee atributos tales como:

- ✓ Facultad,
- ✓ grupo,
- ✓ año,
- ✓ año de matrícula.

Adicionalmente el estudiante en el ciclo profesional presenta los atributos siguientes:

- ✓ Centro de desarrollo,
- ✓ línea de desarrollo,
- ✓ proyecto.

Asimismo el perfil del profesor contiene los atributos que se desglosan a continuación:

---

<sup>13</sup> Identificador del expediente de la persona en la UCI.

- ✓ Categoría docente,
- ✓ categoría académica,
- ✓ departamento a que pertenece,
- ✓ disciplina,
- ✓ asignatura que imparte,
- ✓ grupos de investigación a que pertenece,
- ✓ cargo que ocupa,
- ✓ especialidad.

El proceso de asociación de los usuarios con el tipo de perfil al que pertenece se detalla a continuación:

A partir de los datos del usuario, que se extraen de los servicios del Sistema de Gestión Universitaria, se obtiene en caso de que sea estudiante, el año que cursa. Para el caso de que el año sea de 1ro a 2do se le asigna el tipo de perfil estudiante, en otro caso siendo estudiante pero de los años de 3ro a 5to se asocia al perfil de estudiante en ciclo profesional.

La información de los usuarios que no son estudiantes se encuentra ubicada en los servicios web de sistema de Assets (Sistema de registro de personal contratado en la UCI), del cual se obtiene los datos de las áreas, cargos y especialidades a la que pertenece el usuario, en caso de que sea profesor se le asocia el tipo de perfil Profesor, en otro caso es asociado al tipo de perfil base.

---

### 2.3.3 Integración con la RSU

Para que el servicio propuesto se integre con el núcleo de la RSU, es necesario describir la información de los usuarios a través de la ontología FOAF. Esta ontología se escoge como propuesta de solución ya que la red social que se desarrolla en la universidad basa su estrategia de integración en las tecnologías de la Web Semántica. Los atributos del perfil definidos son asociados a la estructura de atributos de la ontología FOAF, creándose en algunos casos nuevos atributos con los campos que sean necesarios.

#### 2.3.4 Plataforma de desarrollo

La plataforma de desarrollo Elgg incluye en su núcleo una serie de *plugins* que extienden las funcionalidades básicas desarrolladas como parte del motor del sistema. Para satisfacer las necesidades de la solución es necesario desarrollar y reutilizar otros *plugins* existentes.

Para la autenticación en el dominio es necesario modificar la funcionalidad en el sistema para que permita registrar los nuevos usuarios que se autenticuen satisfactoriamente en el servidor de dominio de la universidad.

En el proceso de registro de las relaciones de amistad entre los usuarios se hace necesario emplear un sistema de solicitudes de amistad para que la relación de los usuarios se base en solicitudes de confirmación/rechazo de amistad antes de establecer dicho nexo propiamente entre los miembros.

La plataforma Elgg presenta por defecto un conjunto de campos para el perfil de los usuarios. Para satisfacer los requerimientos planteados se hace necesario modificar esta funcionalidad, de manera que permita la gestión de estos atributos así como la definición de tipos de perfil para los usuarios.

#### 2.4 Modelo de dominio

El modelo de dominio se toma como el punto de partida para el diseño del sistema. Cuando se realiza la programación orientada a objetos, el funcionamiento interno del software va a imitar en alguna medida a la realidad, por lo que el mapa de conceptos del modelo de dominio constituye una primera versión del sistema (ROQUE y PERAZA, 2012).

Para el sistema a desarrollar se define el siguiente diagrama de clases del dominio (ver Figura 2.3):

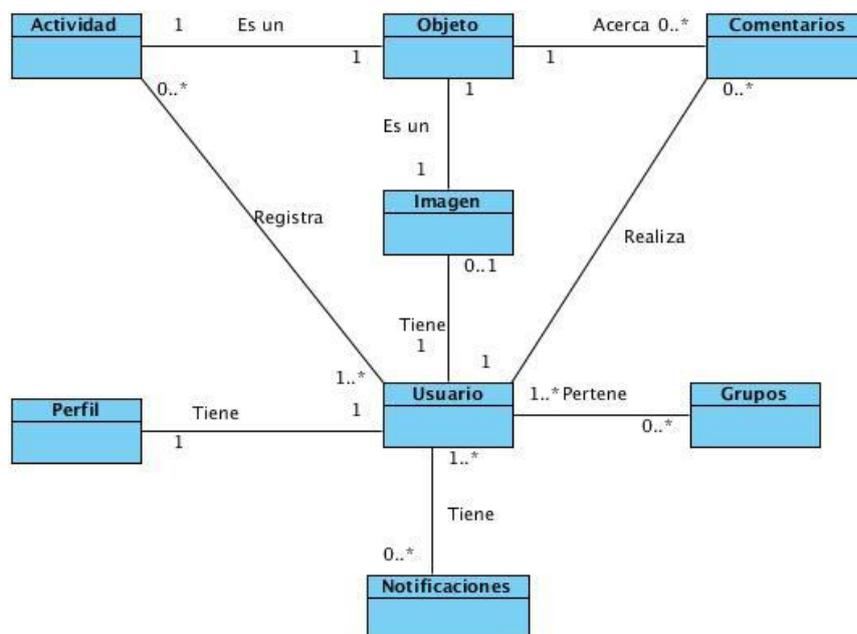


Figura 2.3 Modelo de dominio

Conceptos del modelo de dominio:

- ✓ Usuario: el usuario con el cual se accede a la RSU.
- ✓ Perfil: representa toda la información relacionada con el usuario.
- ✓ Objeto: es cualquier recurso en el sistema, ya sea imagen, actividad o comentarios.
- ✓ Actividades: la acción que realiza el usuario en el sistema.
- ✓ Comentarios: los comentarios que realizan los usuarios en la red social referente a un contenido en específico.
- ✓ Imagen: la imagen con la que cuenta cada usuario en su perfil.
- ✓ Grupos: los grupos son un espacio donde el usuario se une a ellos en dependencia de sus intereses.
- ✓ Notificaciones: las alertas que se le hace al usuario en determinados eventos.

## 2.5 Especificación de requisitos

En el artefacto “Especificación de requisitos” se describe el trabajo a realizar en el proyecto. En este se incluyen los requisitos funcionales y no funcionales del sistema.

### 2.5.1 Requisitos funcionales

Los requerimientos funcionales son las capacidades o condiciones que el sistema debe cumplir sin alterar la funcionalidad del producto y que a su vez cumpla con las necesidades del usuario (SOMMERVILLE, 2005).

Seguidamente se especifican los principales requisitos funcionales con los que cuenta el servicio: perfil de usuario:

- ✓ RF 1. Autenticar usuario en el dominio UCI.
- ✓ RF 2. Realizar solicitud de amistad.
- ✓ RF 3. Confirmar solicitud de amistad.
- ✓ RF 4. Rechazar solicitud de amistad.
- ✓ RF 5. Adicionar atributos del perfil.
- ✓ RF 6. Editar atributos del perfil.
- ✓ RF 7. Eliminar atributos del perfil.
- ✓ RF 8. Crear tipo de perfil.
- ✓ RF 9. Editar tipo de perfil.
- ✓ RF 10. Eliminar tipo de perfil.
- ✓ RF 11. Buscar amigos.
- ✓ RF 12. Mostrar perfil.
- ✓ RF 13. Mostrar atributos del perfil.
- ✓ RF 14. Mostrar tipos de perfil.
- ✓ RF 15. Mostrar solicitud de amistad.
- ✓ RF 16. Mostrar sugerencia de amigos.
- ✓ RF17. Mostrar información de servicios telemáticos.
- ✓ RF18. Enviar mensajes.
- ✓ RF19. Cerrar sesión.

#### **Descripción de requisitos:**

A continuación se muestra la descripción de tres de los requisitos funcionales del sistema (ver, Tabla 2.1, Tabla 2.2, Tabla 2.3), para consultar los restantes remitirse a los Anexos. Descripción de requisitos

del presente trabajo. La complejidad de cada requisito se define a partir del artefacto “Evaluación de requisitos”.

Tabla 2.1 Especificación de requisito. Realizar solicitud de amistad.

No	Nombre	Descripción	Complejidad	Prioridad para el cliente
[RF 2.]	Realizar solicitud de amistad.	Permite establecer relaciones con los miembros de la red social. <ol style="list-style-type: none"> <li>1. A partir de la búsqueda de una persona.</li> <li>2. A través de un listado que contiene algunas sugerencias de amistad.</li> <li>3. Desde la página del perfil.</li> </ol> El sistema muestra el mensaje de confirmación “Solicitud enviada”.	Media	Alta

**Prototipo**



Figura 2.4 Prototipo de interfaz. Realizar solicitud de amistad 1



Figura 2.5 Prototipo de interfaz. Realizar solicitud de amistad 2.



Figura 2.6 Prototipo de interfaz. Realizar solicitud de amistad 3.

Campos	Tipos de Datos	Reglas o restricciones
Nombre o número de carné de identidad del usuario.	string	La entrada de datos en el campo usuario debe puede contener letras y números, pero no admite caracteres extraños. Nombre o número de carné de identidad del usuario.
<b>Observaciones</b>	Cuando se realiza la invitación de un amigo a la red social, se registra además una solicitud de amistad asociado a este usuario.	

Tabla 2.2 Especificación de requisitos. Confirmar solicitud de amistad.

No	Nombre	Descripción	Complejidad	Prioridad para el cliente
[RF 3.]	Confirmar solicitud de amistad.	Permite establecer relaciones de amistad. Se muestra la opción para aceptar a un usuario. El sistema actualiza la lista de amigos del usuario y muestra un mensaje "El usuario (nombre del usuario) ahora es su amigo"	Media	Alta
<b>Prototipo</b>				

<p><b>Solicitudes de amistad de: Yisel Gómez Herrera</b>                  Personas que quieren ser tu amigo</p>  <p>Figura 2.7 Prototipo de interfaz. Confirmar solicitud de amistad.</p>		
Campos	Tipos de Datos	Reglas o restricciones
	No procede	No procede
Observaciones	En caso de no existan solicitudes de amistad pendientes, el sistema muestra un mensaje de información "No tiene solicitudes de amistad".	

Tabla 2.3 Especificación de requisitos. Rechazar solicitud de amistad.

No	Nombre	Descripción	Complejidad	Prioridad para el cliente
[RF 4.]	Rechazar solicitud de amistad.	No permite establecer una relación de amistad. Se muestra la opción de rechazar una solicitud de amistad. Se muestra un mensaje "Se ha rechazado la petición de amistad".	Media	Alta
<b>Prototipo</b>				
<p><b>Solicitudes de amistad de: Yisel Gómez Herrera</b>                  Personas que quieren ser tu amigo</p>  <p>Figura 2.8 Prototipo de interfaz. Rechazar solicitud de amistad.</p>				
Campos	Tipos de Datos	Reglas o restricciones		
	<ul style="list-style-type: none"> <li>No procede</li> </ul>	<ul style="list-style-type: none"> <li>No procede</li> </ul>		
Observaciones				

### 2.5.2 Requisitos no funcionales

Los requisitos no funcionales son propiedades o cualidades que el producto debe tener como las características que hacen al producto atractivo, usable, rápido o confiable (SOMMERVILLE, 2005).

### Usabilidad:

- ✓ RNF 1.1 El sistema solo podrá ser utilizado por los usuarios de la UCI tanto en la sede central como en las facultades regionales.
- ✓ RNF 1.2 El sistema debe presentar una interfaz amigable que permita a los usuarios la fácil interacción con el mismo. Debe además ser una interfaz de manejo cómodo que posibilite a los usuarios sin experiencia una rápida adaptación.
- ✓ RNF 1.3 Los administradores de la aplicación deben tener un nivel de acceso total en la aplicación para poder dar mantenimiento al sistema.

### Confiabilidad:

- ✓ RNF 2.1 Los usuarios deben tener acceso (según sus permisos) a la información solicitada.

### Eficiencia:

- ✓ RNF 3.1 El sistema debe soportar 5000 usuarios trabajando de forma concurrente.
- ✓ RNF3.2 La respuesta del sistema a cualquier petición no debe exceder los 5 segundos.

### Soporte:

- ✓ RNF4.1 Se le debe dar mantenimiento al sistema y corregir cualquier problema que surja con la aplicación.
- ✓ RNF 4.2 Se deberá prever futuras versiones del sistema y la actualización de las tecnologías empleadas en el desarrollo.

### Restricciones de diseño:

- ✓ RNF 5.1 El sistema estará desarrollado sobre la plataforma Elgg 1.8.3.
- ✓ RNF 5.2 Servidor web Apache2.
- ✓ RNF 5.3 Se empleará el navegador web Mozilla Firefox 3.6 o superior.
- ✓ RNF 5.3 Lenguaje de programación PHP 5.2 o superior.
- ✓ RNF 5.4 Servidor de bases de datos MySQL 5 o superior.
- ✓ RNF 5.5 Entorno integrado de desarrollo (IDE) NetBeans 7.1.
- ✓ RNF 5.6 Se debe utilizar herramienta de modelado Visual Paradigm 8.0.

- ✓ RNF 5.7 El proceso que se emplea para el desarrollo de la aplicación es el “Proceso de desarrollo con enfoque ágil orientado al segundo nivel de CMMI”.

Requisitos para la documentación de usuarios en línea y ayuda del sistema:

- ✓ RNF 6.1 El sistema debe presentar un manual de usuario, permitiendo con ello un correcto uso de sus funcionalidades y brindarle al usuario una mayor experiencia del trabajo con el mismo.

Componentes comprados:

- ✓ RNF 7.1 Licencia del Visual Paradigm.

Interfaz:

- ✓ Interfaz de usuario:
  - ❖ RNF 8.1 El sistema está optimizado para una resolución de 1024x768.
  - ❖ RNF 8.2 El diseño gráfico es conforme con las pautas de diseño de la Universidad.
  - ❖ RNF 8.3 El sistema cuenta con una interfaz amigable, fácil de usar, sencilla e interactiva.
- ✓ Interfaz de hardware:
  - ❖ RNF 8.4 Para el desarrollo: PC Intel Pentium 4 o superior, CPU 3GHZ o superior, 2GB RAM o superior, 160 GB HDD o superior.
  - ❖ RNF 8.5 Para explotación del servidor: CPU Dual Core 2.0 GHZ o superior, memoria RAM de 4 GB (recomendado 6 GB), 250 GB HDD.
  - ❖ RNF 8.6 Para explotación del cliente: PC Pentium 3 o superior, CPU 133 MHZ o superior, 256MB RAM mínimo o superior.
- ✓ Requisitos legales, de Derecho de Autor y otros:
  - ❖ RNF 9.1 La plataforma para el desarrollo de la aplicación debe basarse en la licencia GPL.
- ✓ Estándares aplicables:
  - ❖ RNF 10.1 Se debe utilizar los estándares de codificación del Elgg.

## 2.6 Patrones de arquitectura

Los patrones arquitectónicos, o patrones de arquitectura, son patrones de diseño de software que ofrecen soluciones a problemas de arquitectura de software en ingeniería de software. Dan una descripción de los elementos y el tipo de relación que tienen junto con un conjunto de restricciones sobre cómo pueden ser usados. Un patrón arquitectónico expresa un esquema de organización estructural esencial para un sistema de software, que consta de subsistemas, sus responsabilidades e interrelaciones (PRESSMAN, 2011).

El patrón de arquitectura del Elgg es el MVC (Modelo, Vista, Controlador)

A continuación se describen las capas del modelo (ELGG, 2011):

- ✓ El Modelo es el objeto que representa los datos del programa. Maneja los datos y controla todas sus transformaciones. El Modelo no tiene conocimiento específico de los Controladores o de las Vistas, ni siquiera contiene referencias a ellos. Es el propio sistema el que tiene encomendada la responsabilidad de mantener enlaces entre el Modelo y sus Vistas, y notificar a las Vistas cuando cambia el Modelo.
- ✓ La Vista es la responsable de crear lo que vemos. Generalmente, se envía código HTML al navegador web; pero también podría ser RSS, JSON<sup>14</sup> o cualquier otro formato de datos. El sistema de vistas maneja tanto el diseño de las páginas como la generación de fragmentos de presentación (como un pie de página o una barra de herramientas. Elgg tiene un poderoso sistema de vistas, el cual asegura que la presentación este separada de la lógica, permitiendo características avanzadas como generación automática de RSS, desarrollo rápido de interfaces móviles, y presentaciones tales como:
  - HTML estándar.
  - HTML optimizado para accesibilidad.
  - Internet móvil.
  - iPhone.
  - Widgets embebidos.

---

<sup>14</sup> Acrónimo de JavaScript Object Notation, es un formato ligero para el intercambio de datos. JSON es un subconjunto de la notación literal de objetos de JavaScript que no requiere el uso de XML.

- ✓ El Controlador gestiona las acciones que es la forma de Elgg de proporcionar interactividad: cada participación activa por parte del usuario se realiza a través de una acción. Inicio de sesión, crear, actualizar o eliminar el contenido son todas las categorías genéricas de las acciones. El método tradicional para el desarrollo sería la creación de scripts PHP que se encargan de cada acción individual. Elgg desvía todo a través de un solo controlador, la acción unificada. También impide que los autores de forma accidental hagan sus acciones disponibles para los usuarios no registrados (si no quieren que lo sean).

## 2.7 Patrones de diseño

A continuación se describen los patrones de diseño empleados en el desarrollo de la solución:

### 2.7.1 Patrones GRASP

GRASP es el acrónimo de **General Responsibility Assignment Software Patterns**, que traducido al español significa: Patrones de Software para la Asignación General de Responsabilidad. Estos patrones son considerados como uno de los principios fundamentales de diseño de objetos para la asignación de responsabilidades.

En la plataforma Elgg, utilizada para el desarrollo del servicio: perfil de usuario se manifiestan los siguientes patrones: experto, creador, controlador, alta cohesión, bajo acoplamiento y polimorfismo.

**Experto:** Define que la responsabilidad de realizar una labor es de la clase que tiene o puede tener los datos involucrados (atributos) (LARMAN, 2003). Se evidencia este patrón en las clases del Elgg, que contienen toda la información necesaria para realizar la labor que tiene encomendada.

**Creador:** Este patrón como su nombre lo indica es el que crea, guía la asignación de responsabilidades relacionadas con la creación de objetos, se asigna la responsabilidad de que una clase B cree un objeto de la clase A solamente cuando (LARMAN, 2003):

- ✓ B contiene a A
- ✓ B es una agregación (o composición) de A
- ✓ B almacena a A
- ✓ B tiene los datos de inicialización de A (datos que requiere su constructor)

- ✓ B usa a A.

Este patrón se evidencia en la clase ElggUser que representa a los usuarios del sistema, la misma presenta la funcionalidad de registrar una relación de amistad con otro usuario, a través de la creación de una instancia de la clase ElggRelationship.

**Controlador:** Es un evento generado por actores externos. Se asocian con operaciones del sistema, operaciones del sistema como respuestas a los eventos del sistema, tal como se relacionan los mensajes y los métodos. Normalmente un controlador delega en otros objetos el trabajo que se necesita hacer; coordina o controla la actividad (LARMAN, 2003).

Se evidencia en las clases ElggSiteController y ElggUserController.

**Alta cohesión:** La cohesión es una medida de la fuerza con la que se relacionan las clases y el grado de focalización de las responsabilidades de un elemento. Cada elemento de nuestro diseño debe realizar una labor única dentro del sistema, no desempeñada por el resto de los elementos y auto-identificable, una clase con baja cohesión hace muchas cosas no relacionadas o hace demasiado trabajo (LARMAN, 2003).

Se emplea este patrón debido a que es necesario controlar la complejidad de cada clase utilizada para mantener un buen comportamiento de las mismas, por esto, las clases que fueron identificadas con una gran cantidad de funcionalidades se fragmentaron en otras clases de manera que se repartiera equitativamente el peso de la complejidad, manteniendo la coherencia de las clases.

**Bajo Acoplamiento:** Este patrón es un principio que asigna la responsabilidad de controlar el flujo de eventos del sistema, a clases específicas. Esto facilita la centralización de actividades (validaciones, seguridad, etc.). El controlador no realiza estas actividades, las delega en otras clases con las que mantiene un modelo de alta cohesión. Un error muy común es asignarle demasiada responsabilidad y alto nivel de acoplamiento con el resto de los componentes del sistema (LARMAN, 2003).

Este patrón es utilizado con la idea de tener las clases lo menos ligadas entre sí posibles. De forma tal que en caso de producirse una modificación en alguna de ellas, se tenga la mínima repercusión posible en el resto de clases, potenciando la reutilización, y disminuyendo la dependencia entre las clases.

**Polimorfismo:** Siempre que se tenga que llevar a cabo una responsabilidad que dependa del tipo, se tiene que hacer uso del polimorfismo, cuando las alternativas o comportamientos relacionados varían según el tipo (clase), asigne la responsabilidad para el comportamiento utilizando operaciones polimórficas a los tipos para los que varía el comportamiento (LARMAN, 2003).

Se evidencia en las clases del Elgg para el desarrollo de las funciones entre las entidades, como son la extracción de metadatos y las imágenes representativas.

---

### 2.7.2 Patrones GOF

Los patrones GOF Gang-of-Four (“pandilla de los cuatro”) descritos en el libro *Design Patterns* (Gama 1995) definieron un catálogo con 23 patrones básicos.

Seguidamente se describe el patrón que se pone de manifiesto en la plataforma Elgg:

**Singleton:** Este patrón está diseñado para restringir la creación de objetos pertenecientes a una clase o el valor de un tipo a un único objeto. Su intención consiste en garantizar que una clase sólo tenga una instancia y proporcionar un punto de acceso global a ella (GAMMA *et al.*, 2003).

Este patrón se evidencia en la definición de variables globales del Elgg, tales como la variable `current_site` y `current_user`.

## 2.8 Modelo de datos

El modelo de datos es un conjunto de conceptos que sirven para describir la estructura, semántica, restricciones y relaciones existen entre las entidades de una base de datos. Para el desarrollo del servicio: perfil usuario se emplea el siguiente modelo de datos (ver Figura 2.9), generado de las entidades persistentes en la plataforma Elgg.

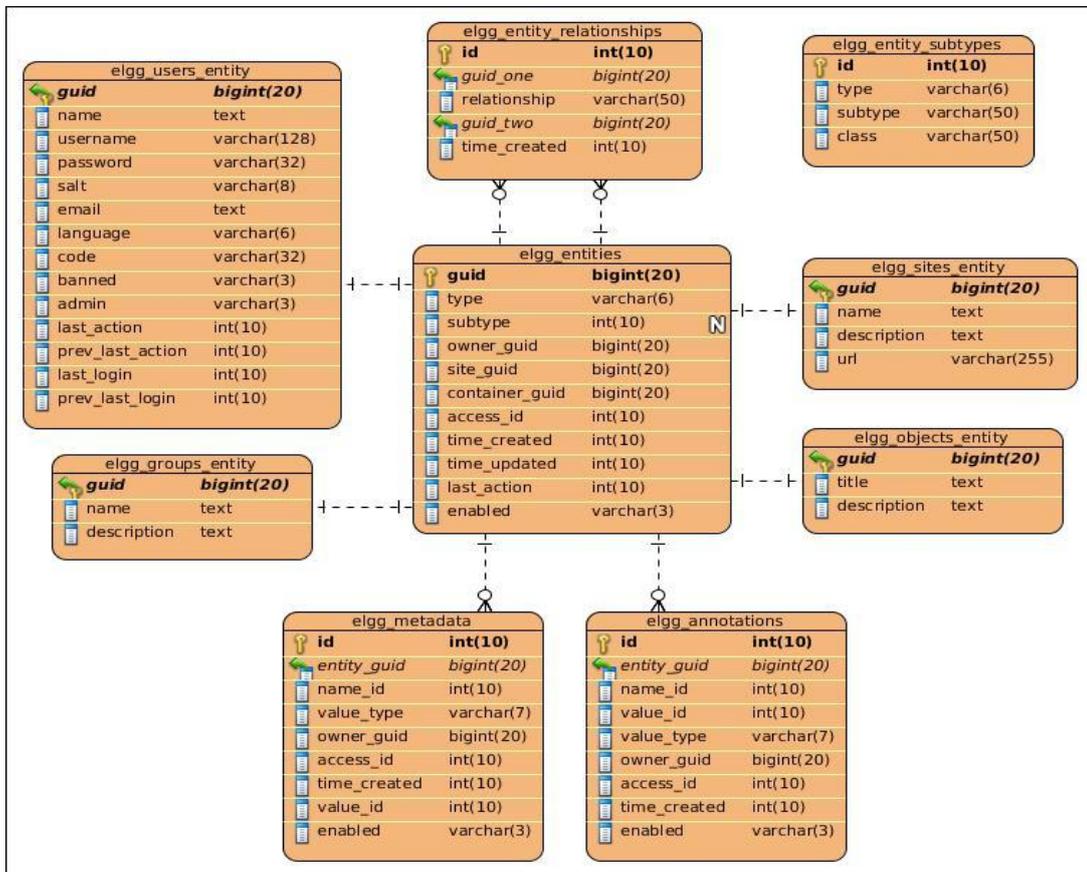


Figura 2.9 Modelo de datos

## 2.9 Conclusiones parciales

La descripción de los requerimientos funcionales y no funcionales del sistema permitió detallar con claridad las funcionalidades a implementar en la aplicación. Además se identificaron los patrones de diseño que se manifiestan en la construcción del servicio así como el patrón de arquitectura utilizado por la herramienta propuesta el cual sirvió de guía para el desarrollo del servicio. También se representa el modelo de dominio el cual posibilitó tener un mejor entendimiento del sistema.

### 3 Capítulo III: Implementación y validación de la propuesta

#### 3.1 Introducción

Propiciar la calidad en el software es una actividad que ha surgido como consecuencia de la fuerte demanda de sistemas de software en todos los procesos que se desarrollan en la actualidad. Las pruebas representan un elemento importante para garantizar la calidad y constituyen una revisión final de las especificaciones. El objetivo de la etapa de pruebas es garantizar la calidad del producto desarrollado.

En el presente capítulo se tratan los temas referentes a la validación del sistema donde se describen los estándares de codificación definidos por la plataforma propuesta, el diagrama de despliegue que contiene los componentes del sistema, el modelo de datos con las tablas generadas del Elgg y los tipos de pruebas al sistema.

#### 3.2 Desarrollo de la propuesta de solución

Para el desarrollo del sistema se utilizan un conjunto de extensiones de la herramienta, denominadas *plugin*, que permiten ampliar las funcionalidades de la misma y sirven de apoyo para el desarrollo de los requisitos funcionales. A continuación se desglosan los mismos:

- ✓ **UCI Authentication 1.0:** integra la autenticación por la pasarela UCI al sistema Elgg.
- ✓ **Red Social UCI API Integration 1.0:** posibilita la integración del servicio: perfil de usuario a la API del núcleo de la RSU.
- ✓ **Site Pages 1.8:** permite crear páginas públicas para mostrar información a los usuarios tales como: “Acerca”, “Términos de uso” y “Privacidad”.
- ✓ **Garbage Collector 1.5:** realiza algunas tareas de limpieza de base de datos, con el objetivo de mantener optimizadas las tablas de la misma.
- ✓ **Groups 1.8:** permite crear grupos de personas que se organizan por un interés en común.
- ✓ **HTMLawed 1.5:** proporciona la protección contra ataques de validación de datos.
- ✓ **Invite Friends 1.8:** permite invitar a tus amigos a través de correo electrónico.
- ✓ **Log Browser 1.8:** visualiza el registro de eventos del sistema.

- ✓ **Log Rotate 1.5:** permite realizar la extracción y almacenamiento de los registros del sistema periódicamente.
- ✓ **Members 1.8:** muestra una lista de los miembros del sitio.
- ✓ **Messages 1.8:** permite al usuario enviar mensajes internos a los otros usuarios del sistema.
- ✓ **Notifications 1.7:** permite que otros *plugins* generen notificaciones a los usuarios.
- ✓ **Profile 1.8:** permite que los usuarios se puedan crear perfiles.
- ✓ **Friend request 3.0.2:** permite a los usuarios confirmar las solicitudes de amistad.
- ✓ **Reported Content 1.8:** brinda la posibilidad de que los usuarios denuncien un contenido y le informen a los administradores.
- ✓ **Search 1.8:** permite que los usuarios realicen búsquedas en el sistema a partir de un criterio de búsqueda determinado.
- ✓ **The Wire 1.8:** permite enviar mensajes cortos.
- ✓ **TinyMCE 1.8:** proporciona al usuario opciones de edición de texto avanzadas en las áreas de escritura.
- ✓ **Profile Manager 7.2:** permite gestionar los campos del perfil.
- ✓ **Red Social UCI Theme 1.0:** representa el diseño del sistema.
- ✓ **Telematicos API Integration 1.0:** integra la API de los servicios telemáticos de la UCI al sistema Elgg.

### 3.3 Estándares de codificación

La implementación del sistema se rige por los estándares de codificación definidos por la plataforma Elgg:

#### 3.3.1 Estándares de codificación de PHP

Todo el desarrollo del núcleo y *plugins* incrustados están obligados a estar en este formato.

Finales de línea de Unix

- ✓ Define un espaciado de 4 caracteres de tabulación
- ✓ No hay ninguna etiqueta de acceso directo de PHP (`<?or? = or<%`).
- ✓ Comentarios de PHPDoc sobre las funciones y clases.

- ✓ Cerrar los bloques de código entre llaves de manera obligatoria.

Mal:

```
if (true)
    foreach($arr as $elem)
        echo $elem;
```

Bien:

```
if (true) {
    foreach ($arr as $elem) {
        echo $elem;
    }
}
```

- ✓ Nombre de las funciones independientes usando el carácter de subrayado, ejemplo: (autenticar\_usuario).
- ✓ Nombre clases utilizando CamelCase() y los métodos que utilizan lowerCamelCase ().
- ✓ Nombre globales y constantes en ALL\_CAPS (ACCESS\_FRIENDS, \$ CONFIG).
- ✓ Uso de guión bajo y CamelCase para separar palabras en inglés en las funciones, clases y métodos. (get\_default\_site(), elggUser->isLoggedIn()).
- ✓ El espacio en las funciones tal como se muestra en el ejemplo: like\_this(\$required, \$optional = TRUE).
- ✓ Espacio, palabras clave y las construcciones deben ser de este tipo: if (FALSO) {...}.
- ✓ Utilizar correctamente los espacios, las comillas, y llaves en las cadenas:

Mal (difícil de leer, mal uso de comillas y llaves):

```
echo `Hola, `.$nombre.`! Como es tu {$momento_del_dia}?`;
```

Bién:

```
echo "Hola, $nombre! Como es tu $ momento_del_dia?";
```

- ✓ Las líneas de código no deben pasar de los 100 caracteres.
- ✓ Utilice // or / \*\* / al comentar.

### 3.3.2 Estándares de codificación de CSS

- ✓ Se debe utilizar una abreviatura cuando sea posible:

Mal:

```
background-color: #333333;
background-image: url(...);
background-repeat: repeat-x;
background-position: left 10px;
padding: 2px 9px 2px 9px;
```

Bien:

```
background: #333 url(...) repeat-x left 10px;
padding: 2px 9px;
```

- ✓ Se debe utilizar guiones como separadores en las clases / IDS, no pone de relieve:

Mal:

```
.example_class
```

Bien:

```
.example-class
```

- ✓ Una propiedad por línea:

Mal:

```
color: white;font-size: smaller;
```

Bien:

```
color: white;
```

```
font-size: smaller;
```

- ✓ Las declaraciones de las propiedades deben tener una separación de esta manera: `la propiedad: valor`;

**Mal:**

```
color:value;
color :value;
color : value;
```

**Bien:**

```
color: value;
```

- ✓ Grupo de proveedores de prefijos para la misma propiedad en conjunto:
- ✓ Incluya siempre la versión sin el prefijo del proveedor:
- ✓ Coloque una nueva línea adicional entre los grupos de proveedor prefijados y otras propiedades:

**Mal:**

```
-moz-border-radius: 5px;
border: 1px solid #999999;
-webkit-border-radius: 5px;
width: auto;
```

**Bien:**

```
border: 1px solid #999999;
-webkit-border-radius: 5px;
-moz-border-radius: 5px;
border-radius: 5px;
width: auto;
```

- ✓ Las declaraciones del Grupo de subpropiedades:

Mal:

```
background-color: white;
color: #0054A7;
background-position: 2px -257px;
```

Bien:

```
background-color: white;
background-position: 2px -257px;
color: #0054a7;
```

### 3.3.3 Estándares de codificación Javascript

- ✓ Las normas del mismo formato de PHP.
- ✓ Todas las funciones deben estar en el espacio de nombres elgg.
- ✓ Expresiones de función deben terminar con un punto y coma:

```
elgg.ui.toggles = function(event) {
    event.preventDefault();
    $(target).slideToggle('medium');
};
```

## 3.4 Diagrama de despliegue del sistema

Los diagramas de despliegue proveen una vista de cómo los componentes se despliegan en el transcurso de la infraestructura del sistema. Permite comprender la correspondencia entre la arquitectura de software y la de hardware. El siguiente diagrama de despliegue (ver Figura 3.1), muestra la forma en que se van a comunicar los componentes del sistema. Las PC clientes se van a comunicar con los servicios que se encuentran en los servidores de aplicaciones a través del protocolo

HTTP<sup>15</sup>. Este a su vez se comunica con el servidor de base de datos por medio del protocolo TCP/IP<sup>16</sup> para la obtención de la información que maneja el sistema y a su vez se comunica con el servidor LDAP<sup>17</sup> para la autenticación de los usuarios que acceden al sistema.

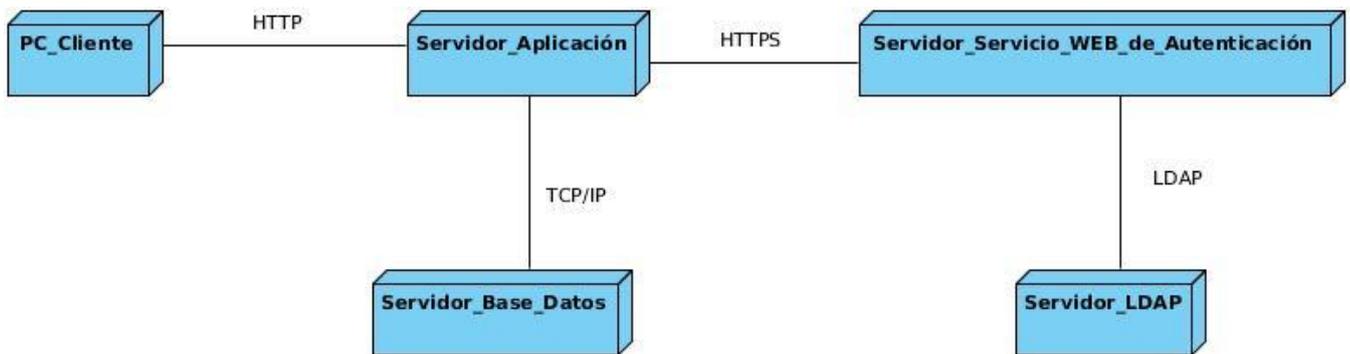


Figura 3.1 Diagrama de despliegue del sistema

### 3.5 Validación

Para realizar la validación de los requerimientos funcionales y no funcionales del sistema se presenta como tipo de prueba realizar las pruebas funcionales. Estas tienen como principal objetivo validar que el sistema creado cumpla con las funcionalidades por las que fue creado.

Además se validan los requerimientos no funcionales a través de las pruebas de carga y estrés con el objetivo de comprobar los requerimientos establecidos. Estas pruebas permiten comprobar el rendimiento del sistema en condiciones de trabajo extremas, asegurando la calidad de respuesta del mismo en dichas condiciones.

#### 3.5.1 Pruebas de software

El instrumento adecuado para determinar el estado de la calidad de un producto software es el proceso de pruebas. En este proceso se ejecutan pruebas dirigidas a componentes del software o al sistema de software en su totalidad, con el objetivo de medir el grado en que el software cumple con los requerimientos. Las pruebas de software constituyen un conjunto de herramientas, técnicas y métodos

<sup>15</sup> HTTP (protocolo de transferencia de hipertexto), HTTP define la sintaxis y la semántica que utilizan los elementos de software de la arquitectura web (clientes, servidores, proxies) para comunicarse.

<sup>16</sup> TCP/IP (Protocolo de Control de Transmisión).

<sup>17</sup> LDAP (Protocolo Ligero de Acceso a Directorios).

que evalúan el desempeño de un programa (PRESSMAN, 2005). Estas involucran las operaciones del sistema, evaluando los resultados bajo condiciones controladas, lo que hace que la realización de pruebas al software sea un factor de vital importancia. La etapa de pruebas implica los siguientes aspectos (GARCERANT, 2008):

- ✓ Verificar la integración adecuada de los componentes.
- ✓ Verificar que todos los requisitos sean implementados correctamente.
- ✓ Identificar y asegurar que los defectos encontrados sean corregidos antes de entregar el software al cliente.

### 3.5.2 Pruebas funcionales

Se denominan pruebas funcionales o *Functional Testing*, a las pruebas de software que tienen por objetivo probar que los sistemas desarrollados, cumplan con las funciones específicas para los cuales han sido creados (B., 2009).

Dentro de las pruebas funcionales se encuentra el método de caja negra, las cuales están diseñadas para validar los requisitos funcionales sin fijarse en el funcionamiento del programa. La prueba de caja negra permite al ingeniero del software obtener conjuntos de condiciones de entrada que ejerciten completamente todos los requisitos funcionales de un programa. Estas pruebas no son una alternativa a las técnicas de prueba de caja blanca, más bien se trata de un enfoque complementario que intenta descubrir diferentes tipos de errores que los métodos de caja blanca.

La prueba de caja negra intenta encontrar errores de las siguientes categorías (PRESSMAN, 2005):

- ✓ Las funcionalidades que están incorrectas o ausentes.
- ✓ Los errores de interfaz.
- ✓ Los errores en estructuras de datos o en accesos a las bases de datos externas.
- ✓ Los errores de rendimiento.
- ✓ Los errores de inicialización y terminación.

**3.5.3 Descripción de casos de prueba**

A continuación se presenta una muestra del diseño de casos de pruebas realizados al sistema (ver, Tabla 3.1, Tabla 3.2, Tabla 3.3), para consultar los restantes dirigirse a los Anexos. Descripción de casos de pruebas del presente trabajo:

Tabla 3.1 Descripción de caso de prueba. Realizar solicitud de amistad.

<b>Escenario</b>	<b>Descripción</b>	<b>Respuesta del sistema</b>	<b>Flujo central</b>
EC 1.1 Realizar solicitud correctamente.	Mediante este escenario se realiza la solicitud de amistad a los usuarios registrados en el sistema.	El sistema envía la solicitud y muestra un mensaje de confirmación: "Solicitud enviada".	Luego de acceder al sistema el usuario selecciona a partir de la búsqueda de un usuario que esté registrado en uno de los dominios de la universidad y selecciona en el menú asociado al usuario la opción "Agregar a mis amigos". También puede acceder desde el bloque que contiene posibles relaciones de amistad, se selecciona un usuario y del menú asociado al usuario selecciona la opción "Agregar a mis amigos". Otra forma de acceso es a partir de la interfaz que muestra el perfil del usuario seleccionado y selecciona la opción "Agregar a mis amigos".

Tabla 3.2 Descripción de caso de prueba. Aceptar solicitud de amistad.

<b>Escenario</b>	<b>Descripción</b>	<b>Respuesta del sistema</b>	<b>Flujo central</b>
EC 1.1 Aceptar solicitud de amistad correctamente.	Mediante este escenario se acepta la solicitud de amistad realizada a un usuario en el sistema.	El sistema añade al listado de amigos del usuario la solicitud de amistad aceptada y muestra el mensaje "El (nombre de usuario) ahora es su amigo".	El usuario una vez autenticado en el sistema selecciona en el panel de navegación la opción "Amigos", donde se muestra una interfaz que contiene el listado de amigos y la petición de amistad pendiente, el usuario selecciona la opción "Aceptar". También puede aceptar la solicitud de amistad desde el perfil del usuario donde se muestra un bloque que contiene información sobre las peticiones de amistad del usuario asociado, el usuario selecciona la opción solicitudes de amistad y se muestra una interfaz que contiene el listado de amigos y las peticiones de amistad pendientes, el usuario selecciona la opción "Aceptar" del usuario que desea agregar a su listado

			de amigos.
--	--	--	------------

Tabla 3.3 Descripción de caso de prueba. Realizar solicitud de amistad.

Escenario	Descripción	Respuesta del sistema	Flujo central
EC 1.1 Cancelar solicitud de amistad correctamente.	Mediante este escenario se cancela la solicitud de amistad de un usuario.	El sistema muestra un mensaje "Se ha rechazado la petición de amistad".	El usuario una vez autenticado en el sistema selecciona en el panel de navegación la opción "Amigos", donde se muestra una interfaz que contiene el listado de amigos y la petición de amistad pendiente, el usuario selecciona la opción "Rechazar".

Para verificar el correcto funcionamiento de la interfaz del software se realiza la prueba de caja negra, la cual permite identificar las no conformidades encontradas en el sistema. En el gráfico que se muestra a continuación (ver, Figura 3.2), este brinda información sobre el total de no conformidades encontradas y las que se resolvieron por cada iteración. Para un total de 18 requisitos funcionales se detectaron 4 no conformidades en la primera iteración de las cuales fueron resueltas satisfactoriamente, y en la segunda iteración se redujo las no conformidades llegando solo a 3 que también fueron resueltas, número que se redujo hasta la tercera iteración donde no se obtuvieron no conformidades.

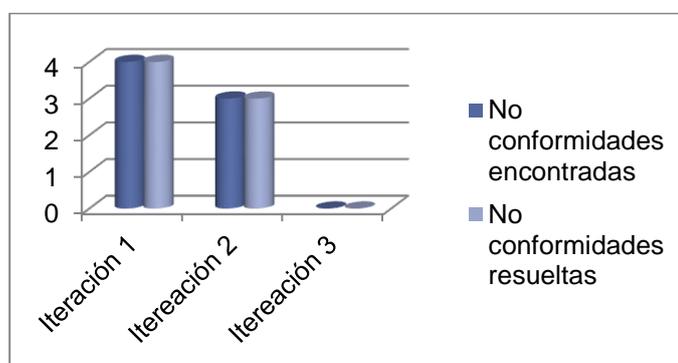


Figura 3.2 Cantidad de no conformidades por iteración

### 3.5.4 Prueba de carga y estrés

Este tipo de pruebas se realiza con el objetivo de determinar lo rápido que realiza una tarea el sistema en condiciones extremas de trabajo. Este tipo de prueba se realiza para determinar la solidez de la aplicación en los momentos de carga extrema y ayuda a los administradores para determinar si la

aplicación rendirá lo suficiente en caso de que la carga real supere a la carga esperada. Para realizar este tipo de prueba se utiliza la herramienta Jmeter <sup>18</sup> en su versión 2.3.4, su instalación se encuentra ubicada en los repositorios de las distribuciones de Ubuntu.

### 3.5.5 Entorno de pruebas

Para la ejecución de las pruebas fue necesario especificar el entorno de pruebas que a continuación se detalla, teniendo en cuenta las características de las máquinas donde se instaló la aplicación y los componentes del sistema.

Para la instalación de la aplicación se utiliza una máquina con las siguientes características:

- ✓ Microprocesador Intel Core i3 – 2100 a 3.10 GHz
- ✓ Memoria RAM de 2Gb
- ✓ Disco duro de 250 GB
- ✓ Sistema Operativo Ubuntu 12.04

Se realiza la prueba de carga y estrés a través de la herramienta Jmeter, la cual brinda los resultados del rendimiento del sistema a partir de los casos de prueba que se ejecutan. Estos casos de prueba se enfocan en la ejecución de las peticiones HTTP a la aplicación que se especifican a partir de los siguientes parámetros de entrada:

- ✓ No. Usuarios: cantidad de usuarios conectados concurrentemente.
- ✓ Promedio: tiempo promedio de respuesta a las peticiones realizadas dado en milisegundos.
- ✓ Mínimo: menor número de respuesta obtenido de todas las muestras realizadas dado en milisegundos.
- ✓ Máximo: mayor número de respuesta obtenido de todas las muestras realizadas dado en milisegundos.
- ✓ Rendimiento: número que muestra la cantidad de peticiones por segundo realizadas.
- ✓ Error: porcentaje de error que se obtiene a partir de las peticiones que se realizan.

---

<sup>18</sup> Aplicación de escritorio, es un software de código abierto, diseñado para cargar el comportamiento de pruebas funcionales y medir el rendimiento. (<http://jmeter.apache.org/>)

3.5.6 Resultados de la prueba de carga y estrés

Seguidamente se presenta la ejecución de las pruebas de rendimiento a 3 de las vistas del sistema con el objetivo de validar los requerimientos no funcionales anteriormente definidos. Las interfaces seleccionadas son la página de inicio, página del perfil y la página de miembros (ver, Tabla 3.4, Tabla 3.5, Tabla 3.6).

Tabla 3.4 Página de inicio.

No. usuarios	Promedio	Mínimo	Máximo	Rendimiento	Error
1	356	356	356	2.8	0.00%
10	491	370	581	7.4	0.00%
100	6935	686	15420	6.2	0.00%
300	8526	526	20345	7.2	0.12%

Tabla 3.5 Página perfil.

No. usuarios	Promedio	Mínimo	Máximo	Rendimiento	Error
1	135	135	135	6.5	0.00%
10	154	141	166	9.6	0.00%
100	2061	267	3420	23.6	0.00%
300	2250	289	4870	15.6	0.00%

Tabla 3.6 Página miembros.

No. usuarios	Promedio	Mínimo	Máximo	Rendimiento	Error
1	173	173	173	5.8	0.00%
10	182	165	206	9.0	0.00%
100	2756	337	4892	17.3	0.00%
300	3824	374	9421	11.2	0.00%

De los resultados obtenidos en las pruebas se determina que la aplicación cumple con los requerimientos que se presentan. Además evidencia que el sistema presenta un tiempo de respuesta mínimo de 135 milisegundos con un 0 % de errores para la página del perfil y en un tiempo máximo de 356 milisegundos con un 0 % de error para la página de inicio. Estos resultados representan un tiempo 14 veces menor del tiempo definido de 5 segundos para las peticiones del sistema.

3.5.7 INTEGRACIÓN CON OTROS SERVICIOS DE LA RED SOCIAL

Se define como otra forma de validación del funcionamiento de la aplicación la integración del servicio: perfil de usuario con otros componentes que forman parte de la RSU que actualmente están en desarrollo. A continuación se muestra el procedimiento de integración de cada uno de los servicios tomados (ver Figura 3.3).

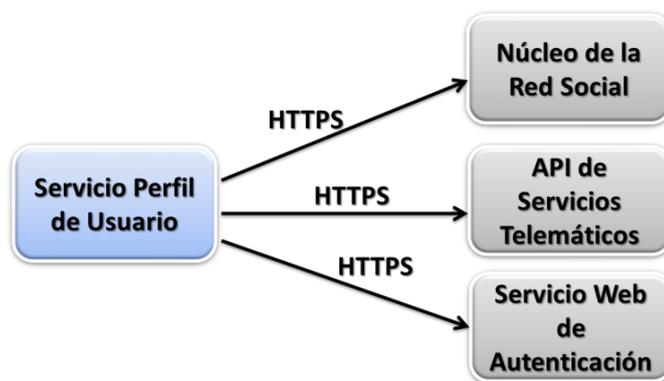


Figura 3.3 Integración con servicios de la RSU

- ✓ **Núcleo de la Red Social:** tiene como misión permitir la integración basada en Web Semántica de todos los servicios de la RSU a través de un API bien definida. El servicio: perfil de usuario se integra a través de la utilización de un componente del Elgg que realiza las peticiones a los métodos del API mediante el protocolo HTTPS. La integración permite mostrar las sugerencias de amistad de los usuarios basadas en la cantidad de amigos en común.
- ✓ **API de Servicios Telemáticos:** Interfaz de Programación de Aplicaciones que tiene como objetivo integrar los principales servicios telemáticos de la UCI con la Red Social. La integración con esta API se realiza con el objetivo de mostrarle al usuario la información sobre la cuota de navegación y para contactar a los usuarios a través de mensajes instantáneos. La aplicación se integra a través de la utilización de un componente del Elgg que realiza las peticiones a los métodos del API mediante el protocolo HTTPS.
- ✓ **Servicio Web de Autenticación:** posibilita la autenticación de los usuarios en el dominio de la institución de la misma forma recupera los datos básicos de la cuenta registrada. La integración

con este servicio se realiza mediante de la utilización de un componente del Elgg que realiza las peticiones a los métodos del servicio web a través del protocolo HTTPS.

### 3.5.8 Resultados de la pruebas de integración

Esta técnica sistemática se aplica con el objetivo de detectar errores aislados y fáciles de corregir asociados con la interacción, las siguientes tablas (Ver Tabla 3.7, Tabla 3.8, Tabla 3.9) muestran los resultados después de realizar una iteración de este tipo de prueba en interacción con el Núcleo de la red social y la API de servicios telemáticos las cuales arrojan los siguientes resultados:

Tabla 3.7 Prueba de integración Int\_1. Núcleo de la red social.

<b>Caso de prueba: Mostrar sugerencia de amistad.</b>
<b>Número de caso de prueba:</b> Int_1.
<b>Subsistema a integrar:</b> Núcleo de la red social.
<b>Condiciones de ejecución:</b> El servicio: perfil de usuario debe autenticarse con la API del núcleo.
<b>Descripción de la prueba:</b> Comprobar que el servicio perfil de usuario consulte los datos almacenados en el Núcleo de la red social.
<b>Entradas/Pasos de ejecución:</b> El Núcleo de la red social brinda al servicio: perfil de usuario los resultados de las sugerencias de amistad.
<b>Resultado esperado:</b> Se muestra un bloque con información de las sugerencias de amistad.
<b>Evaluación:</b> Prueba satisfactoria.

Tabla 3.8 Prueba de integración Int\_2. Núcleo de la red social.

<b>Caso de prueba: Registrar una persona en el sistema.</b>
<b>Número de caso de prueba:</b> Int_2
<b>Subsistema a integrar:</b> Núcleo de la red social.
<b>Condiciones de ejecución:</b> El servicio: perfil de usuario debe autenticarse con la API del núcleo.
<b>Descripción de la prueba:</b> Comprobar que el servicio: perfil de usuario envíe la petición para registrar los datos en el Núcleo de la red social.
<b>Entradas/Pasos de ejecución:</b> El Núcleo de la red social envía un mensaje al servicio: perfil de usuario indicando que la persona ha sido registrada satisfactoriamente.
<b>Resultado esperado:</b> La persona se registre en el servicio: perfil de usuario correctamente.
<b>Evaluación:</b> Prueba satisfactoria.

Tabla 3.9 Prueba de integración Int\_3. API de servicios telemáticos.

<b>Caso de prueba: Mostrar información de servicios telemáticos</b>
<b>Número de caso de prueba:</b> Int_3
<b>Subsistema a integrar:</b> API de servicios telemáticos.
<b>Condiciones de ejecución:</b> El servicio: perfil de usuario debe autenticarse con la API de servicios telemáticos.
<b>Descripción de la prueba:</b> Comprobar que el servicio: perfil de usuario consulte los datos almacenados en la API de servicios telemáticos.
<b>Entradas/Pasos de ejecución:</b> La API de servicios telemáticos brinda al servicio: perfil de usuario información sobre los servicios telemáticos.
<b>Resultado esperado:</b> Se muestra un bloque con información de los servicios telemáticos.
<b>Evaluación:</b> Prueba satisfactoria.

### 3.6 Conclusiones parciales

La descripción de los *plugins* utilizados en el desarrollo de la propuesta de solución permitió ampliar las funcionalidades del sistema. Además la descripción del diagrama de despliegue propuesto, permitió representar el hardware utilizado en la implementación del sistema y las relaciones entre sus componentes. También como parte de la validación del sistema se realizaron pruebas a los requisitos funcionales y no funcionales a través del método de caja negra, la prueba de carga y estrés así como la de integración con otros servicios de la red social, lo que corroboró el correcto funcionamiento del sistema.

### Conclusiones generales

Se desarrolló el servicio: perfil de usuario para la red social de la Universidad de las Ciencias Informáticas de manera satisfactoria permitiendo gestionar toda la información asociada a los usuarios y posibilitando la comunicación entre ellos y el resto de los servicios.

Las características que se incluyeron en el servicio: perfil de usuario, se basan en las tendencias actuales presentes en las redes sociales estudiadas, en las necesidades propias de la institución y en los servicios que brinda, constituyeron un aporte teórico de la investigación.

La ontología FOAF para la descripción del perfil de usuario, así como el uso de la herramienta Elgg para el desarrollo de la aplicación, proporcionaron eficiencia y robustez a la solución propuesta.

El diseño y la aplicación de las pruebas de caja negra y las de carga y estrés, permitieron corroborar el correcto funcionamiento del sistema. Asimismo las pruebas de integración realizadas al sistema arrojaron resultados satisfactorios lo que demostró la integración de la aplicación desarrollada con algunos servicios de la RSU.

**RECOMENDACIONES**

Las recomendaciones que se proponen en el presente trabajo son las siguientes:

- ✓ Desarrollar la funcionalidad para obtener el currículum vitae de una persona a partir de la información que brinda.
- ✓ Definir el tipo de perfil trabajador que recoja las características que no son del perfil estudiante ni del profesor.
- ✓ Se recomienda que el servicio: perfil de usuario se integre al resto de los sistemas que componen la RSU, de forma tal que el usuario pueda elaborar su perfil de usuario una sola vez en el mismo lugar.
- ✓ Establecer un mecanismo de autenticación única para todas las aplicaciones que compone la RSU y que se acceda a través del servicio: perfil de usuario.
- ✓ Se propone continuar con la investigación para garantizar el perfeccionamiento del sistema.

## BIBLIOGRAFÍA REFERENCIADA

- ✓ (TIC), D. D. L. T. D. L. I. Y. L. C. *Las posibilidades de Mahara* [Consultado el: 24 de febrero de 2012]. Disponible en: <http://camarotic.es/?p=505>.
- ✓ ÁLVAREZ, A. N. *Uso Actual de las Redes Sociales Universitarias* [Consultado el: 16 de marzo de 2012]. Disponible en: <http://andresnunez.com/2011/02/24/uso-actual-de-las-redes-sociales-universitarias/>.
- ✓ APACHE. *Apache* [Consultado el: 27 de marzo de 2012]. Disponible en: [http://httpd.apache.org/docs/2.0/new\\_features\\_2\\_0.html](http://httpd.apache.org/docs/2.0/new_features_2_0.html).
- ✓ ARUGUETE, G. *Una propuesta organizacional alternativa*. [Consultado el: 16 de abril de 2012]. Disponible en: [http://practicasgrupales.com.ar/index.php?option=com\\_content&task=view%20&id=76](http://practicasgrupales.com.ar/index.php?option=com_content&task=view%20&id=76).
- ✓ AURELIO, M. *Redes sociales* [Consultado el: 07 de 2012]. Disponible en: [http://es.wikibooks.org/wiki/Web\\_2.0/Redes\\_Sociales](http://es.wikibooks.org/wiki/Web_2.0/Redes_Sociales).
- ✓ B., A. O. *FUNCTIONAL TESTING - PRUEBAS FUNCIONALES* [Consultado el: 11 de mayo de 2012]. Disponible en: [http://www.calidadyssoftware.com/testing/pruebas\\_funcionales.php](http://www.calidadyssoftware.com/testing/pruebas_funcionales.php).
- ✓ BERNERS-LEE. *Tejando la red*. 2000, nº
- ✓ BLOCLY. *Cómo crear una red social con un CMS libre* [Consultado el: 23 de marzo de 2012]. Disponible en: <http://blocly.com/internet/como-crear-una-red-social-con-un-cms-libre/gmx-niv80-con516.htm>.
- ✓ BOYD, D. M. y ELLISON, N. B. *Social Network Sites: Definition, History, and Scholarship* *Journal of Computer-Mediated Communication*, 2007, nº [Consultado el: 10 de febrero]. Disponible en: <http://jcmc.indiana.edu/vol13/issue1/boyd.ellison.html>.
- ✓ BURGUEÑO, P. F. *Clasificación de Redes Sociales* [Consultado el: 18 de febrero de 2012]. Disponible en: <http://www.pabloburgueno.com/2009/03/clasificacion-de-redes-sociales/>.
- ✓ CALISOFT, C. D. C. P. S. I. *Proceso de Mejora* [Consultado el: 24 de abril de 2012]. Disponible en: <http://calisoft.uci.cu/index.php/proceso-de-mejora>.
- ✓ CARVALHO, M. *The Friend of a Friend (FOAF) project* [Consultado el: 27 de marzo de 2012]. Disponible en: <http://www.foaf-project.org/original-intro>.
- ✓ CORCHUELO, R. *Introducción a la Web Semántica*. 2007, nº p. 18. [Consultado el: 18 de febrero del 2012]. Disponible en: <http://www.tdg-seville.info/Download.ashx?id=60>.
- ✓ DAVIS, I. y GALBRAITH, D. *BIO: A vocabulary for biographical information*. 2011, [Consultado el: 14 de abril del 2012]. Disponible en: <http://vocab.org/bio/0.1/.html>.

- ✓ DOMINGUE, J.; FENSEL, D., *et al. Handbook of Semantic Web Technologies*. 2011. 1077 p. ISBN 978-3-540-92912-3.
- ✓ ELGG. *elgg* [Consultado el: 24 de febrero de 2012]. Disponible en: <http://www.elgg.org/>.
- ✓ GAMMA, E.; ACEBAL, C. F., *et al. Patrones de diseño: Elementos de software orientado a objetos reutilizables*. Addison Wesley, 2003. ISBN 8478290591.
- ✓ GARCERANT, I. Tipos de prueba. 2008, nº [Consultado el: 10 de mayo del 2012]. Disponible en: [http://issuu.com/diegoma363/docs/2.tipos\\_de\\_pruebas](http://issuu.com/diegoma363/docs/2.tipos_de_pruebas).
- ✓ GONZÁLEZ, F. S. Posibilidades pedagógicas. Redes sociales y comunidades educativas. 2008, nº p. 13. [Consultado el: 20 de enero]. Disponible en: [http://www.delfos.co.cu/boletines/bsa/PDF/1redes\\_sociales\\_comunidad\\_educativa.pdf](http://www.delfos.co.cu/boletines/bsa/PDF/1redes_sociales_comunidad_educativa.pdf).
- ✓ GRACIA, J. *CMM - CMMI. Calidad. Ingeniería del Software. CMM - CMMI. Calidad. Ingeniería del Software*. [Consultado el: 26 de marzo de 2012]. Disponible en: <http://www.ingenierosoftware.com/calidad/cmm-cmmi.php>.
- ✓ GRUBER, T. R. Toward Principles for the Design of Ontologies Used for Knowledge Sharing. 1993, nº p. 23. [Consultado el: 22 de abril del 2012]. Disponible en: <http://tomgruber.org/writing/onto-design.pdf>.
- ✓ HUAMANI, J. A. R. Implementación de una ontología OWL para la sección de postgrado de la FIGMM-UNI. 2011, nº [Consultado el: 29 de marzo del 2012]. Disponible en: <http://www.figmm.uni.edu.pe/Publicaciones/articulopostfinal.pdf>.
- ✓ KLYNE, G. y CARROLL, J. J. *Resource Description Framework (RDF)* [Consultado el: 09 de febrero de 2012]. Disponible en: <http://www.w3.org/TR/rdf-concepts/>.
- ✓ LARMAN, C. Y. M. V., BEGOÑA. *UML y patrones. Una introducción al análisis y diseño orientado a objetos y al proceso unificado*. Editado por: Pearson Educación, S. A. Segunda ed. Madrid 2003. ISBN 84-205-3438-2.
- ✓ LETELIER, P. *Proceso de desarrollo de software*. En *Laboratorio de sistemas de información*. Departamento de Sistemas Informáticos y Computación. Universidad Politécnica de Valencia. . 2003.
- ✓ MOZILLA. *Concepto de JavaScript* [Consultado el: 22 de marzo de 2012]. Disponible en: [https://developer.mozilla.org/es/Gu%C3%ADa\\_JavaScript\\_1.5/Concepto\\_de\\_JavaScript](https://developer.mozilla.org/es/Gu%C3%ADa_JavaScript_1.5/Concepto_de_JavaScript).
- ✓ MYSQL. *MySQL* [Consultado el: 25 de marzo de 2012]. Disponible en: <http://www.mysql.com>.
- ✓ NETBEANS, I. *IDE Netbeans* [Consultado el: 27 de marzo de 2012]. Disponible en: <http://netbeans.org/>.

- ✓ PALACIO, J. Gestión y procesos en empresas de software 2005, nº [Consultado el: 26 de marzo del 2012].
- ✓ PARADIGM, V. *Define CRUD Chart between Activity and Class Diagram* [Consultado el: 28 de marzo de 2012]. Disponible en: <http://www.visual-paradigm.com/product/vpuml/> .
- ✓ PHP. *PHP* [Consultado el: 22 de marzo de 2012]. Disponible en: <http://www.php.net/>
- ✓ PRESSMAN, R. S. *Ingeniería del Software: Un Enfoque Práctico*. 6ta Edición ed. Nueva York, E.U.A
- ✓ Editorial McGraw-Hill. Nueva York, E.U.A, 2005. ISBN 9701054733.
- ✓ PRESSMAN, R. S. *Software Engineering, a practitioner's approach* 7ma. ed. McGraw-Hill, [Consultado el: 18 de abril de 2012]. ISBN 9780071267823.
- ✓ REILLY. What is the Web 2.0. 2005, nº [Consultado el: 18 de marzo]. Disponible en: <http://oreilly.com/web2/archive/what-is-web-20.html>.
- ✓ ROMERO, D. L. S. *Publicación de datos universitarios observando los principios de Linked Data*. UNIVERSIDAD TÉCNICA PARTICULAR DE LOJA, 2011.
- ✓ ROQUE, D. M. y PERAZA, A. G. Modelo de dominio. 2012, nº [Consultado el: 24 de marzo del 2012]. Disponible en: [http://www.ecured.cu/index.php?title=Modelo\\_de\\_dominio&oldid=1481134](http://www.ecured.cu/index.php?title=Modelo_de_dominio&oldid=1481134).
- ✓ RUIZ, R. G. *TFC: XML y Web Semántica. Estudio del impacto de las aplicaciones comerciales basadas en tecnología Web Semántica*. Tutor: Gómez, S. A. 2008.
- ✓ SALAZAR, P. H. El perfil del usuario de información. 1993, nº [Consultado el: 20 de marzo]. Disponible en: <http://www.ejournal.unam.mx/ibi/vol07-15/IBI000701502.pdf>.
- ✓ SOCIALENGINE. *Social Engine* [Consultado el: 23 de marzo de 2012]. Disponible en: <http://www.socialengine.net/features/all>.
- ✓ SOMMERVILLE, I. *Ingeniería del software*. Séptima ed. Madrid 2005.
- ✓ W3C. *Guía Breve de CSS* [Consultado el: 22 de marzo de 2012]. Disponible en: <http://www.w3c.es/Divulgacion/GuiasBreves/HojasEstilo>.
- ✓ W3C. *HTML & CSS* [Consultado el: 22 de marzo de 2012]. Disponible en: <http://www.w3.org/standards/webdesign/htmlcss>.

## BIBLIOGRAFÍA CONSULTADA

- ✓ BASTIDAS, A. Q. y CARRERA, I. B. Definición de una ontología para la guía de conocimiento Swebok. Maestría, UNIVERSIDAD DE LOS ANDES. MÉRIDA, 2007.
- ✓ BERRUETA, D. Especificación de ontologías avanzadas para la descripción del perfil de usuario. 2006, nº p. 11. [Consultado el: 26 de abril del 2012]. Disponible en: <http://forge.morfeo-project.org/wiki/images/0/04/ND.3.2.2.pdf>.
- ✓ BRICKLEY, D. y MILLER, L. FOAF vocabulary specification 0.91. 2007, nº [Consultado el: 17 de abril del 2012]. Disponible en: <http://lyle.smu.edu/~coyle/cse7347.pprev/handouts/s14.FOAF%20Vocabulary%20Specification.pdf>.
- ✓ CASTELLS, P. Sistemas Interactivos y Colaborativos en la Web. Editado por: Mancha., S. D. P. D. L. U. D. C.-L. España: 2003. 195-212 p. ISBN 84-8427-352-0.
- ✓ CORCHUELO, R. Introducción a la Web Semántica. 2007, nº p. 18. [Consultado el: 18 de febrero del 2012]. Disponible en: <http://www.tdg-seville.info/Download.ashx?id=60>.
- ✓ DOMINGUE, J.; FENSEL, D., et al. Handbook of Semantic Web Technologies. 2011. 1077 p. ISBN 978-3-540-92912-3.
- ✓ DOMÍNGUEZ, D. C. Las Redes Sociales. Tipología, uso y consumo de las redes 2.0 en la sociedad digital actual. 2010, nº p. 24. [Consultado el: 05 de febrero del 2012]. Disponible en: <http://blog.ididactic.com/wp-content/uploads/2010/12/Las-Redes-Sociales-D-Caldevilla.pdf>.
- ✓ FLORES CUETO, J. J.; MORÁN CORZO, J. J., et al. LAS REDES SOCIALES. 2010, nº [Consultado el: 09 de febrero del 2012]. Disponible en: [http://mc142.uib.es:8080/rid=1HY8TVCCBB-15599LW-1S6Z/redes\\_sociales.pdf](http://mc142.uib.es:8080/rid=1HY8TVCCBB-15599LW-1S6Z/redes_sociales.pdf).
- ✓ GARRIGÓS, D. T. Evolución de DATEX II a un modelo semántico. DEPARTAMENTO DE INFORMÁTICA. UNIVERSIDAD DE VALENCIA, 2009.
- ✓ LEBLANC, J. Programming Social Applications. United States of America. O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472., 2011. ISBN 978-1-449-39491-2.
- ✓ PEÑALVO, F. J. G. Web Semántica y Ontologías. 2005, nº p. 23. [Consultado el: 17 de marzo del 2012]. Disponible en: <http://zarza.usal.es/~fgarcia/doctorado/iuce/WSemantica.pdf>.
- ✓ PRESSMAN, R. S. Ingeniería del Software: Un Enfoque Práctico. 6ta Edición ed. Nueva York, E.U.A. Editorial McGraw-Hill. Nueva York, E.U.A, 2005. ISBN 9701054733.

- ✓ PRESSMAN, R. S. Software Engineering, a practitioner's approach. 7ma ed. McGraw-Hill, 2011. vol. 2012, ISBN 9780071267823.
- ✓ ROS MARTÍN, M. Evolución de los servicios de redes sociales en internet. El profesional de la información, 2009, vol. 18, nº 5, p. 552-558. [Consultado el: 06 de febrero del 2012]. Disponible en: <http://elprofesionaldelainformacion.metapress.com/index/Q575747035032628.pdf>. ISSN 1386-6710.
- ✓ RUIZ, R. G. TFC: XML y Web Semántica. Estudio del impacto de las aplicaciones comerciales basadas en tecnología Web Semántica. Tutor: Gómez, S. A. 2008.
- ✓ SAMPER, J.; CARRILLO, E., et al. Algoritmo de emparejamiento de perfiles en servicios web semánticos. Revista Colombiana de Computación, 2010, vol. 6, nº 1, [Consultado el: 14 de marzo del 2012]. Disponible en: [http://www.unab.edu.co/editorialunab/revistas/rcc/pdfs/r61\\_art5\\_c.pdf](http://www.unab.edu.co/editorialunab/revistas/rcc/pdfs/r61_art5_c.pdf).
- ✓ SHARMA, M. Elgg Social Networking. 2008. ISBN 978-1-847192-80-6.
- ✓ TELLO, A. L. Ontologías en la Web Semántica. 2001, nº [Consultado el: 20 de marzo del 2012]. Disponible en: [http://www.anobium.es/docs/gc\\_fichas/doc/68ERfhjkmv.pdf](http://www.anobium.es/docs/gc_fichas/doc/68ERfhjkmv.pdf).
- ✓ VALLEZ, M. La Web semántica y las tecnologías del lenguaje humano. España: Trea, publicado el: 06 de abril del 2012 de 2009, última actualización: 06 de abril del 2012. [Consultado el: 28 de abril del 2012]. Disponible en: <http://eprints.rclis.org/handle/10760/15586>. ISBN 155-180.

## ANEXOS. DESCRIPCIÓN DE REQUISITOS

Tabla 0.1 Especificación de requisitos. Autenticar usuario.

No	Nombre	Descripción	Complejidad	Prioridad para el cliente
[RF1.]	Autenticar usuario en el dominio uci.	El acceso al sistema será a través del usuario y contraseña de la cuenta uci. Se muestran los campos a llenar y además se muestra la opción para seleccionar el dominio y para iniciar la sesión. El sistema actualiza los datos y muestra un mensaje de confirmación: "Ha iniciado sesión".	Media	Alta
<b>Prototipo</b>				
				
Figura 0.1 Prototipo de interfaz. Autenticar usuario.				
Campos	Tipos de Datos		Reglas o restricciones	
Nombre del usuario uci.	String		La entrada de datos en el campo usuario debe contener solo letras, sin espacio, números y ni caracteres extraños.	
Contraseña	String		La entrada de datos en el campo contraseña debe contener solo letras, sin espacio, números y ni caracteres extraños.	
Dominio	String		La entrada de datos en el campo dominio es a través de un selector donde los dominios va a ser: Sede Central, FR Artemisa, FR Ciego de Ávila y FR Granma.	
Observaciones	1. En caso que el usuario no sea de la Sede Central debe seleccionar a cual dominio pertenece: FR Artemisa, FR Ciego de Ávila o FR Granma.			

		2. En caso de que se introduzcan los datos de usuario, contraseña o ambos mal el sistema muestra un mensaje “No pudo iniciarse la sesión. Por favor verifique que los datos introducidos estén correctos”.
--	--	--

Tabla 0.2 Especificación de requisitos. Adicionar atributo al perfil.

No	Nombre	Descripción	Complejidad	Prioridad para el cliente
[RF5.]	Adicionar atributo al perfil.	Permite adicionar un nuevo atributo al perfil. Se muestran los campos a llenar para adicionar él atributo al perfil y las opciones de Adicionar y Cancelar. Una vez creado el campo al perfil se actualiza el sistema y se muestra un mensaje de información “El atributo ha sido adicionado satisfactoriamente”.	Media	Alta
<b>Prototipo</b>				
<b>Campos</b>		<b>Tipos de Datos</b>	<b>Reglas o restricciones</b>	

Figura 0.2 Prototipo de interfaz. Adicionar atributo al perfil.

Nombre del campo	string	La entrada de datos en el campo nombre debe contener solo letras, sin espacio, números, caracteres extraños, mayúsculas, ni tildes.
Etiqueta	string	La entrada de datos en el campo etiqueta puede contener letras, números espacio, caracteres extraños.
Sugerencia	string	La entrada de datos en el campo etiqueta puede contener letras, números espacio, caracteres extraños.
Tipo de campo	Lista desplegable	Debe contener un elemento seleccionado.
Opciones	string	La entrada de datos en el campo opciones puede contener letras, números espacio, caracteres extraños.
Opciones adicionales	Lista desplegable	Debe contener un elemento seleccionado.
<b>Observaciones</b>		

Tabla 0.3 Especificación de requisitos. Editar atributos del perfil.

No	Nombre	Descripción	Complejidad	Prioridad para el cliente
[RF6.]	Editar atributo al perfil.	Permite modificar el atributo del perfil. Se muestran los campos para editar el atributo del perfil y las opciones de Guardar y Cancelar. Una vez editado el campo del perfil se actualiza el sistema y se muestra un mensaje de información "El atributo ha sido editado satisfactoriamente".	Media	Alta
<b>Prototipo</b>				

**Editar un campo de perfil** ?

Nombre del campo:

Etiqueta\*:

Sugerencia\*:

Tipo de campo:

Opciones (separadas por coma)\*:

**Opciones adicionales** ?

Mostrar en el formulario de registro:  Si desea que este campo para estar en el formulario de registro.

Obligatorio:  Si desea que este campo sea obligatorio (sólo se aplica al formulario de registro).

El usuario puede editar este campo:  Si se establece en 'No' los usuarios no podrán editar este campo (muy útil cuando los datos se gestiona en un sistema externo).

Mostrar en el perfil como etiquetas:  La salida de datos será manejada como etiquetas (sólo se aplica en el perfil de usuario).

El campo tiene una opción en blanco:  Seleccione 'Si' si una opción en blanco, debe añadirse a las opciones de campo

Campo sólo para administradores:  Seleccione 'Si' si el campo estara solo disponible para los administradores.

**Guardar**

Figura 0.3 Prototipo de interfaz. Editar atributo del perfil.

Campos	Tipos de Datos	Reglas o restricciones
Nombre del campo	string	La entrada de datos en el campo nombre debe contener solo letras, sin espacio, números, caracteres extraños, mayúsculas, ni tildes.
Etiqueta	string	La entrada de datos en el campo etiqueta puede contener letras, números espacio, caracteres extraños.
Sugerencia	string	La entrada de datos en el campo etiqueta puede contener letras, números espacio, caracteres extraños.
Tipo de campo	Lista desplegable	Debe contener un elemento seleccionado.
Opciones	string	La entrada de datos en el campo opciones puede contener letras, números espacio, caracteres extraños.
Opciones adicionales	Lista desplegable	Debe contener un elemento seleccionado.
<b>Observaciones</b>		

Tabla 0.4 Especificación de requisitos. Eliminar atributos del perfil.

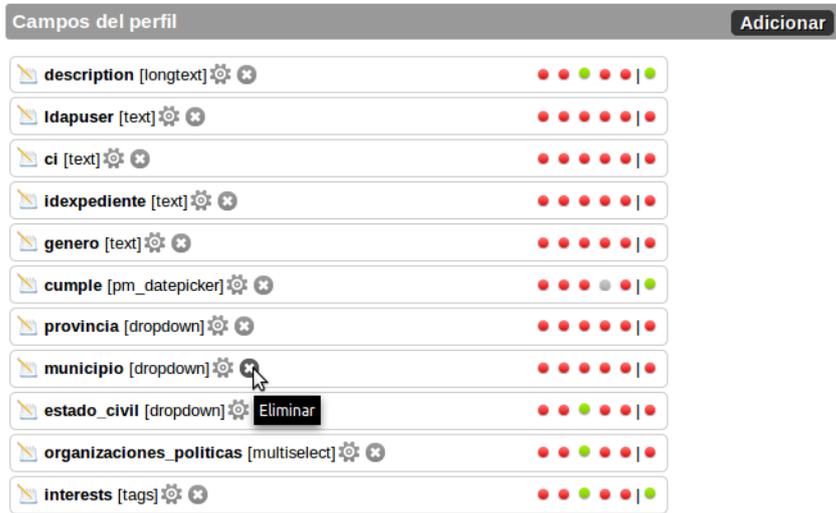
No	Nombre	Descripción	Complejidad	Prioridad para el cliente
[RF7.]	Eliminar atributos al perfil.	Permite eliminar los atributos del perfil. Muestra la opción de eliminar un atributo del perfil. Una vez eliminado el atributo el sistema actualiza los datos y emite un mensaje indicando "El atributo ha sido eliminado satisfactoriamente".	Baja	Alta
<b>Prototipo</b>				
 <p>The screenshot shows a section titled 'Campos del perfil' with an 'Adicionar' button. Below it is a list of fields, each with a name, data type, and a set of colored dots representing rules or restrictions. The fields are: description [longtext], ldapuser [text], ci [text], idexpediente [text], genero [text], cumple [pm_datepicker], provincia [dropdown], municipio [dropdown], estado_civil [dropdown], organizaciones_politicas [multiselect], and interests [tags]. A mouse cursor is hovering over the 'Eliminar' button next to the 'estado_civil' field.</p>				
Figura 0.4 Prototipo de interfaz. Eliminar atributo del perfil.				
<b>Campos</b>		<b>Tipos de Datos</b>	<b>Reglas o restricciones</b>	
		No procede	No precede	
<b>Observaciones</b>				

Tabla 0.5 Especificación de requisitos. Crear tipo de perfil.

No	Nombre	Descripción	Complejidad	Prioridad para el cliente
[RF8.]	Crear tipo de perfil.	Permite crear un nuevo tipo de perfil. Se muestran los campos a llenar y las opciones de Adicionar y Canelar. Una vez que se adicione el tipo de perfil se actualiza el sistema y muestra un mensaje de información "El perfil se ha creado satisfactoriamente".	Alta	Alta

Campos			Tipos de Datos			Reglas o restricciones		
Nombre del campo	del		string			La entrada de datos en el campo nombre debe contener solo letras, sin espacio, números, caracteres extraños, mayúsculas, ni tildes.		
Etiqueta			string			La entrada de datos en el campo etiqueta puede contener letras, números espacio, caracteres extraños.		
Descripción			string			La entrada de datos en el campo etiqueta puede contener letras, números espacio, caracteres extraños.		
Observaciones								

Figura 0.5 Prototipo de interfaz. Crear tipo de perfil.

Tabla 0.6 Especificación de requisitos. Editar tipo de perfil.

No	Nombre	Descripción	Complejidad	Prioridad para el cliente
[RF9.]	Editar tipo de perfil.	Permite modificar los datos de un tipo de perfil. Se muestran los campos a editar y las opciones de Guardar y Canelar. Una vez que se adicione el tipo de perfil se actualiza el sistema y muestra un mensaje de información "Se han guardado los cambios satisfactoriamente".	Alta	Alta
Prototipo				

Figura 0.6 Prototipo de interfaz. Editar tipo de perfil.

Campos	Tipos de Datos	Reglas o restricciones
Nombre del campo	string	La entrada de datos en el campo nombre debe contener solo letras, sin espacio, números, caracteres extraños, mayúsculas, ni tildes.
Etiqueta	string	La entrada de datos en el campo etiqueta puede contener letras, números espacio, caracteres extraños.
Descripción	string	La entrada de datos en el campo etiqueta puede contener letras, números espacio, caracteres extraños.
<b>Observaciones</b>		

Tabla 0.7 Especificación de requisitos. Eliminar tipo de perfil.

No	Nombre	Descripción	Complejidad	Prioridad para el cliente
[RF10 .]	Eliminar tipo de perfil.	Permite eliminar el perfil. Muestra la opción de Eliminar y Cancelar. Una vez eliminado el perfil el sistema actualiza los datos y muestra un mensaje de información "El atributo se ha sido eliminado correctamente".	Baja	Alta
<b>Prototipo</b>				

**Apariencia : Editar campos de perfil**



Figura 0.7 Prototipo de interfaz. Eliminar tipo de perfil.

Campos	Tipos de Datos	Reglas o restricciones
	No procede	No procede
Observaciones		

Tabla 0.8 Especificación de requisitos. Buscar amigos.

No	Nombre	Descripción	Complejidad	Prioridad para el cliente
[RF11 .]	Buscar amigos.	Permite buscar un amigo en la red social. 1. A partir de la opción buscar amigos. 2. A través del buscador de la red social. Se muestra el campo para insertar el dato (nombre, usuario, carné de identidad) del usuario que se quiere buscar.	Media	Alta

**Prototipo**

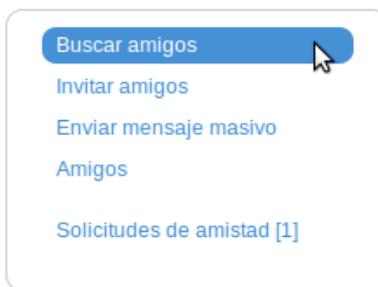


Figura 0.8 Prototipo de interfaz. Buscar amigos 1.



Figura 0.9 Prototipo de interfaz. Buscar amigos 2.

Campos	Tipos de Datos	Reglas o restricciones
	No procede	No procede
Observaciones		

Tabla 0.9 Especificación de requisitos. Mostrar perfil.

No	Nombre	Descripción	Complejidad	Prioridad para el cliente
[RF12.]	Mostrar perfil.	Se muestra el perfil de cada usuario en la red social. Se muestran las opciones de Editar perfil, Información, Muro, Amigos, Grupos, Actualizar datos UCI. Además se muestra la información personal de cada usuario y los datos del perfil.	Alta	Alta
<b>Prototipo</b>				
 <p>Figura 0.10 Prototipo de interfaz. Mostrar perfil.</p>				
<b>Campos</b>	<b>Tipos de Datos</b>	<b>Reglas o restricciones</b>		
	No procede	No procede		
<b>Observaciones</b>				

Tabla 0.10 Especificación de requisitos. Mostrar atributos del perfil.

No	Nombre	Descripción	Complejidad	Prioridad para el cliente
[RF13.]	Mostrar atributos del perfil.	Permite mostrar la información de los atributos del perfil.	Media	Alta

Prototipo		
<p>Campos del perfil <span style="float: right;">Adicionar</span></p> <ul style="list-style-type: none"> <li>description [longtext]</li> <li>Idapuser [text]</li> <li>ci [text]</li> <li>idexpediente [text]</li> <li>genero [text]</li> <li>cumple [pm_datepicker]</li> <li>provincia [dropdown]</li> <li>municipio [dropdown]</li> <li>estado_civil [dropdown]</li> <li>organizaciones_politicas [multiselect]</li> <li>interests [tags]</li> </ul>		
Figura 0.11 Prototipo de interfaz. Mostrar atributos del perfil.		
Campos	Tipos de Datos	Reglas o restricciones
	No procede	No procede
Observaciones		

Tabla 0.11 Especificación de requisitos. Mostrar tipo de perfil.

No	Nombre	Descripción	Complejidad	Prioridad para el cliente
[RF14 .]	Mostrar tipos de perfil.	Muestra la información de los tipos de perfil.	Media	Alta
<p>Tipos de perfil <span style="float: right;">Adicionar</span></p> <ul style="list-style-type: none"> <li>Estudiante ciclo profesional</li> <li>Trabajador no docente</li> <li>Profesor</li> <li>Estudiante</li> </ul>				
Figura 0.12 Prototipo de interfaz. Mostrar tipo de perfil.				
Campos	Tipos de Datos	Reglas o restricciones		
	No procede	No procede		
Observaciones				

Tabla 0.12 Especificación de requisitos. Mostrar solicitud de amistad.

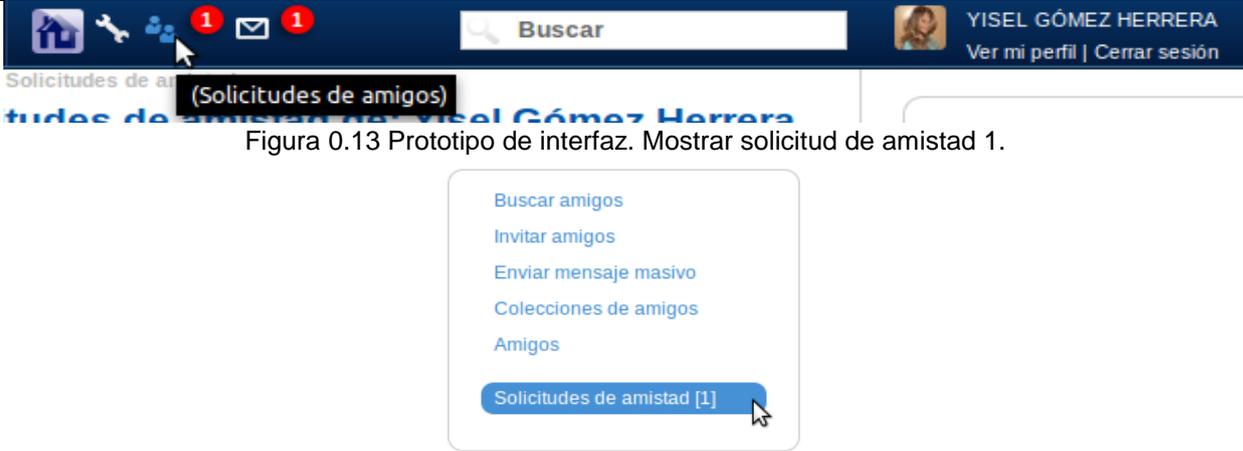
No	Nombre	Descripción	Complejidad	Prioridad para el cliente
[RF15 .]	Mostrar solicitud de amistad.	Muestra la solicitud de amistad. 1. A través de un ícono que indica la solicitud de amistad. 2. A partir de la opción Solicitud de amistad.	Media	Alta
<b>Prototipo</b>				
 <p>Figura 0.13 Prototipo de interfaz. Mostrar solicitud de amistad 1.</p> <p>Figura 0.14 Prototipo de interfaz. Mostrar solicitud de amistad 2.</p>				
<b>Campos</b>		<b>Tipos de Datos</b>	<b>Reglas o restricciones</b>	
		No procede	No procede	
<b>Observaciones</b>				

Tabla 0.13 Especificación de requisitos. Mostrar sugerencias de amigos.

No	Nombre	Descripción	Complejidad	Prioridad para el cliente
[RF16 .]	Mostrar sugerencias de amigos.	Muestra un bloque donde se encuentran las personas que quizás conozcas y al mismo tiempo brindará información de cuantos amigos en común tienen ellos contigo.	Media	Alta
<b>Prototipo</b>				

		
<p>Figura 0.15 Prototipo de interfaz. Mostrar sugerencias de amigos.</p>		
Campos	Tipos de Datos	Reglas o restricciones
	No procede	No procede
Observaciones		

Tabla 0.14 Especificación de requisitos. Mostrar información de servicios telemáticos.

No	Nombre	Descripción	Complejidad	Prioridad para el cliente
[RF17 .]	Mostrar información de servicios telemáticos.	Muestra la información de los servicios telemáticos (la cuota, usada, la cuota asignada y los correos no leídos).	Media	Alta
<b>Prototipo</b>				
				
<p>Figura 0.16 Prototipo de interfaz. Mostrar información de servicios telemáticos.</p>				
Campos	Tipos de Datos	Reglas o restricciones		
	No procede	No procede		
Observaciones				

Tabla 0.15 Especificación de requisitos. Enviar mensajes.

No	Nombre	Descripción	Complejidad	Prioridad para el cliente
[RF18 .]	Enviar mensajes.	Permite enviar mensajes a un amigo para contactarlo en la red social de las siguientes formas: 1. Enviar mensaje al correo 2. Enviar mensaje al jabber 3. Enviar mensaje interno	Media	Alta

4. Enviar mensaje a todos  
Muestra el campo de selección para escoger la manera de enviar el mensaje; además se muestran los campos a llenar en el cuerpo del mensaje y la opción de "Enviar".

### Prototipo

**Contactar a webmaster**

**Título:**  
Título del mensaje

**Mensaje:** Quitar editor

**B I U** | ABC |

Cantidad de palabras: 1 p

**Método de envío:** Correo electrónico

**Enviar**

Figura 0.17 Prototipo de interfaz. Enviar mensaje al correo.

**Contactar a webmaster**

**Título:**  
Título del mensaje

**Mensaje:** Quitar editor

**B I U** | ABC |

Cantidad de palabras: 1 p

**Método de envío:** Jabber

**Enviar**

Figura 0.18 Prototipo de interfaz. Enviar mensaje al jabber.

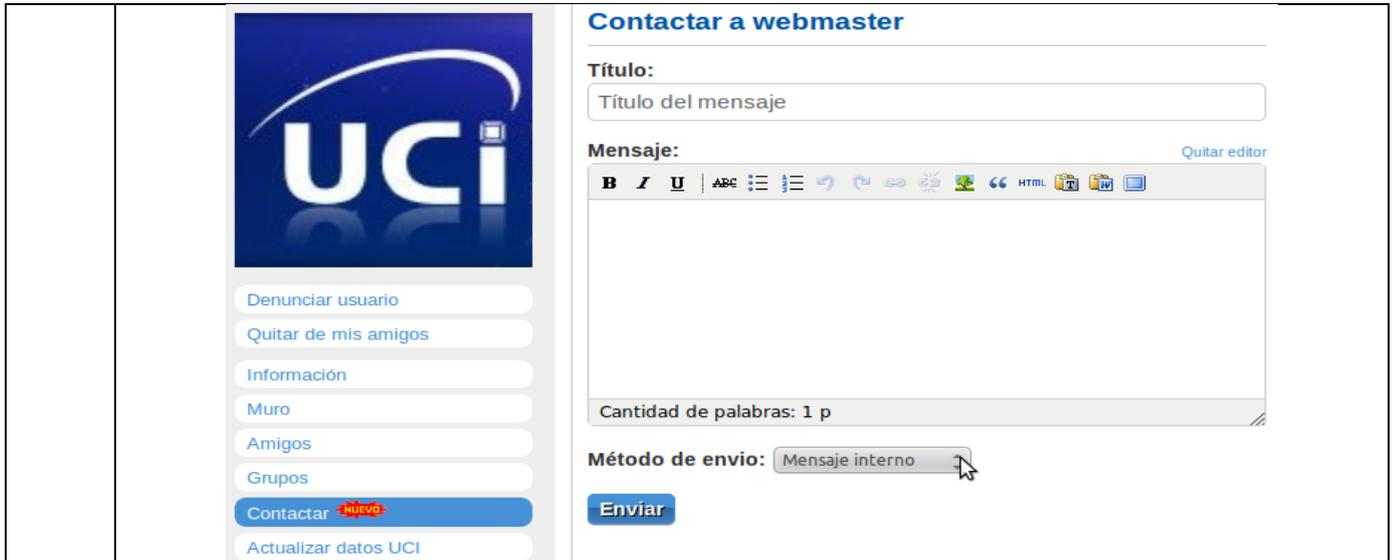


Figura 0.19 Prototipo de interfaz. Enviar mensaje interno.

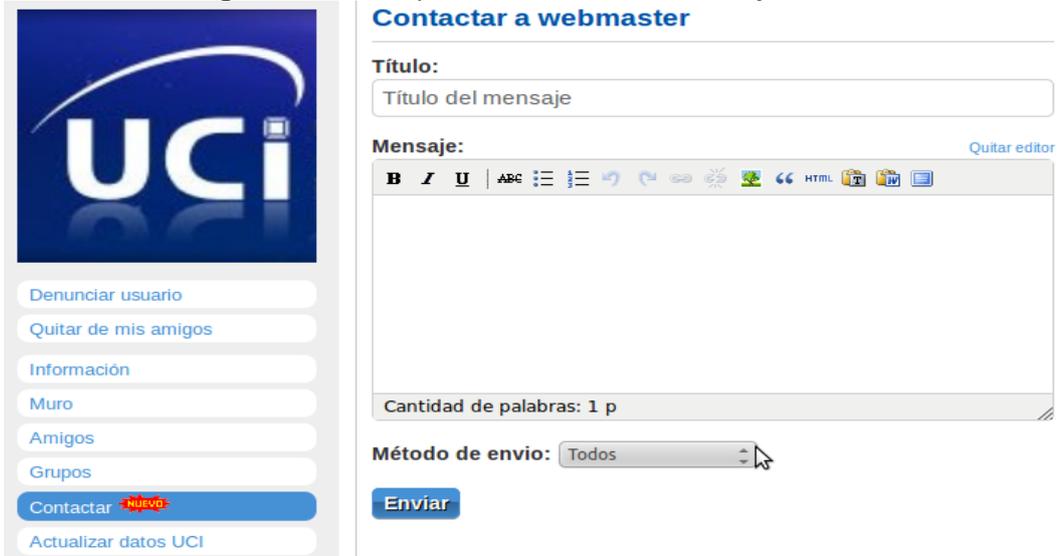


Figura 0.20 Prototipo de interfaz. Enviar mensaje a todos.

Campos	Tipos de Datos	Reglas o restricciones
	No procede	No procede
<b>Observaciones</b>		

Tabla 0.16 Especificación de requisitos. Cerrar sesión.

No	Nombre	Descripción	Complejidad	Prioridad para el cliente
[RF19 .]	Cerrar sesión	Permite desloguearse del sistema, por dos vías: 1. En el menú de navegación.	Baja	Media

2. En las opciones asociadas al usuario en el menú de navegación.

### Prototipo



Figura 0.21 Prototipo de interfaz. Cerrar sesión 1.

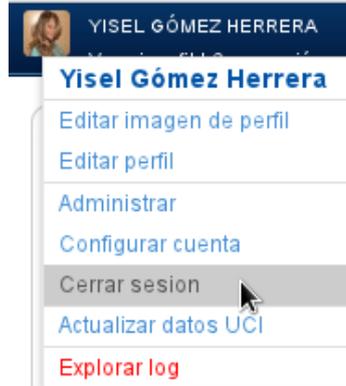


Figura 0.22 Prototipo de interfaz. Cerrar sesión 2.

Campos	Tipos de Datos	Reglas o restricciones
	No procede	No procede
Observaciones		

## ANEXOS. DESCRIPCIÓN DE CASOS DE PRUEBAS

Tabla 0.1 Diseño de casos de prueba. Autenticar usuario.

Escenario	Descripción	Usuario UCI	Contraseña	Dominio	Respuesta del sistema	Flujo central
EC 1.1 Insertar los datos correctamente.	Mediante este escenario se introducen datos correctos para autenticarse en el sistema.	V	V	V	El sistema actualiza los datos y muestra un mensaje de confirmación: "Ha iniciado sesión".	Luego de acceder al sistema el usuario selecciona la opción "Autenticarse". El sistema muestra el formulario de autenticación. El usuario llena los campos correctamente y selecciona el botón "Iniciar sesión".
		pepe	123	FR Artemisa		
		V	V	V		
		ygherrera	987456	Sede Central		
		V	V	V		
		Se introduce un usuario que esté registrado en los dominios de la institución.	Se introduce una contraseña asociada al usuario especificado.	Se selecciona el dominio en el que se encuentra registrado el usuario.		
EC 1.2 Insertar datos incorrectamente.	Mediante este escenario se introducen datos incorrectos para autenticarse en el sistema.	I	V	N/A	El sistema muestra un mensaje de alerta "No pudo iniciarse la sesión. Por favor verifique que los datos introducidos estén correctos".	Luego de acceder al sistema el usuario selecciona la opción "Autenticarse". El sistema muestra el formulario de autenticación. El usuario llena los campos incorrectamente y selecciona el botón "Iniciar sesión".
		Vacío	123			
		V	I	N/A		
		pepe	Vacío			

Tabla 0.2 Diseño de casos de prueba. Adicionar atributos al perfil 1.

Escenario	Descripción	Nombre del campo	Etiqueta	Sugerencia	Tipo de campo	Opciones	Mostrar en el formulario de registro	Obligatorio	El usuario puede editar este campo
EC 1.1 Insertar	Mediante este	V	V	V	V	V	V	V	V
		interests	Inter-	Enum-	texto	Progra-	N/A	N/A	N/A

datos correctamente.	escenario se inserta en el sistema los datos correctos para adicionarle atributos al perfil.	V	V	V	V	V	V	V	V
		Estado civil	Estado Civil	Escoja su estado civil Soltero, Casado. etc.	lista desplegable	Soltero Casado	N/A	N/A	N/A
EC 1.2 Insertar datos repetidos	Mediante este escenario se introducen datos para adicionar un atributo que ya existe en el sistema.	I	V	V	V	V	V	V	V
		interests	Preferencias	N/A	texto	verde, azul y negro	N/A	N/A	N/A
		V	I	V	V	V	V	V	V
		raza	Intereses	N/A	lista desplegable	N/A	N/A	N/A	N/A
		provincia	Provincia	N/A	texto	N/A	N/A	N/A	N/A
EC 1.3 Insertar datos incompletos	Mediante este escenario no se introducen todos los datos para adicionar un atributo en el sistema.	I	V	V	V	V	V	V	V
		Vacío	Intereses	Enumere sus intereses separados por comas.	texto	programación web, drupal	N/A	N/A	N/A
		V	I	V	V	V	V	V	V
		interests	Vacío	Enumere sus intereses separados por comas.	texto	Programación web, drupal	N/A	N/A	N/A
EC1.4 Insertar datos incorrectos	Mediante este escenario se introdu-	I	V	V	V	V	V	V	V
		fasffP12 @ 65	Intereses	Enumere sus intereses	texto	Programación web, drupal	N/A	N/A	N/A

tos	cen datos incorec- tos para para adicio- nar un atributo en el sistema.			separad os por comas.					
		V	I	V	V	V	V	V	V
		interests	N/A	Enume- re sus interese s separad os por comas.	texto	Progra- mación web, drupal	N/A	N/A	N/A
EC1.5 Cance- lar opera- ción	Se cancela la adición del atributo del perfil.	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A

Tabla 0.3 Diseño de casos de prueba. Adicionar atributos al perfil 2.

Mostrar en el perfil como etiquetas	Campo tiene una opción en blanco	Campo solo para administadore s	Respuesta del sistema	Flujo central
V	V	V	El sistema actualiza los datos y muestra un mensaje de confirmación: "El campo se ha creado satisfactoriamente".	El usuario una vez autenticado en el sistema selecciona la opción Configuración, seguidamente selecciona Administrar perfil, luego la opción Editar campos de perfil y con ella la opción "Adicionar campos del perfil". El sistema muestra las opciones de menú, el usuario llena los campos correctamente y presiona el botón "Guardar".
N/A	N/A	N/A		
V	V	V		
N/A	N/A	N/A	El sistema muestra un mensaje de error indicando: "El elemento ya existe".	El usuario una vez autenticado en el sistema selecciona la opción Configuración, seguidamente selecciona Administrar perfil, luego la opción Editar campos de perfil y con ella la opción "Adicionar campos del perfil". El sistema muestra las opciones de menú, el usuario llena los campos con un elemento que ya existe en el sistema y presiona el botón "Guardar".
V	V	V		
N/A	N/A	N/A		
V	V	V		
N/A	N/A	N/A		
V	V	V	El sistema muestra un	El usuario una vez autenticado en

N/A	N/A	N/A	mensaje en color rojo indicando: Campo requerido.	el sistema selecciona la opción Configuración, seguidamente selecciona Administrar perfil, luego la opción Editar campos de perfil y con ella la opción "Adicionar campos del perfil". El sistema muestra las opciones de menú, el usuario deja campos obligatorios vacíos y presiona el botón "Guardar".
V	V	V		
N/A	N/A	N/A		
V	V	V	El sistema muestra un mensaje indicando "No se admiten palabras que contengan mayúsculas, ni tildes, espacios, números ni caracteres extraños".	El usuario una vez autenticado en el sistema selecciona la opción Configuración, seguidamente selecciona Administrar perfil, luego la opción Editar campos de perfil y con ella la opción "Adicionar campos del perfil". El sistema muestra las opciones de menú, el usuario llena los campos de forma incorrecta y presiona el botón "Guardar".
N/A	N/A	N/A		
V	V	V		
N/A	N/A	N/A		
N/A	N/A	N/A	El sistema muestra el mensaje de confirmación ¿Está seguro de realizar la acción?	El usuario una vez autenticado en el sistema selecciona la opción Configuración, seguidamente selecciona Administrar perfil, luego la opción Editar campos de perfil y con ella la opción "Adicionar campos del perfil". El usuario llena o no los datos y presiona el botón Cancelar.

Tabla 0.4 Diseño de casos de prueba. Editar atributos del perfil 1.

Escena-rio	Descripción	Nombre del campo	Etiqueta	Sugerencia	Tipo de campo	Opciones	Mostrar en el formulario de registro	Obligatorio	El usuario puede editar este campo
EC 1.1 Editar datos correctamente	Mediante este escenario se modifica en el sistema un atributo del perfil.	V	V	V	V	V	V	V	V
		interests	Intereses	Enumere sus intereses separados por comas.	texto	programación web, drupal	N/A	N/A	N/A
		V	V	V	V	V	V	V	V
		Estado civil	Estado Civil	Escoja su estado civil Soltero, Casado.	lista desplegable	Soltero Casado	N/A	N/A	N/A

				etc.					
EC 1.2 Editar elemento repetido.	Mediante este escenario se introducen datos para editar un atributo del con datos que ya existen en el sistema.	I	V	V	V	V	V	V	V
		interests	Preferencias	N/A	texto	verde, azul y negro	N/A	N/A	N/A
		V	I	V	V	V	V	V	V
		raza	Intereses	N/A	lista desplegable	N/A	N/A	N/A	N/A
EC 1.3 Insertar datos incompletos	Mediante este escenario no se introducen todos los datos para editar un atributo en el sistema.	I	V	V	V	V	V	V	V
		Vacío	Intereses	Enumere sus intereses separados por comas.	texto	programación web, drupal	N/A	N/A	N/A
		V	I	V	V	V	V	V	V
		interests	Vacío	Enumere sus intereses separados por comas.	texto	Programación web, drupal	N/A	N/A	N/A
EC1.4 Editar datos incorrectos.	Mediante este escenario se introducen datos incorrectos para editar un atributo en el sistema.	I	V	V	V	V	V	V	V
		fasffP12 @ 65	Intereses	Enumere sus intereses separados por comas.	texto	Programación web, drupal	N/A	N/A	N/A
		V	I	V	V	V	V	V	V
		interests	N/A	Enumere sus intereses separados por comas.	texto	Programación web, drupal	N/A	N/A	N/A
EC1.5 Cancelar operación.	Se cancela la operación de editar atributos del perfil.	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A

Tabla 0.5 Diseño de casos de prueba. Editar atributo del perfil 2.

Mostrar en el perfil como etiquetas	Campo tiene una opción en blanco	Campo solo para administradores	Respuesta del sistema	Flujo central
V	V	V	El sistema actualiza el listado y muestra el mensaje "El elemento ha sido modificado satisfactoriamente".	Luego del administrador acceder al sistema solicita la opción de "Apariencia" o "Configuración ", el sistema muestra las opciones de menú, el usuario selecciona la opción "Administración del perfil ", seguidamente escoge la opción "Editar atributos de perfil". El usuario edita los datos y presiona el botón de "Guardar".
N/A	N/A	N/A		
V	V	V		
N/A	N/A	N/A		
V	V	V	El sistema muestra un mensaje de error indicando: "El elemento ya existe".	Luego del administrador acceder al sistema solicita la opción de "Apariencia" o "Configuración ", el sistema muestra las opciones de menú, el usuario selecciona la opción "Administración del perfil ", seguidamente escoge la opción "Editar atributos de perfil". El usuario edita los datos de un atributo que ya esté en el sistema y presiona el botón de "Guardar".
N/A	N/A	N/A		
V	V	V		
N/A	N/A	N/A		
V	V	V		
N/A	N/A	N/A		
V	V	V	El sistema muestra un mensaje en color rojo indicando: Campo requerido.	Luego del administrador acceder al sistema solicita la opción de "Apariencia" o "Configuración ", el sistema muestra las opciones de menú, el usuario selecciona la opción "Administración del perfil ", seguidamente escoge la opción "Editar atributos de perfil". El usuario edita los datos y deja campos obligatorios vacíos y presiona el botón "Guardar".
N/A	N/A	N/A		
V	V	V		
N/A	N/A	N/A		
V	V	V	El sistema muestra un mensaje indicando "No se admiten palabras que contengan mayúsculas, ni tildes, espacios, números ni caracteres extraños".	Luego del administrador acceder al sistema solicita la opción de "Apariencia" o "Configuración ", el sistema muestra las opciones de menú, el usuario selecciona la opción "Administración del perfil ", seguidamente escoge la opción "Editar atributos de perfil". El usuario edita los datos de forma incorrecta y presiona el botón
N/A	N/A	N/A		
V	V	V		
N/A	N/A	N/A		

				"Guardar".
N/A	N/A	N/A	El sistema muestra el mensaje de confirmación ¿Está seguro de realizar la acción?	Luego del administrador acceder al sistema solicita la opción de "Apariencia" o "Configuración ", el sistema muestra las opciones de menú, el usuario selecciona la opción "Administración del perfil ", seguidamente escoge la opción "Editar atributos de perfil". El usuario edita o no los datos y presiona el botón "Cancelar".

Tabla 0.6 Diseño de casos de prueba. Eliminar atributo del perfil.

Escenario	Descripción	Respuesta del sistema	Flujo central
EC 1.1 Eliminar atributos al perfil correctamente.	Mediante este escenario se elimina un atributo del perfil.	El sistema actualiza el listado de los atributos asociados al perfil.	Luego del administrador acceder al sistema solicita la opción de "Apariencia" o "Configuración ", el sistema muestra las opciones de menú, el usuario selecciona la opción "Administración del perfil ", seguidamente escoge la opción "Editar atributos de perfil", el sistema muestra el listado de atributos asociados al perfil, el usuario selecciona la opción "Eliminar".
EC 1.2 Cancelar la eliminación.	Mediante este escenario se cancela la eliminación de un atributo.	El sistema muestra el mensaje de confirmación ¿Está seguro de realizar la acción?	Luego del administrador acceder al sistema solicita la opción de "Apariencia" o "Configuración ", el sistema muestra las opciones de menú, el usuario selecciona la opción "Administración del perfil ", seguidamente escoge la opción "Editar atributos de perfil", el sistema muestra el listado de atributos asociados al perfil, el usuario selecciona la opción "Eliminar".

Tabla 0.7 Diseño de casos de prueba. Crear tipo de perfil.

Escenario	Descripción	Nombre del campo	Etiqueta	Descripción	Respuesta del sistema	Flujo central
EC 1.1 Crear tipo de perfil correcta-	Mediante este escenario se crea en el	V	V	V	El sistema actualiza los datos y muestra el mensaje de	El usuario una vez autenticado en el sistema selecciona la opción Configuración,
		Estudiante	Estudiante	Este perfil es de tipo estudiante.		

mente.	sistema un tipo de perfil.	V profesor	V Profesor	V Este perfil es de tipo profesor.	información "El tipo de perfil ha sido creado satisfactoriamente".	seguidamente selecciona Administrar perfil, luego la opción Editar campos de perfil y con ella la opción "Adicionar tipo de perfil". El sistema muestra las opciones de menú, el usuario llena los campos correctamente y presiona el botón "Guardar".
EC 1.2 Insertar datos repetidos.	Mediante este escenario se introducen datos para adicionar un tipo de perfil que ya existe en el sistema.	I estudiante V profesor	V Estudiante I N/A	V Este perfil es de tipo estudiante. V N/A	El sistema muestra un mensaje de error indicando: "El tipo de perfil ya existe".	El usuario una vez autenticado en el sistema selecciona la opción Configuración, seguidamente selecciona Administrar perfil, luego la opción Editar campos de perfil y con ella la opción "Adicionar tipo de perfil". El sistema muestra las opciones de menú, el usuario llena los campos con un tipo de perfil que ya existe en el sistema y presiona el botón "Guardar".
EC 1.3 Insertar datos incompletos.	Mediante este escenario no se introducen todos los datos para adicionar un tipo de perfil en el sistema.	I estudiante V profesor	V Estudiante I N/A	V Este perfil es de tipo estudiante. V N/A	El sistema muestra un mensaje en color rojo indicando: Campo requerido.	El usuario una vez autenticado en el sistema selecciona la opción Configuración, seguidamente selecciona Administrar perfil, luego la opción Editar campos de perfil y con ella la opción "Adicionar tipo de perfil". El sistema muestra las opciones de menú. El sistema muestra las opciones de menú, el usuario deja campos obligatorios vacíos y presiona el botón "Guardar".

EC1.4 Insertar datos incorrectos .	Mediante este escenario se introducen datos incorrectos para para crear un tipo de perfil en el sistema.	I	V	V	El sistema muestra un mensaje indicando "No se admiten palabras que contengan mayúsculas, ni tildes, espacios, números ni caracteres extraños".	El usuario una vez autenticado en el sistema selecciona la opción Configuración, seguidamente selecciona Administrar perfil, luego la opción Editar campos de perfil y con ella la opción "Adicionar tipo de perfil". El sistema muestra las opciones de menú. El sistema muestra las opciones de menú, el usuario llena los campos de forma incorrecta y presiona el botón "Guardar".
		fasffP12 @ 65	Estudiante	Este perfil es de tipo estudiante.		
		V	I	V		
		profesor	N/A	N/A		
EC1.5 Cancelar operación.	Mediante este escenario se cancela la adición del tipo de perfil.	N/A	N/A	N/A	El sistema muestra el mensaje de confirmación ¿Está seguro de realizar la acción?.	El usuario una vez autenticado en el sistema selecciona la opción Configuración, seguidamente selecciona Administrar perfil, luego la opción Editar campos de perfil y con ella la opción "Adicionar tipo de perfil". El sistema muestra las opciones de menú. El sistema muestra las opciones de menú, el usuario llena o no los campos y presiona el botón "Cancelar".

Tabla 0.8 Diseño de casos de prueba. Editar tipo de perfil.

Escenario	Descripción	Nombre del campo	Etiqueta	Descripción	Respuesta del sistema	Flujo central
EC 1.1 Editar perfil correctamente.	Mediante este escenario se edita en el sistema un tipo de perfil.	V	V	V	El sistema actualiza los datos y muestra un mensaje de confirmación: "El perfil ha sido editado"	El usuario una vez autenticado en el sistema selecciona la opción Configuración, seguidamente selecciona Administrar perfil, luego la opción
		Estudiante	Estudiante	Este perfil es de tipo estudiante.		
		V	V	V		
		profesor	Profesor	Este perfil es de tipo		

				profesor.	correctamente".	Editar campos de perfil y con ella la opción "Adicionar tipo de perfil". El sistema muestra las opciones de menú, el usuario edita los datos correctamente y presiona el botón "Guardar".
EC 1.2 Insertar datos repetidos.	Mediante este escenario se introducen datos para adicionar un tipo de perfil que ya existe en el sistema.	I	V	V	El sistema muestra un mensaje de error indicando: "El tipo de perfil ya existe".	El usuario una vez autenticado en el sistema selecciona la opción Configuración, seguidamente selecciona Administrar perfil, luego la opción Editar campos de perfil y con ella la opción "Adicionar tipo de perfil". El sistema muestra las opciones de menú, el usuario llena los campos con un tipo de perfil que ya existe en el sistema y presiona el botón "Guardar".
		estudiante	Estudiante	Este perfil es de tipo estudiante.		
		V	I	V		
		profesor	N/A	N/A		
EC 1.3 Insertar datos incompletos.	Mediante este escenario no se introducen todos los datos para adicionar un tipo de perfil en el sistema.	I	V	V	El sistema muestra un mensaje en color rojo indicando: Campo requerido.	El usuario una vez autenticado en el sistema selecciona la opción Configuración, seguidamente selecciona Administrar perfil, luego la opción Editar campos de perfil y con ella la opción "Adicionar tipo de perfil". El sistema muestra las opciones de menú. El sistema muestra las opciones de menú, el usuario deja campos obligatorios vacíos y presiona el botón "Guardar".
		Vacío	Estudiante	Este perfil es de tipo estudiante.		
		V	I	V		
		profesor	N/A	N/A		
EC1.4 Insertar datos	Mediante este escenario se	I	V	V	El sistema muestra un mensaje	El usuario una vez autenticado en el sistema selecciona la
		fasffP12 @ 65	Estudiante	Este perfil es de tipo		

incorrectos	introducen datos incorrectos para para editar un tipo de perfil en el sistema.			estudiante.	indicando "No se admiten palabras que contengan mayúsculas, ni tildes, espacios, números ni caracteres extraños".	opción Configuración, seguidamente selecciona Administrar perfil, luego la opción Editar campos de perfil y con ella la opción "Adicionar tipo de perfil". El sistema muestra las opciones de menú. El sistema muestra las opciones de menú, el usuario llena los campos de forma incorrecta y presiona el botón "Guardar".
		V	I	V		
		profesor	N/A	N/A		
EC1.5 Cancelar operación.	Mediante este escenario se cancela la edición del tipo de perfil.	N/A	N/A	N/A	El sistema muestra el mensaje de confirmación ¿Está seguro de realizar la acción?	El usuario una vez autenticado en el sistema selecciona la opción Configuración, seguidamente selecciona Administrar perfil, luego la opción Editar campos de perfil y con ella la opción "Adicionar tipo de perfil". El sistema muestra las opciones de menú. El sistema muestra las opciones de menú, el usuario llena o no los campos y presiona el botón "Cancelar".

Tabla 0.9 Diseño de casos de prueba. Eliminar tipo de perfil.

Escenario	Descripción	Respuesta del sistema	Flujo central
EC 1.1 Eliminar tipo de perfil correctamente.	Mediante este escenario se elimina un tipo de perfil.	El sistema actualiza el listado de perfiles y muestra un mensaje indicando "El tipo de perfil ha sido eliminado satisfactoriamente".	Luego del administrador acceder al sistema solicita la opción de "Apariencia" o "Configuración ", el sistema muestra las opciones de menú, el usuario selecciona la opción "Administración del perfil ", seguidamente escoge la opción "Editar atributos de perfil", el sistema muestra el listado de tipos de perfiles, el usuario selecciona la

			opción "Eliminar".
EC 1.2 Cancelar la eliminación.	Mediante este escenario se cancela la eliminación de un atributo.	El sistema muestra el mensaje de confirmación ¿Está seguro de eliminar este tipo de perfil?	Luego del administrador acceder al sistema solicita la opción de "Apariencia" o "Configuración ", el sistema muestra las opciones de menú, el usuario selecciona la opción "Administración del perfil ", seguidamente escoge la opción "Editar atributos de perfil", el sistema muestra el listado de tipos de perfiles, el usuario selecciona la opción "Eliminar".