

**Universidad de las Ciencias Informáticas**

**Facultad 1**



**Título: Herramienta de comunicación de la Web 2.0 para la  
Universidad de las Ciencias Informáticas**

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

**Autores:**

Guillermo Alejandro Pino González

Abel Ochoa Izquierdo

**Tutor:**

MsC. Ariel Vázquez Riverón

La Habana, 2015

"Año 57 de la Revolución"

*“Las nuevas tecnologías de las comunicaciones han dividido al mundo entre los conectados y los no conectados a las redes globales.”*

*Fidel Castro*



*Dedicatoria*

*A nuestros padres, nuestro ejemplo a seguir.*

## *Agradecimientos*

*De Abel:*

*A Guillermo mi compañero de tesis por su esfuerzo y dedicación. A mi familia por el apoyo que me brinda. A mi hermano Isbel por estar ahí cuando más lo necesitaba. A Angel Antonio por ser la mayor ayuda en esta tesis. Al equipo #1 por 5 años de trabajo. A mis colegas de siempre y al tribunal por ayudarnos y corregirnos en todo lo que pudieron.*

*GRACIAS A TODOS!!!!!!!!!!!!!!*

*De Guillermo:*

*Agradezco primeramente a mi compañero de tesis por la ayuda que brindó y el esfuerzo para el desarrollo del trabajo. A mi familia y en especial a mi querida mamá, a mi papá, hermanos, abuelos y a mi nena por el apoyo incondicional que me dan. A mis amigos que son muchos, pero buenos. A mi tutor y todo el que de una forma u otra me ayudo para poder alcanzar esta meta.*

Declaración de autoría.

Por este medio declaramos que Abel Ochoa Izquierdo y Guillermo Alejandro Pino González somos los únicos autores de este trabajo y autorizamos a la Universidad de las Ciencias Informáticas (UCI) para que haga el uso que estime pertinente del mismo. Para que así conste firmamos la presente a los \_\_\_\_ días del mes de \_\_\_\_ del año \_\_\_\_\_.

Abel Ochoa Izquierdo

Guillermo Alejandro Pino González

\_\_\_\_\_

\_\_\_\_\_

Firma del autor Firma del autor

MsC. Ariel Vázquez Riverón

\_\_\_\_\_

Firma del tutor

## **Resumen**

La necesidad de comunicación de las personas junto al fenómeno de la Web 2.0 y su acelerado progreso, han generado nuevos modelos de producir y compartir información que demandan una valoración dentro del ámbito universitario. A través de la Web 2.0 se presentan nuevas formas revolucionarias de crear, colaborar, editar y compartir en línea el contenido generado por el usuario. Por tanto el objetivo de este trabajo es presentar una herramienta informática que combine herramientas de las la Web 2.0 que contribuyan al proceso de la comunicación interna en la Universidad de las Ciencias Informáticas. En el presente trabajo se realizó un estudio de conceptos relacionados a la comunicación y especialmente a la comunicación interna, así como a las características, funcionalidades y potencialidades que presentan las herramientas de la Web 2.0. A lo anterior se suma el análisis de varias plataformas para el desarrollo de este tipo de aplicaciones, las herramientas utilizadas y la metodología empleada. La herramienta propuesta fue sometida a diferentes pruebas que validan el cumplimiento de los requisitos definidos y permitirá ofrecer un punto de encuentro donde las personas se convierten en protagonistas activos, creando y compartiendo contenidos, opinando, participando y relacionándose en tiempo real.

**Palabras clave:** Web 2.0, comunicación interna.

## Contenido

<b>INTRODUCCIÓN</b> .....	1
<b>CAPÍTULO I: FUNDAMENTACIÓN TEÓRICA</b> .....	6
1.1 CONCEPTOS FUNDAMENTALES .....	6
1.2 HERRAMIENTAS DE COMUNICACIÓN EN LA WEB 2.0 .....	9
1.2.1 Sitios web destinados a la comunicación .....	15
1.2.2 A nivel nacional .....	17
1.2.3 Resultados del estudio realizado .....	19
1.3 HERRAMIENTAS DE COMUNICACIÓN INTERNA EN LA UNIVERSIDAD .....	20
1.4 SOFTWARE UTILIZADO .....	22
1.4.1 Plataformas para desarrollar el sistema propuesto .....	22
1.4.2 Resultado del análisis de las plataformas .....	29
1.5 LENGUAJES Y HERRAMIENTAS DE DESARROLLO .....	30
1.5.1 Lenguajes de programación .....	30
1.5.2 Lenguajes de marcado .....	30
1.5.3 Lenguaje de modelado .....	31
1.5.4 Otras Tecnologías .....	31
1.5.5 Herramientas .....	32
1.6 METODOLOGÍA .....	34
1.7 CONCLUSIONES PARCIALES .....	39
<b>CAPÍTULO II: DESCRIPCIÓN DE LA PROPUESTA DE SOLUCIÓN</b> .....	40
2.1 RED SOCIAL DE MICROBLOGGING .....	40
2.2 REQUISITOS FUNCIONALES .....	44
2.3 REQUISITOS NO FUNCIONALES .....	45
2.4 ROLES DEL SISTEMA .....	47
2.5 DESCRIPCIÓN DE LA ARQUITECTURA .....	47
2.5.1 Arquitectura cliente-servidor .....	47
2.5.2 Arquitectura de Exo platform .....	48
2.5.3 Diagrama de despliegue .....	51

2.6 DESCRIPCIÓN DEL DISEÑO .....	53
2.6.1 Patrones de diseño GoF .....	53
2.6.2 Patrones de base de datos .....	56
2.7 CONCLUSIONES PARCIALES .....	57
<b>CAPÍTULO III: VALIDACIÓN DE LA PROPUESTA DE SOLUCIÓN .....</b>	<b>58</b>
3.1 ESTÁNDARES DE CODIFICACIÓN.....	58
3.2 VALIDACIÓN .....	60
3.2.1 Pruebas de software .....	61
3.2.2 Pruebas de aceptación .....	65
3.3 CONCLUSIONES PARCIALES.....	65
CONCLUSIONES GENERALES.....	66
RECOMENDACIONES .....	67
BIBLIOGRAFÍA.....	68

## **Introducción**

La comunicación es uno de los pilares que ha permitido que la humanidad se desarrolle. El hombre, como ser social, se relaciona con el mundo natural y el creado por él mismo; de modo que en el transcurso de su desarrollo individual, se apropia del conocimiento a través de la actividad y la comunicación con otras personas. La necesidad de transmitir ideas y debatir opiniones en los distintos entornos en que se manifiesta durante su vida, da lugar a relaciones con personas que a su vez interactúan con otras a quienes no conocen, creando lazos o vínculos (1).

La comunicación es un instrumento social importante, mediante el que se puede influir en las ideas, en los sentimientos o en las acciones de las personas. La comunicación logra vincular al individuo consigo mismo, con su entorno y con los demás hombres (1).

En la actualidad el desarrollo de la red de redes ha ido incorporando servicios que potencian la comunicación, al facilitar el acceso a la información y ampliar las posibilidades de comunicación sincrónica y asincrónica entre usuarios; con herramientas como el chat, las videoconferencias y el correo electrónico, por solo mencionar algunas.

La Universidad de las Ciencias Informáticas (UCI) cuenta con una infraestructura tecnológica para dar servicio a una considerable cantidad de usuarios de las Tecnologías de la Informática y las Comunicaciones (TIC), quienes tienen acceso a la navegación en Internet e intranet cubana, así como a los servicios telemáticos y a una variedad de aplicaciones de uso interno. Como institución, la UCI posee una estructura organizativa con canales de comunicación y se apoya en diversos medios de comunicación informativos y vías de difusión de la información como la televisión, la radio universitaria, los portales web y los servicios de correo electrónico y chat.

Uno de los fenómenos que se presenta frecuentemente en el proceso de comunicación interna en la organización se conoce con el nombre de información seriada y consiste en la repetición de un mismo mensaje a través de distintos sujetos. Estas cadenas comunicacionales que se originan reciben el nombre de redes comunicativas de la institución y son los canales por los que normalmente circula la información. Cuando dichas cadenas llegan a ser excesivamente largas puede originarse dificultad y distorsión en el proceso de comunicación interna.

En la UCI el problema antes descrito se evidencia cuando el Consejo Universitario (CU) toma una decisión que debe ser de conocimiento de todos los miembros de la institución. Cada directivo allí presente comunica a sus subordinados inmediatos, los cuales tienen a su vez otros subordinados que son informados y así sucesivamente. De este modo la información se transmite de forma piramidal y en cadenas que puede alcanzar alrededor de cinco eslabones. Esto trae como consecuencias que los mensajes a medida que se transmiten por las redes comunicativas de la institución tienden a perder detalles.

Facilitar el acceso a la información directamente de su fuente original de forma rápida y efectiva empleando los mecanismos actuales disponibles (correo electrónico, portales, radio universitaria, canal de televisión interna, etcétera) no siempre es posible ya sea por limitaciones de dichos mecanismos o por la complejidad de sus operaciones hasta lograr la publicación de la información, lo que en la mayoría de los casos atenta contra la inmediatez.

Otra de las insuficiencias identificadas en el proceso de comunicación se refiere a la poca participación e interacción de los miembros de la institución en tiempo real debido a restricciones que impone la dirección de los canales de comunicación.

El uso de la tecnología por sí sola no es capaz de solucionar los problemas que presenta la comunicación en sí, pero las herramientas informáticas se pueden especializar en dependencia de las necesidades de comunicación interna. Hoy es una realidad que los servicios de redes sociales están en camino de sustituir al correo electrónico como principal método de comunicación, las redes sociales corporativas son avaladas por el enorme potencial que supone formar parte de un entorno de trabajo colaborativo. Con estas plataformas el individuo tiene la capacidad de participar en una conversación basada en la inteligencia compartida.

Por los argumentos anteriormente expuestos se presenta como **problema a resolver**: ¿Cómo proveer al proceso de comunicación interna en la Universidad de las Ciencias Informáticas de una aplicación que combine herramientas de la Web 2.0? Se define por tanto como **objeto de estudio** aplicaciones informáticas que combinen herramientas de comunicación de la Web 2.0, enmarcado en el **campo de acción** el proceso de comunicación interna en la Universidad de las Ciencias Informáticas. Con el propósito de resolver el problema planteado se precisa como **objetivo general**: obtener una solución tecnológica para la Web 2.0 de bajo costo de implantación y fácil de usar como canal de comunicación interna en la UCI. Para dar cumplimiento al objetivo general se desglosan los siguientes **objetivos específicos**:

1. Caracterizar la comunicación interna desde sus referentes teóricos.
2. Caracterizar herramientas de comunicación de la web 2.0.
3. Implementar la herramienta informática propuesta.
4. Evaluar la solución mediante el uso de pruebas funcionales, de aceptación y de carga y estrés.

### **Idea a defender**

Una solución informática que emplee herramientas de la Web 2.0 para la comunicación, provee un espacio virtual donde las personas pueden intercambiar conocimiento y experiencias de forma permanente y en tiempo real.

Para lograr una solución y darle cumplimiento a los objetivos general y específicos se llevaran a cabo las siguientes **tareas de investigación**.

- Análisis de los conceptos y elementos de la comunicación interna.
- Análisis de las tendencias actuales del uso de sistemas para la comunicación en la web 2.0.
- Estudio de homólogos de sistemas de comunicación en la web para la comunicación interna.
- Selección de las herramientas y tecnologías informáticas a utilizar para el desarrollo del *software* propuesto.
- Identificación y especificación de los requisitos funcionales y no funcionales.
- Implementación de los requisitos especificados.
- Diseño de las pruebas a realizar al sistema.
- Realización de pruebas al sistema.

Para el desarrollo de esta investigación se utilizaron diferentes **métodos científicos**:

- **Teóricos**

*Analítico-Sintético*: Constituyen dos procesos cognitivos que cumplen funciones muy importantes en la investigación científica. El análisis es una operación intelectual que posibilita descomponer

mentalmente un todo complejo en sus partes y cualidades. El análisis permite la división mental del todo en sus múltiples relaciones y componentes. La síntesis es la operación inversa, que establece mentalmente la unión entre las partes, previamente analizadas y posibilita descubrir relaciones y características generales entre los elementos de la realidad (2).

Con este método se logró obtener gran cantidad de información y a partir de la misma resumir, analizar y extraer los conocimientos necesarios y elementos esenciales para el desarrollo de una red social de *microblogging* que combine otras herramientas de la Web 2.0 y permita dar solución al problema planteado.

*Histórico-lógico*: El método histórico estudia la trayectoria real de los fenómenos y acontecimientos en el decursar de su historia. El método lógico investiga las leyes generales del funcionamiento y desarrollo de los fenómenos. Lo lógico no repite lo histórico en todos sus detalles, sino que reproduce en el plano teórico lo más importante del fenómeno, lo que constituye su esencia (2).

Este método permitió evaluar y estudiar el avance alcanzado por los medios de comunicación de la Web 2.0 y su impacto en la comunicación interna de una organización. Además posibilitó estudiar las diferentes herramientas utilizadas para la creación de sitios de esta índole.

- **Empírico**

*Observación*: Es uno de los métodos más utilizados en la investigación científica, debido a que es un procedimiento fácil de llevar a cabo y que exige técnicas de tabulación muy sencillas. De la misma forma permite percibir directamente, sin intermediarios que deformen la percepción, los hechos de la realidad objetiva, con lo cual se eliminan las deformaciones de otros métodos indirectos (2).

Este método se empleó para comparar y analizar el comportamiento de los medios de comunicación en la Web 2.0 y de las herramientas que se emplean para la creación de los mismos.

*Encuesta*: Se utiliza cuando la información que se realiza puede ser obtenida a partir de la respuesta de una o varias personas a través del uso de un cuestionario pre-elaborado. La encuesta es semejante a la entrevista pero escrita, donde a través de un conjunto de preguntas se pretende obtener una información sobre el mundo interior del encuestado o su percepción del fenómeno que se investiga, por

lo que no puede ser obtenida por observación. Se empleó el método anterior para conocer las opiniones e inquietudes de los miembros de la Universidad sobre los medios de comunicación que utilizan.

Como **posibles resultados** se espera:

Una herramienta informática que permita ofrecer un punto de encuentro donde las personas podrán intercambiar conocimientos, experiencias y valoraciones.

### **Estructura del documento:**

El trabajo se estructuró en tres capítulos los cuales abarcan todo el estudio, investigación y desarrollo realizado para dar solución al problema que se presenta.

### **Capítulo I: Fundamentación teórica**

En este capítulo se plantea la infraestructura teórica que soporta toda la investigación. Se explican todos los términos y definiciones que se relacionan con el estudio del tema abordado. También se realiza un estudio sobre las diferentes herramientas que pueden utilizarse para desarrollar la solución al problema planteado.

### **Capítulo II: Descripción de la propuesta de solución**

En el capítulo se describe de manera detallada la propuesta de solución. Se exponen los requisitos funcionales y no funcionales identificados para llevar a cabo su desarrollo. Se determinan los actores del sistema y la relación entre ellos. Además, se realiza la descripción textual de las historias de usuario más significativas y se define la arquitectura de la propuesta de solución.

### **Capítulo III: Validación de la propuesta de solución**

En este capítulo se realiza la validación de los requisitos y las historias de usuario mediante la aplicación de pruebas de *software* y de aceptación. Además se describe como fue implementada la solución al problema planteado teniendo en cuenta las buenas prácticas para el desarrollo de *software*.

## **Capítulo I: Fundamentación teórica**

En este capítulo se exponen un conjunto de conceptos y características esenciales que fundamentan lo investigado. Incluye un estudio sobre varias definiciones relacionadas con la comunicación y la comunicación organizacional interna. Se hace un análisis de las tendencias actuales referente a los medios de comunicación a través de la Web 2.0, así como de aplicaciones para su desarrollo. Además se plasman los resultados de la investigación y se describen las tecnologías, metodologías, herramientas y lenguajes de programación utilizados.

### **1.1 Conceptos fundamentales**

Dado que el punto fundamental de este análisis es la representación del proceso de comunicación organizacional interna haciendo uso de herramientas informáticas actualizadas, será necesario plantear algunos aspectos que sirvan de ejes conceptuales sobre los que apoyar el entendimiento del proceso de comunicación y sus principales elementos a partir del uso de la tecnología.

#### **Comunicación**

Fernando Rey González en su libro "Comunicación personalidad y desarrollo" define la comunicación como el proceso de transmisión y recepción de ideas, información y mensajes. El acto de comunicar es un proceso complejo en el que dos o más personas se relacionan y a través de un intercambio de mensajes con códigos similares, tratan de comprenderse e influirse de forma que sus objetivos sean aceptados en la forma prevista, utilizando un canal que actúa de soporte en la transmisión de la información. Es más un hecho sociocultural que un proceso mecánico (3).

Para María del Socorro Fonseca, comunicar es "llegar a compartir algo de nosotros mismos. Es una cualidad racional y emocional específica del hombre que surge de la necesidad de ponerse en contacto con los demás, intercambiando ideas que adquieren sentido o significación de acuerdo con experiencias previas comunes" (4).

Según Idalberto Chiavenato, comunicación es "el intercambio de información entre personas. Significa volver común un mensaje o una información. Constituye uno de los procesos fundamentales de la experiencia humana y la organización social" (5).

Luego de haber analizado las definiciones expuestas anteriormente se puede concluir que sus autores coinciden en que la comunicación es: el proceso mediante el cual el emisor y el receptor establecen una conexión en un momento y espacio determinados para transmitir, intercambiar o compartir ideas, información o significados que son comprensibles para ambos.

### **Comunicación organizacional**

Fernández Collado define la comunicación organizacional como “el conjunto total de mensajes que se intercambian entre los integrantes de una organización, y entre ésta y su medio”, también la entiende como: “Un conjunto de técnicas y actividades encaminadas a facilitar y agilizar el flujo de mensajes que se dan entre los miembros de la organización, entre la organización y su medio; o bien, influir en las opiniones, aptitudes y conductas de los públicos internos y externos de la organización, todo ello con el fin de que ésta última cumpla mejor y más rápido los objetivos” (6).

### **Comunicación organizacional interna**

Actividades que se realizan dentro de una organización para mantener las buenas relaciones entre los miembros de la institución por medio de la circulación de mensajes originados por los diversos medios de información, con el objetivo de proveer comunicación, unión, participación, motivación y así alcanzar las metas establecidas por la organización (6).

### **Flujos de comunicación interna**

Los flujos de comunicación son tomados en cuenta a partir de la estructura comunicacional a nivel interno dentro de las organizaciones. La comunicación interna está dispuesta en tres formas principales: descendente, ascendente y horizontal. Cada una de ellas obedece a la razón de ser; es decir la comunicación que se da a partir de las disposiciones de la gerencia, la participación de los colaboradores en las decisiones y el flujo de información entre las unidades de la organización respectivamente.

#### **Comunicación descendente**

La comunicación descendente se ocupa de comunicar el orden establecido para el desarrollo de tareas, roles, entrenamiento, capacitación, directrices y políticas.

Su carácter principal se establece por la naturaleza de la organización, dar orden, no solamente desde el sentido imperativo de quien da una orden, sino desde el sentido de organizar y ordenar. Una de las principales funciones de la comunicación descendente es la de comunicar la cultura organizacional y las directrices de cómo cumplirlas.

#### Comunicación ascendente

La comunicación ascendente es aquella que, al contrario de la descendente se da desde los niveles jerárquicos más bajos de la organización hacia los niveles más altos , este flujo de comunicación permite que los colaboradores trabajen de manera pro activa en la toma de decisiones y participen de manera constante en la creación e innovación de la organización, hay quienes consideran que la comunicación ascendente es indispensable pues desde aquí se habla de retroalimentación constante y se fundamenta en el conocimiento mutuo de pensamientos y acciones.

#### Comunicación horizontal

La comunicación horizontal es aquella que se establece entre miembros de un mismo nivel jerárquico. Pueden ser entre departamentos, grupos o de forma individual, no hay presencia de autoridad y sirven para agilizar la estructura organizativa. Ese tipo de información se puede obtener a través de juntas, informes y asambleas.

Se da entre los empleados de un mismo nivel jerárquico.

Y se utiliza para:

- Coordinar las actividades de los distintos empleados o departamentos de la organización.
- Resolver problemas de un departamento.
- Tomar decisiones en las que tengan que intervenir varios departamentos.
- Se usan reuniones de trabajo, notas, comunicados internos, trabajo en equipo, debates, teléfono e Internet.

## 1.2 Herramientas de comunicación en la Web 2.0

Las herramientas de comunicación en línea en su mayoría, gratuitos y de fácil acceso, permiten participar en la producción y expansión de conocimientos. Asimismo, sirven como medio para desarrollar las habilidades comunicativas, las auditivas y las habilidades de lecto-escritura, además de que pueden resultar de mucho valor para una organización.

### Web 2.0

El término Web 2.0 surgió hace pocos años para determinar el fenómeno social basado en la interacción de diferentes aplicaciones web centradas en el usuario las cuales facilitan el intercambio de información, la colaboración e interactividad multimedia en tiempo real. El término se asocia estrechamente con Tim O'Reilly, debido a la conferencia sobre la Web 2.0 que impartió en 2004 acuñando el nuevo concepto. Aunque el término sugiere una nueva versión de la *World Wide Web*, no se refiere a una actualización de las especificaciones técnicas de la web, sino más bien a cambios acumulativos en la forma en la que desarrolladores de *software* y usuarios finales utilizan la Web.

Además de las redes sociales o los espacios de almacenamiento de fotos y videos, son también ejemplos de herramientas de las Web 2.0 los blogs, los microblogging y los espacios de contenidos colaborativos como las wikis. Un sitio de la Web 2.0 permite a sus usuarios interactuar con otros usuarios o modificar su contenido, en contraste a sitios web no-interactivos, donde los usuarios se limitan a la visualización pasiva de información que se les proporciona.

En la Web 2.0 el usuario es creador y protagonista. En este nuevo entorno, los usuarios no son meros receptores. Las tecnologías digitales han facilitado que usuarios de todo el mundo puedan relacionarse y compartir opiniones y experiencias.

Las herramientas de la web 2.0 permiten incrementar el *know-how* de la organización al almacenar las conversaciones (mientras que con el teléfono, los e-mails y los chats se pierden o queda sólo en el ordenador del usuario). Esto posibilita que la información y el conocimiento que se generan sean compartidos, para ser utilizados por todos, de manera que se fomente el sentido de pertenencia y se contribuya la mejora de la organización. A lo anterior se suma la rapidez que facilitan estas herramientas al solucionar conflictos y resolver problemas promoviendo el trabajo en equipo y la participación en proyectos comunes.

## **Foro**

Los foros de Internet, son aplicaciones que permiten expresar opiniones o participar de debates por medio de la Web. Funcionan a partir de un mensaje que es publicado por un usuario o moderador y que genera respuestas por parte del resto de los usuarios. Su diseño permite seguir el hilo de la conversación desde el mensaje original hasta las respuestas más recientes (7).

En las organizaciones los foros son instrumentos de utilidad para las comunidades de usuarios, clientes o personas interesadas dado que puedan tener un punto de encuentro para compartir sus conocimientos con libertad absoluta.

Algunas de las ventajas que presenta el foro son:

- Los usuarios comparten opiniones, experiencias y dudas sobre un tema específico.
- Permite que debates de interés sean más abiertos, cualquier persona registrada puede participar de este.
- Las publicaciones de usuarios permiten a los demás mantenerse informados sobre temas específicos o de actualidad.

A lo anterior se contraponen las siguientes dificultades:

- No todos los usuarios que opinan tienen conocimiento suficiente del tema a tratar.
- Existe inseguridad ya que no siempre se conoce las personas con la que se debate.
- Se puede encontrar información inadecuada y en términos no respetuosos.

## **Blog**

La Real Academia de la Lengua Española (RAE) define blog como un sitio web que incluye, a modo de diario personal de su autor o autores, contenidos de su interés, actualizados con frecuencia y a menudo comentados por los lectores (8).

Los blogs de una organización permiten a sus empleados expresarse con un lenguaje más cercano y no tan distante como el corporativo. También suponen otra manera de vender productos e imagen, de ganar

clientes y de fidelizarlos. Además, gracias a los comentarios, los visitantes pueden expresar sus opiniones, algo de suma relevancia para cualquier empresa ya que les puede ahorrar mucho tiempo y trabajo en consultas (7).

El éxito y desarrollo de los blogs se debe a tres factores:

Usabilidad: es fácil bloguear y no se precisan especiales habilidades (técnicas) para hacerlo.

Colaboración: la gente conecta con otras personas, comparte intereses y debate sobre temas concretos.

Personal: se realiza bajo una visión personal (sobre todo los blogs no corporativos), de forma que lo que se publica responde a opiniones propias del autor.

A partir del estudio de los blog y la interacción con estos se definen las siguientes ventajas:

- Ayuda al ejercicio creativo, a generar debates, opiniones y soluciones a problemas.
- Es útil para conocer personas y formarse en diferentes temas.
- Un blog puede ser colectivo integrando temáticas interesantes y productivas.
- El administrador puede tomar la decisión de moderar los comentarios sin tener conocimientos avanzados en lenguaje de programación.

Por otra parte presenta desventajas como:

- Bloguear consume tiempo y muchas veces este tiempo es tomado de otras actividades.
- En ocasiones pueden crear conflictos entre las personas debido a comentarios ofensivos o malinterpretados.

## **Red social**

De acuerdo a lo que plantea Jaime Royero se definen las redes sociales como "el conjunto de personas, comunidades, entes u organizaciones que producen, reciben e intercambian bienes o servicios sociales para su sostenimiento en un esquema de desarrollo y bienestar esperado. Dicho bienestar es mediatizado por los avances en el campo de la ciencia y la tecnología producidos y ofrecidos en su valor social y mercantil

a las personas o grupos de ellas, en un territorio y en unas condiciones económicas sociales determinadas. Estos intercambios se dan a nivel local, regional, nacional, internacional y global" (9).

Al estar basadas en la Web 2.0, la información está disponible en cualquier momento y en cualquier lugar. Basta con ingresar a la página web o aplicación desde una computadora o cualquier dispositivo con conexión a Internet para conocer el status que tienen los compañeros de trabajo.

Sus funciones principales pueden ir desde resolver problemas de información, elevando la productividad al eliminar tiempos muertos, así como almacenar toda la información compartida en la memoria disponible para búsquedas semánticas.

Entre los beneficios que se pueden obtener gracias a las redes sociales se encuentran:

- Agilizar la transmisión de información entre trabajadores y hacer fluido todo el proceso de comunicación, interna y externa.
- Lograr mayor vinculación con la empresa. Este tipo de comunicación ayuda a que los departamentos de las empresas estén relacionados y comunicados entre sí, incluso si se encuentran en diferentes ciudades o países.

También presenta desventajas como:

- Compañías especialistas en seguridad afirman que para los hackers es muy sencillo obtener información confidencial de sus usuarios.
- La existencia de perfiles falsos atenta contra la confiabilidad del conocimiento generado
- Falta de privacidad, siendo mostrada públicamente información personal.

### ***Microblogging***

El *microblogging* es una forma de comunicación o un sistema de publicación en Internet que consiste en el envío de mensajes cortos de texto (con una longitud máxima de 140 caracteres) a través de herramientas creadas específicamente para esta función. Su finalidad es la de explicar qué se está haciendo en un momento determinado, compartir información con otros usuarios u ofrecer enlaces hacia otras páginas web (10).

Como apunta Cristóbal Fernández en su tesis doctoral podemos definir qué es el *microblogging* a partir de la suma de cuatro conceptos:

1. Blog: se escribe como un diario personal o un espacio propio del autor, publicando contenidos en orden cronológico.
2. Mensajería instantánea: se trata de sistemas de comunicación que permiten mantener conversaciones en tiempo real en Internet entre usuarios.
3. SMS: se basa en el envío de mensajes de texto de hasta 140 caracteres (que pueden incluir enlaces).
4. Red social: el envío de mensajes se hace a un espacio público compartido por todas las personas que “sigan” un perfil produciéndose un intercambio de contenidos y conversaciones (10).

Luego de analizar la conceptualización anterior se definen las siguientes ventajas:

- El tamaño de las entradas permite que las actualizaciones sean rápidas y concisas.
- Combinan diferentes medios de comunicación lo que hace que sean más versátiles que otras tecnologías del mismo tipo.
- Permite la comunicación sincrónica y asincrónica dado que al recibir una actualización se puede responder instantáneamente o hacerlo en otro momento.

Aunque también presenta algunas dificultades como:

- Al tener solo 140 caracteres a veces es difícil contar todo lo que está sucediendo y puede generar inconvenientes al tener que abreviar todo.
- Existen gran cantidad de perfiles falsos lo que puede generar desconfianza entre los usuarios.
- La información no verídica se difunde rápidamente propiciando la difamación y el malestar de los usuarios que comparten información en este ámbito.

## Wiki

Es un sistema de creación, intercambio y revisión de contenido en la web, de forma colaborativa, fácil y automática. La finalidad de una wiki es permitir que varios usuarios puedan crear páginas web sobre un mismo tema, de esta forma cada usuario aporta un poco de su conocimiento para que el contenido de la página web sea más completo, creando una comunidad de usuarios que comparten contenidos enmarcados en un mismo tema o categoría específica (7).

Las wikis buscan la colaboración, cada día aparecen más Webs que incluyen una sección wiki donde el lector encontrará bien organizada y accesible gran cantidad de información. Estas herramientas pueden aglutinar desde las FAQ<sup>1</sup> o preguntas frecuentes a tutoriales y manuales.

El uso de las wikis como medio de comunicación interna ofrece las siguientes ventajas:

- Posibilita la participación de múltiples editores de forma colaborativa.
- Permite publicar y difundir el material creado por usuarios a través de una interfaz simple.
- Favorece la revisión del trabajo a medida que se va realizando, haciendo posible que se pueda comprobar el progreso que se lleva a cabo.

Además existen factores que desfavorecen su uso:

- Al ser un servicio abierto a todo el público, cualquiera puede subir, borrar y modificar información, por lo que la veracidad de los contenidos puede ser cuestionada.
- El mal control y supervisión del contenido atentan contra la confiabilidad de la información
- En ocasiones se introduce material que viola los derechos de autoría.

---

<sup>1</sup> FAQ (*Frequently Asked Questions*): Preguntas frecuentes.

## 1.2.1 Sitios web destinados a la comunicación

### Wikipedia

Wikipedia es una enciclopedia libre, políglota y editada colaborativamente. Es administrada por la Fundación Wikimedia, una Organización sin ánimo de lucro cuya financiación está basada en donaciones. Sus más de 37 millones de artículos en 287 idiomas (cantidad que incluye idiomas artificiales como el esperanto, lenguas indígenas o aborígenes como el náhuatl, el maya y las lenguas de las islas Andamán, o lenguas muertas, como el latín, el chino clásico o el anglosajón) han sido redactados conjuntamente por voluntarios de todo el mundo y prácticamente cualquier persona con acceso al proyecto puede editarlos. Iniciada en enero de 2001 por Jimmy Wales y Larry Sanger, es la mayor y más popular obra de consulta en Internet.

Desde su fundación, Wikipedia no solo ha ganado en popularidad (se encuentra entre los 10 sitios web más populares del mundo), sino que su éxito ha propiciado la aparición de proyectos hermanos. No obstante, existen numerosos detractores. Entre ellos, algunos la han acusado de parcialidad sistémica y de inconsistencias, con críticas centradas sobre lo que algunos, como Larry Sanger, han convenido en llamar “antielitismo”, y que no es otra cosa que la política del proyecto enciclopédico de favorecer el consenso sobre las credenciales en su proceso editorial. Otras críticas han estado centradas en su susceptibilidad de ser vandalizada y en la aparición de información falsa o falta de verificación, aunque estudios eruditos sugieren que el vandalismo en general es deshecho con prontitud (11).

### Twitter

Twitter es un servicio gratuito de *microblogging* que permite a sus usuarios enviar micro-entradas basadas en texto, denominadas “*tweets*”, de una longitud máxima de 140 caracteres. Lo interesante es que esta tecnología vincula la web con los celulares vía SMS<sup>2</sup> o a través de programas de mensajería instantánea como MSN<sup>3</sup>, Googletalk o incluso desde cualquier aplicación de terceros. Hoy Twitter es uno de los sistemas de comunicación más utilizados, no sólo para información intrascendente, social, sino como herramienta de comunicación entre profesionales. Twitter estalló al gran público en 2007, y pasó de 20,000 *tweets* al día a

---

<sup>2</sup> SMS (*Short Message Service*): Servicio de Mensajes Cortos.

<sup>3</sup> MSN (*Microsoft Network*): Red de Microsoft.

60,000. El servicio rápidamente comenzó a ganar seguidores y en marzo de 2007 ganó el premio *South by Southwest Award* en la categoría de blog (12).

### **Características**

- Twitter es una red social simple y didáctica ya que permite a sus usuarios estar al tanto de temas de interés.
- El hecho de que haya que escribir todo en 140 caracteres hace que la red social sea sencilla y práctica para el pensamiento humano.
- Si una persona publica un *tweet* y a alguno de sus seguidores les parece importante este lo puede retwittear así a este *tweet* lo ven también todos los seguidores de este seguidor.
- Es posible combinar la cuenta de Twitter con la de Facebook con lo cual lo que se publica en Twitter también se realiza en la otra red social. Esto es una gran ventaja para no perder tanto tiempo publicando en ambas redes sociales.
- Se pueden colocar botones en los sitios webs para twittear la página y así otros usuarios pueden observar y opinar el tema en cuestión (13).

### **Facebook**

Facebook nació en 2004 como un hobby de Mark Zuckerberg, en aquel momento estudiante de Harvard, y como un servicio para los estudiantes de su Universidad. En su primer mes de funcionamiento Facebook contaba con la suscripción de más de la mitad de los estudiantes de *Harvard*, y se expandió luego a las Universidades MIT<sup>4</sup>, *Boston University* y *Boston College* y las más prestigiosas instituciones de Estados Unidos. Un año después, Facebook tenía más de un millón de usuarios, una oficina en Palo Alto, California y había recibido un enorme apoyo financiero. Ese mismo año incorporó, a los alumnos de más de 25 mil escuelas secundarias y dos mil Universidades de Estados Unidos y el extranjero, logrando un total de 11 millones de usuarios (14).

---

<sup>4</sup> MIT (*Massachusetts Institute of Technology*): Instituto de Tecnología de Massachusetts.

## Características

- Permite estar conectado mediante el chat con personas del trabajo, familiares o amigos. Esta es una gran ventaja ya que se puede estar en contacto todo el tiempo no solo cuando la otra persona está conectada sino que también cuando está desconectada, porque el mensaje se envía de la misma forma y será leído cuando la otra persona se conecte.
- Permite conocer las últimas noticias de la actualidad debido a aplicaciones en Facebook posibilitando comentarlas para intercambiar ideas con otras personas.
- Permite adjuntar contenido multimedia como fotos o videos para poder estar más en contacto con personas de interés sin depender únicamente del chat.
- Es posible crear encuestas para tener una vista panorámica y democrática de lo que piensa la mayoría de la gente.
- Facebook da la posibilidad de difundir y dar a conocer ideologías o hechos de la actualidad que pueden resultar interesantes (15).

### 1.2.2 A nivel nacional

Especialistas del proyecto informático CubaVa trabajan en un servicio de *microblogging* para la Isla, mediante el cual los cubanos podrán intercambiar en la red .cu frases cortas, imágenes individuales o enlaces de video. En declaraciones a la ACN<sup>5</sup>, Kirenia Fagundo, líder de este proyecto, explicó que **Pitazo** será la denominación de ese sistema y resaltó que el *microblogging* es una exitosa forma de comunicación en el ciberespacio, como lo constituye Twitter. Según apuntó, un rasgo distintivo de esos medios de difusión, es su sencillez, porque en la mayoría de sus versiones el tamaño máximo de las entradas de texto es de 140 caracteres. Además, sobresale entre sus principales prestaciones la posibilidad de interactuar con otras personas, colocar anuncios y promociones, además de constituir espacios para la búsqueda de opciones laborales. CubaVa, integrado por 15 expertos del área de las tecnologías, habilitará también un sitio digital nombrado **El Estanquillo**, que reunirá una amplia variedad de prensa nacional y extranjera. Entre las misiones fundamentales del grupo, están desarrollar aplicaciones informáticas a tono con las tendencias de

---

<sup>5</sup> ACN: Agencia Cubana de Noticias

la Web 2.0, y que estas tengan un impacto valioso en la sociedad. En tal sentido, sobresalen creaciones como la red social cubana **La Tendedera** y la plataforma para blogs **Reflejos**, que ya cuentan desde el 7 de septiembre de 2014 con 4 275 usuarios registrados, así como 275 bitácoras digitales, respectivamente (16).

### **Cubadebate**

Es un sitio digital cubano afiliado a las posibilidades interactivas de la Web 2.0 con proyección y participación universal, que ofrece a sus usuarios servicios multimedia y un espacio interactivo para debatir temas y opiniones de interés de las más amplias audiencias, en particular la cubana. Su aspiración es alcanzar, mediante el trabajo de su colectivo y la interacción con nuestros lectores, el mayor rigor en los contenidos.

Cubadebate es además la voz del Círculo de Periodistas Cubanos contra el Terrorismo, organización creada para agrupar a los colegas residentes en la Isla, con el propósito de denunciar las acciones organizadas y financiadas por el gobierno de Estados Unidos contra nuestro país desde hace más de 50 años y que han ocasionado la muerte a más de 3000 inocentes. Esta publicación digital se edita en español (con versiones en otros siete idiomas) y sus únicos recursos son el tiempo y el trabajo no remunerado de sus colaboradores (17).

### **Ecured**

EcuRed es un proyecto de enciclopedia en red cubano desarrollado y administrado principalmente por personas afines a los Joven Club de Computación y Electrónica, entidad del Ministerio de Informática y Comunicaciones, que posee más de 600 filiales en todo el país encargadas de enseñar computación. Fue lanzada oficialmente el 14 de diciembre de 2010, con el auspicio del gobierno cubano y apoyado por la Oficina Nacional para la Informatización (ONI) y el Instituto de Información Científica y Tecnológica (IDICT).

El contenido está entrelazado con la Revolución Cubana y está escrito desde una perspectiva cubana. Su filosofía es la acumulación y el desarrollo del conocimiento con un objetivo democratizador y no lucrativo. La enciclopedia cubana EcuRed surge ante la necesidad de aportar conocimiento universal en un dominio .cu para de esa manera ser accesible a la inmensa mayoría de los cubanos, que por causa de las restricciones que el bloqueo del gobierno de Estados Unidos impone contra Cuba, no puede acceder a

Internet. La EcuRed además quiere llegar a ser una poderosa herramienta educativa con todas las ventajas que tiene una enciclopedia en línea.

Para la EcuRed la fiabilidad de sus contenidos está por encima de todo, por lo que los artículos son confeccionados utilizando fuentes confiables y de probada veracidad. El sitio utiliza el motor Mediawiki (en su versión 1.16.0), un motor para wikis bajo licencia GNU<sup>6</sup>, programado en PHP<sup>7</sup>, desarrollado inicialmente para Wikipedia y otros proyectos de la fundación Wikimedia pero que ha tenido una gran expansión desde el año 2005 en numerosos proyectos (18).

### **1.2.3 Resultados del estudio realizado**

A partir del estudio realizado de cada uno de los diferentes medios de comunicación a través de la Web 2.0 y sus ejemplos más fehacientes se decidió optar por una red social de *microblogging* a la cual se le incluirán otros de los medios de comunicación anteriormente mencionados en aras de aprovechar las potencialidades de cada uno de estos.

### **Ventajas y desventajas del uso de la red social de *microblogging* en la UCI**

#### **Ventajas**

- Ayuda al ejercicio creativo, a generar debates, opiniones y soluciones a problemas, motivando a los trabajadores, estudiantes y profesores a compartir conocimientos entre ellos.
- Permite publicar y difundir el material creado por usuarios a través de una interfaz simple.
- Combina diferentes medios de comunicación lo que hace que sean más versátiles que otras tecnologías del mismo tipo.
- El tamaño de las entradas permiten que las actualizaciones sean rápidas y concisas.

---

<sup>6</sup> GNU: Acrónimo recursivo de "*GNU's Not Unix!*", GNU no es Unix

<sup>7</sup> PHP: Acrónimo recursivo de *PHP Hypertext Preprocessor*, Procesador de Hipertexto PHP

- Permite a los usuarios mantenerse actualizados sobre las personas que les interesen y temas específicos o de actualidad.
- Permite la comunicación sincrónica y asincrónica dado que al recibir una actualización se puede responder instantáneamente o hacerlo en otro momento.
- Se puede difundir información ampliamente debido a que en estas redes el contenido se propaga exponencialmente rápido.

### **Desventajas**

- Al tener solo 140 caracteres a veces es difícil contar todo lo que está sucediendo y puede generar inconvenientes al tener que abreviar todo.
- Existen gran cantidad de perfiles falsos lo que puede generar desconfianza entre los usuarios.
- La información no verídica se difunde rápidamente propiciando la difamación y el malestar de los usuarios que comparten información en este ámbito.

Para contrarrestar el efecto de estas desventajas, se llevarán a cabo las siguientes acciones:

- Extender el tamaño de las entradas para que el usuario pueda dar una mejor explicación de lo que desea informar.
- El acceso al sistema se realizará mediante el servicio de autenticación LDAP de forma que todos los usuarios tengan una única cuenta y no permita la creación de perfiles falsos.

### **1.3 Herramientas de Comunicación interna en la Universidad**

Actualmente en la Universidad existen diferentes medios a través de los cuales se establece comunicación. Además de contar con correo electrónico y mensajería instantánea hay que destacar que nuestra institución cuenta con un total de 104 sitios web en línea divididos por diferentes temáticas. Estas van desde sitios web que responden a los intereses de cada facultad hasta blogs o comunidades de temas y categorías específicas. Entre estos podemos mencionar el blog Humanos, Intranet y Periódico Mella.

Con el motivo de ver el impacto de estos medios en la comunicación interna de la universidad se realizó una encuesta, para la cual se tomó una muestra de 50 personas de ellas 35 estudiantes y 15 trabajadores (ver anexo 1):

### **Resultados de la encuesta**

El 94% de los encuestados coincide en que es difícil hacer pública o compartir información de su interés con un gran grupo de personas a través de los medios de comunicación actuales. Medios como el correo o el chat permiten la interacción entre usuarios pero la información que estos comparten no queda accesible a otros usuarios que pueden estar interesados. Por otra parte las publicaciones en los sitios web de la Universidad son temáticas por lo que si un usuario necesitara publicar un tema no relacionado con alguno de los creados, debe seguir un conjunto de pasos que hacen engorroso el proceso de hacer llegar la información a lo demás.

El 97.4% de los encuestados afirma que no reciben información en tiempo real sobre las actividades o temas que le interesan. El correo es el medio más utilizado en la Universidad para informar a todos sus miembros, pero al ser una herramienta de comunicación asincrónica el receptor solo la recibe al acceder a su cuenta de correo e incluso es posible que no reciba esta información por las diferentes limitaciones que este medio presenta (ya sea que el usuario tenga el buzón lleno o que por alguna razón no haya sido incluido en la lista de distribución). Los sitios de la intranet publican información de todo el ámbito universitario pero solo será de conocimiento del usuario si este conoce que esta publicada y accede a ella.

Un 86% de los encuestados plantea que no existe un espacio donde debatir temas de interés en cualquier momento. Aunque existen en la Universidad espacios de debate están restringidos solo a reuniones de grupo o de la UJC y en estas últimas solo participan los miembros de dicha organización. Esto hace difícil el debate de un tema puntual, que haya surgido espontáneamente, con otras personas de la institución con el objetivo de obtener opiniones o soluciones.

Por último el 100% de los encuestados escogió al correo electrónico como vía para comunicar sus inquietudes a los directivos o superiores. A través de esta vía de comunicación la retroalimentación es un proceso lento por lo que a menudo a las personas se les hace necesario enviar más de un correo o personarse en la oficina o local de dicho directivo.

## 1.4 Software utilizado

### 1.4.1 Plataformas para desarrollar el sistema propuesto

#### Mahara

Se inicia con esta opción de código abierto que está bajo la Licencia Pública General de GNU/Linux. El nombre significa “pensar” o “pensamiento” en idioma Maorí. Es una herramienta para construir redes sociales de aprendizaje en línea donde los usuarios crean o colaboran con portafolios digitales: metáfora usada para representar una amplia gama de recursos de aprendizaje y colaboración como blogs, multimedia, imágenes y documentos. Además, ha sido creada para el aprendizaje y enseñanza promoviendo que las comunidades estén en constante interacción (19).

La misma incluye:

1 - Repositorio de ficheros que permite al usuario:

- Crear estructuras de carpetas y subcarpetas.
- Subir múltiples archivos rápida y eficientemente.
- Dar a cada fichero un nombre y descripción.
- Gestionar la cuota de subida de ficheros.
- Cuando se cargue un fichero los usuarios deben aceptar una cláusula configurable de descargo de responsabilidad.

2 - Blogs:

Proporciona una herramienta detallada para la creación de blogs, en la que los blogs y los envíos al blog se consideran “Artefactos” y pueden añadirse a la “Vista”. La herramienta de creación de blogs permite al usuario:

Crear un blog usando un editor WYSIWYG (*What You See Is What You Get*, lo que ves es lo que obtienes).

- Adjuntar ficheros a la entrada (*post*).
- Grabar imágenes en las entradas (*postings*).

- Configurar si en el blog pueden recibirse comentarios o no.
- Crear borradores de entradas para su posterior publicación.

### 3 - Perfil:

Dentro los usuarios pueden compartir detalles mediante una variedad de campos de información opcionales sobre su perfil que incluyen:

- Nombre preferido.
- Id como estudiante.
- Dirección postal y números de teléfono.
- Presentación.
- Imágenes para el perfil.

### 4 - Administración:

Los administradores pueden personalizar la interfaz a través de un cierto número de elementos de configuración que incluyen:

- Paquetes de idioma y temas.
- Protocolo en caso de presencia de virus.
- Vida media de sesiones y cuentas.
- Métodos de autenticación.
- Organización de la Institución.
- Página central del editor.
- Menú principal del editor.

Además de la estructura de *plugin* modular de la misma, los tipos de “Artefactos” se pueden configurar, deshabilitar o habilitar de acuerdo con los requerimientos de la organización.

#### 5 - Seguridad:

- Detecta automáticamente entornos que representen una amenaza a la seguridad.
- Se han integrado códigos de manejo de claves de sesión con el núcleo de la interfaz de programación de aplicaciones (API) formulario/solicitud.
- Proporciona abstracción de bases de datos que impide cualquier ataque a la base de datos y validación de inputs que impiden ataques mediante inyección de scripts.
- La autenticación del usuario puede unirse a sistemas externos tales como sistemas de gestión de estudiantes u otras bases de datos tales como su ID de estudiante.

Entorno Tecnológico:

Servidor: Linux, Windows

Servidor web: Apache 2.2

Lenguaje: PHP 5.2.9 o superior.

Base de datos: PostgreSQL 8.1 (recomendado), o MySQL 5.0.91.

### **Elgg**

Es una aplicación para crear redes sociales con un sólido código abierto que contempla la escalabilidad de proyectos. Una plataforma social que brinda elección, flexibilidad y apertura, que puede adecuarse a toda clase de ambientes sociales: academia, negocios, deportes y más. Los usuarios tienen libertad de manejar todas sus herramientas preferidas y compartir con los familiares o amigos a través de una red social. Ha sido muy utilizado para colaborar en entornos educativos y científicos (20).

Características:

- Cada usuario tiene un espacio personal con su blog, una carpeta personal donde puede subir archivos, una red de amigos, sistemas de sindicación simple (RSS), perfil personal.

- Tiene comunidades, que reproducen el mismo esquema que los espacios personales pero con varios usuarios.
- Hay grupos de trabajo, que engloban a varios usuarios y permite gestionar el acceso a nivel de usuario o comunidad.

#### Elementos disponibles:

- Blog: Permite que los usuarios gestionen sus propios blogs.
- Foros: Disponibilidad de crear foros.
- Sistema de Mensajería Instantánea entre usuarios.
- Flujo de Eventos y noticias entre los blogs.
- Ampliable mediante *widgets*: Existen múltiples elementos que permiten la aplicación del mismo.
- Administración: Dispone de un panel de control de administración basado en tecnologías web para administrar sitios de manera fácil, con una interfaz limpia.

#### Funcionamiento:

- Permite crear espacios personales. Cada uno de estos espacios personales cuenta con un blog, una carpeta personal, una red de contactos, un agregador que alimenten los RSS y un perfil personal.
- Permite crear comunidades. Las comunidades son elementos que reproducen el mismo esquema que los espacios personales pero con varios miembros.
- Permite crear grupos de trabajo. Permiten gestionar el acceso los recursos de la carpeta de un usuario o una comunidad.

#### Requerimientos Técnicos:

Servidor: Linux, Windows.

Servidor web: Apache 1.3 o superior, recomendado Apache 2.0

Lenguaje: PHP 5.5 o superior.

Base de datos: MySQL 5.0 o superior.

## StatusNet

StatusNet es un servidor de *microblogging* de *software* libre escrito en PHP que implementa el estándar OStatus para interoperar entre distintas redes sociales. A la par que ofrece una funcionalidad similar a Twitter, StatusNet busca proporcionar el potencial para una comunicación abierta, inter-servicios y distribuida entre comunidades de *microblogging*. Empresas e individuos pueden instalar y controlar sus propios servicios y datos. Su última versión estable (1.1.1) fue liberada el 16 de julio de 2013 en la cual se le dejó de dar soporte.

Luego de haber analizado y trabajado en el código fuente de la plataforma, con el objetivo de adaptarla a las necesidades del cliente, se realizaron tareas que dieron lugar a la siguiente conclusión.

La plataforma no se encuentra soportada por ningún *framework* por lo cual no ofrece una estructura clara y organizada a varios niveles, como la estructura de directorios y la separación por capas. Esto conllevaba que resultara más complicado encontrar cualquier recurso cuando fuera necesario modificarlo.

## Características

- Actualizaciones vía cliente XMPP<sup>8</sup>.
- Autenticación OpenID<sup>9</sup>.
- Actualizaciones y notificaciones SMS.
- Una API compatible con Twitter.
- Hashtags<sup>10</sup>.

---

<sup>8</sup> XMPP (*Extensible Messaging and Presence Protocol*): Protocolo de Presencia y Mensajería Extensible.

<sup>9</sup> OpenID: Estándar de identificación digital descentralizado, con el que un usuario puede identificarse en una página web a través de una URL y puede ser verificado por cualquier servidor que soporte el protocolo.

<sup>10</sup> Hashtag: Es una etiqueta de metadatos precedida de un carácter especial con el fin de que tanto el sistema como el usuario la identifiquen de forma rápida.

- Interfaz multilingüe.

Requerimientos Técnicos:

Servidor: Linux, Windows.

Servidor web: Apache 2.2.x

Lenguaje: PHP 5.2.3.

### **Exo platform**

Exo platform comenzó como un proyecto de código abierto en el año 2002, como el primer contenedor de *portlets* Java de la industria. Hoy en día, Exo platform es una compañía global de rápido crecimiento, con sede en EE.UU. en San Francisco, California, sede central en Francia, y oficinas en Túnez, Ucrania y Vietnam. La compañía ha establecido su liderazgo tecnológico y valor demostrado por su gran base Europea y la asociación estratégica con Red Hat. Sus socios principales son 4Linux, Groupe Onepoint, IntelliJIdea, Vizuri y JBoss entre otros (21).

Exo platform está basada en el marco de portal GateIn en código abierto y tiene toda la flexibilidad de un *framework* de portal empresarial: *portlets*, *gadgets* y el contenido puede ser agregado de manera segura y personalizable.

Los *portlets* son componentes web que se implementan en el interior de un contenedor y generan contenido dinámico. En el aspecto técnico, un *portlet* es una clase que implementa la interfaz *javax.portlet.Portlet* y se empaqueta y se despliega como un archivo *.war* dentro de un contenedor de *portlets*.

### **Características**

- **Rol basado en la entrega de contenido y apps:** Las políticas basadas en roles agregan y muestran algunas páginas, contenido, *gadgets* y *portlets* dependiendo de los roles de los usuarios conectados.
- **Personalizable:** Gestión fácil de diseño y página usando arrastrar y soltar y editores WYSIWYG. Altamente flexible para implementar un sitio web con cualquier diseño de marca.

- **Acceso al flujo de noticias:** Actualizar estado, leer mensajes de otros usuarios y las notificaciones de actualizaciones, nuevos mensajes, nuevos eventos de calendario o de nuevos documentos en su repositorio.
- **Autenticación y seguridad:** Compatible con soluciones mayores de SSO<sup>11</sup> y JAAS<sup>12</sup>, la autenticación y mecanismo de autorización para JEE (Java Enterprise Edition).
- **Apoyo multilingüe:** Disponible en inglés, francés, italiano, portugués y japonés. Fácil a traducir a otros idiomas. Apoyo de localización para los idiomas de Derecha a Izquierda.
- **Perfiles:** Permite edición de perfil con información acerca de sus antecedentes y habilidades. Permite encontrar personas con conocimiento relacionado o relevante y experiencia y conectar con otros en su organización a través del directorio de personas.
- **Calendario:** Calendario completo e integrado permite a los usuarios gestionar múltiples agendas personales y grupales y compartirlas con otras personas.
- **Basado en Estándares:** Contenedor certificado de JSR-286<sup>13</sup>, apoyo WSRP y contenedor OpenSocial. Esto facilita la integración y reutilización de aplicaciones de terceros desarrollados con otros marcos de trabajo (incluyendo JSF, Spring, GWT).

Requerimientos técnicos:

Servidor: Linux, Windows, Mac.

Servidor web: Apache Tomcat 7.0.55.

Base de datos: MySQL, PostgreSQL, MSSQL, Oracle, Sybase, DB2

---

<sup>11</sup> SSO (Single Sign On): Autenticación Única. Permite a un usuario acceder a varios sistemas con una sola instancia de identificación.

<sup>12</sup> JAAS (Java Authentication and Autorization Service): Servicio de Autenticación y Autorización de Java.

<sup>13</sup> JSR (Java Specification Request): Solicitud de especificación Java. Define un contrato entre el contenedor del portlet y el portlet, además provee un modelo de programación conveniente para los desarrolladores de portlets en Java.

Máquina Virtual de Java 7.0, con la variable de entorno JAVA\_HOME definida.

#### 1.4.2 Resultado del análisis de las plataformas

Se decide utilizar la plataforma Exo platform atendiendo a los siguientes factores:

1- Al investigar el nivel de actualización de cada una de las plataformas analizadas se decide eliminar la plataforma Statusnet aunque a pesar de responder a las necesidades del cliente, darle soporte requeriría mucho tiempo y recursos. Por otra parte las plataformas Elgg, Mahara y Exo platform presentan un nivel adecuado de actualización siendo las últimas actualizaciones de estas 1.11.1 abril de 2015, 1.10 octubre 2014 y 4.2 febrero 2015 respectivamente.

2- Al analizar la conexión a base de datos de cada una de las plataformas se llega a la conclusión que Elgg no es conveniente debido a que solo posee conexión con el gestor de base de datos MySQL el cual posee cláusulas privativas dentro de su licencia. A diferencia de esta Mahara y StatusNet permiten conexión con los gestores de base de datos MySQL y PostgreSQL junto a Exo platform la cual además incorpora otras como MSSQL, Oracle, Sybase, DB2.

3- Desde el punto de vista de escalabilidad y soporte Exo platform es la plataforma que mejor se desempeña debido a que permite realizar modificaciones mediante el empleo de *portlets*, ya sean de contenido o de aplicación. Esto facilita que al ser publicada una nueva versión el sistema pueda ser actualizado sin necesidad de cambiar los *portlets* creados previamente, los cuales permiten adaptar la plataforma a nuestras necesidades, en caso de que sea necesario algún cambio este no requerirá de muchos esfuerzos; permitiendo dedicar un mínimo de recursos humanos y de tiempo. Las plataformas Mahara y Elgg presentan un cierto nivel de escalabilidad dado que permiten agregar módulos para una misma versión de la aplicación sin afectar otras funcionalidades mediante la creación de *plugins* pero es posible que al ser actualizada la plataforma estos no sean compatibles con una nueva versión. En este aspecto la plataforma StatusNet no presenta mecanismos de escalabilidad siendo necesario reimplementar las funcionalidades al actualizar la plataforma.

## 1.5 Lenguajes y herramientas de desarrollo

### 1.5.1 Lenguajes de programación

#### Groovy

Groovy es un lenguaje opcionalmente tipado y dinámico, con capacidades de compilación y tipado estáticos para la plataforma Java. Se integra sin problemas con cualquier programa de Java, e inmediatamente entrega a la aplicación potentes características, incluyendo capacidades de scripting, de autoría de lenguaje de dominio específico, programación funcional y meta-programación para tiempo de ejecución y tiempo de compilación.

Posee una sintaxis concisa, fácil de leer y expresiva además de sencilla de aprender para los desarrolladores de Java. Transparentemente y sin problemas se integra e interopera con Java y todas las bibliotecas de terceros. Permite integración avanzada y mecanismos de personalización para añadir reglas de negocio legibles en las aplicaciones (22).

#### Javascript

JavaScript es un lenguaje de programación que se utiliza principalmente para crear páginas web dinámicas. Es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos. En otras palabras, los programas escritos con JavaScript se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios. A pesar de su nombre, JavaScript no guarda ninguna relación directa con el lenguaje de programación Java. Legalmente, JavaScript es una marca registrada de la empresa Sun Microsystems (23).

### 1.5.2 Lenguajes de marcado

#### HTML<sup>14</sup>

El lenguaje HTML es un estándar reconocido en todo el mundo y cuyas normas define un organismo sin ánimo de lucro llamado W3C (*World Wide Web Consortium*). Como se trata de un estándar reconocido por todas las empresas relacionadas con el mundo de Internet, una misma página HTML se visualiza de la misma manera en cualquier navegador de cualquier sistema operativo. El propio W3C define el lenguaje

---

<sup>14</sup> HTML (*Hypertext Markup Language*): Lenguaje de Marcado de Hipertexto.

HTML como “un lenguaje reconocido universalmente y que permite publicar información de forma global” (24).

## **XML**

El lenguaje de marcado extensible (XML por sus siglas en inglés *Extensible Markup Language*) es un sencillo formato basado en texto para representar información estructurada: documentos, datos, configuración, libros, transacciones, facturas y mucho más. Fue derivado de un formato estándar antiguo llamado SGML<sup>15</sup>(ISO 8879), con el fin de ser más adecuado para su uso Web (25).

### **1.5.3 Lenguaje de modelado**

#### **UML (Lenguaje de modelado unificado)**

Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema. UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocio, funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y compuestos reciclados (26).

### **1.5.4 Otras Tecnologías**

#### **CSS<sup>16</sup>**

CSS es un lenguaje de hojas de estilos creado para controlar la presentación de los documentos electrónicos definidos con HTML y XHTML. CSS es la mejor forma de separar los contenidos y su presentación y es imprescindible para la creación de páginas web complejas. La separación de los contenidos y su presentación presenta numerosas ventajas, ya que obliga a crear documentos HTML/XHTML bien definidos y con significado completo (27).

#### **Jquery 2.1.3**

JQuery es un marco de trabajo rápido, poderoso y fácil de utilizar que permite a los desarrolladores y diseñadores *web* agregar elementos dinámicos e interactivos a sus sitios. Simplifica la manera de interactuar

---

<sup>15</sup> SGML (Standard Generalized Markup Language): Lenguaje de Marcado Generalizado Estándar.

<sup>16</sup> CSS (*Cascading Style Sheet*): Hoja de Estilo en Cascada.

con los documentos HTML, así como la manipulación del árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la técnica AJAX. Además cuenta con un gran número de plugins que permiten extender sus funcionalidades (28).

### **Bootstrap 2.3.2**

Bootstrap es un *framework* o conjunto de herramientas de *software* libre para diseño de sitios y aplicaciones web. Contiene plantillas de diseño con tipografía, formularios, botones, cuadros, menús de navegación y otros elementos de diseño basado en HTML y CSS, así como, extensiones de JavaScript opcionales adicionales.

### **Ajax**

El término AJAX es un acrónimo de *Asynchronous JavaScript + XML*, que se puede traducir como "JavaScript asíncrono + XML". Las tecnologías que forman AJAX son: XHTML y CSS, para crear una presentación basada en estándares. DOM, para la interacción y manipulación dinámica de la presentación. XML, XSLT y JSON, para el intercambio y la manipulación de información. *XMLHttpRequest*, para el intercambio asíncrono de información. JavaScript, para unir todas las demás tecnologías (29).

### **1.5.5 Herramientas**

#### **NetBeans IDE 8.0.2**

NetBeans IDE es un ambiente integrado de desarrollo de código abierto para desarrolladores de *software* que intentan construir aplicaciones usando mayormente Java, o algún otro lenguaje de programación popular como C++, PHP, Python, Groovy, Ruby, y otros. Tomando ventaja de casi 10 años de desarrollo constante, la plataforma NetBeans logró ser creada con la contribución de la comunidad de código abierto, siendo un paquete IDE<sup>17</sup> bien diseñado que puede ser usado para la creación de cualquier tipo de aplicaciones de escritorio, web y móvil. También hace foco en permitir el desarrollo de aplicaciones usando conjuntos de componentes modulares de *software* que pueden ser creados tanto como por los propietarios de NetBeans, la Corporación Oracle, como por terceros desarrolladores que han logrado expandir sus

---

<sup>17</sup>IDE (*Integrated Development Environment*): Ambiente de desarrollo integrado.

funcionalidades con numerosos *plugins*. Tres módulos principales de NetBeans IDE son *NetBeans Profiler* (que puede monitorear aplicaciones, reportar problemas, dificultades de performance y más), el editor de JavaScript NetBeans y la herramientas de diseño GUI<sup>18</sup> (30).

### **Apache Tomcat 7.0.55**

El proyecto Apache HTTP<sup>19</sup> Server es un esfuerzo de desarrollo de *software* de colaboración cuyo objetivo es crear una implementación del código fuente de un servidor HTTP (Web) de grado comercial, robusto y libre. El proyecto está gestionado conjuntamente por un grupo de voluntarios ubicados en todo el mundo, usando Internet y la Web para comunicarse, planear y desarrollar el servidor y su documentación relacionada. Este proyecto forma parte de la *Apache Software Foundation*. Además, cientos de usuarios han aportado ideas, código y documentación al proyecto (31).

Luego de un análisis de los gestores de base de datos postgresql y mysql (ver anexo 2) se escogió postgresql ya que posee funcionalidades necesarias para el desarrollo de la aplicación.

### **PostgreSQL**

PostgreSQL es un sistema de gestión de bases de datos objeto-relacional, distribuido bajo licencia BSD (*Berkeley Software Distribution*) y con su código fuente disponible libremente. Es el sistema de gestión de bases de datos de código abierto más potente del mercado. PostgreSQL utiliza un modelo cliente/servidor y usa multiprocesos en vez de multihilos para garantizar la estabilidad del sistema (32).

### **PgAdmin III**

PgAdmin III es un sistema integral de diseño y gestión de bases de datos PostgreSQL para sistemas Unix y Windows. Es libre bajo los términos de licencia de PostgreSQL y puede ser redistribuido siempre los términos de la licencia se cumplen. Está escrito en C ++ y utiliza las *wxWidgets* (anteriormente conocida como *wxWindows*) multiplataforma. La conexión a PostgreSQL se hace a través de la biblioteca libpq nativo (33).

---

<sup>18</sup> GUI (*Graphic User Interface*): Interfaz Gráfica de Usuario.

<sup>19</sup> HTTP (*Hypertext Transfer Protocol*): Protocolo de Transferencia de Hipertexto.

## Visual Paradigm Suite 5.0

Visual Paradigm es una herramienta CASE (Ingeniería de *Software* Asistida por Computación). La misma propicia un conjunto de ayudas para el desarrollo de programas informáticos, desde la planificación, pasando por el análisis y el diseño, hasta la generación del código fuente de los programas y la documentación (34).

## Jmeter 2.10

La aplicación jmeter es un *software* de código abierto, desarrollado en Java y diseñado para cargar comportamiento funcional de prueba y medir el rendimiento. Originalmente fue diseñado para probar las aplicaciones web, pero desde entonces se ha expandido a otras funciones de prueba (35).

## 1.6 Metodología

Las metodologías de desarrollo pueden ser clasificadas en dos grandes grupos: ágiles y pesadas. Las metodologías ágiles están orientadas a proyectos de pequeña envergadura y para los cuales se cuenta con un equipo de desarrollo pequeño. Principalmente se trata de crear *software* con la menor cantidad de documentación posible y con una gran interacción con el cliente permitiendo que este pueda realizar cambio a conveniencia. Entre las más destacadas cabe mencionar: XP (*Extreme Programming*), SCRUM y OpenUp. Por otra parte las metodologías pesadas presentan una mayor estabilidad y previsibilidad, se centran principalmente en la documentación y en el cumplimiento de los objetivos previamente establecidos por contrato. Generalmente son usadas para proyectos de gran envergadura y que requieren gran cantidad de documentación. De estas se pueden tomar como ejemplos *Microsoft Solution Frameworks* (MSF), Métrica 3 y Proceso Unificado de Desarrollo (RUP, por sus siglas en inglés) (36).

## OpenUP

El Proceso Unificado Abierto (OpenUP) es un proceso unificado y ágil que involucra un conjunto mínimo de prácticas que ayudan a los equipos de trabajo a ser más efectivos en el desarrollo de sistemas de *software* u otros sistemas de ingeniería (37).

Adopta un enfoque pragmático y su filosofía ágil se centra en la naturaleza colaborativa de desarrollo de *software*. Es una metodología agnóstica, ejecuta procesos de baja formalidad que pueden ser usados tal

cual o ampliarse para hacer frente a una amplia variedad de proyectos. Es un proceso de desarrollo de *software* mínimamente suficiente dado que incluye solo el contenido fundamental no teniendo que lidiar con temas tales como el tamaño del equipo, el cumplimiento, seguridad, orientación tecnológica, entre otros. Sin embargo OpenUP es completa en el sentido de que manifiesta por completo el proceso de construir un sistema. Para atender las necesidades que no están cubiertas en su contenido OpenUp es extensible a ser utilizado como base sobre la cual se pueden añadir o adaptarse a contenido de otro proceso que sea necesario. Se pueden resaltar dentro de sus características más importantes las siguientes (37):

- Desarrollo incremental.
- Uso de casos de uso y escenarios.
- Manejo de riesgos.
- Diseño basado en la arquitectura.

OpenUP es un proceso completo, flexible y corto, fomenta el uso de técnicas ágiles y principios, mientras que tiene un ciclo de vida estructurado y probado que hace referencia en la continua entrega de *software* de calidad (37).

## **SCRUM**

Scrum es un *framework* iterativo e incremental para proyectos, productos o desarrollo de aplicaciones. Scrum estructura el desarrollo en ciclos de trabajo llamados *sprints*. Estas iteraciones son no más de un mes cada una, y se realizan una tras otra sin pausa. Los *sprints* son enmarcados por tiempo, terminan en una fecha específica si el trabajo se ha completado o no, y nunca se extienden. Al comienzo de cada *sprint*, un equipo multifuncional selecciona elementos (requisitos de los clientes) de una lista de prioridades y el equipo se compromete a completar dichos elementos para el fin del *sprint*. Cada día el equipo se reúne brevemente para inspeccionar su progreso y ajustar los próximos pasos necesarios para completar el trabajo restante. Al final del *sprint*, el equipo revisa el resultado con las partes interesadas, y demuestra lo que se ha construido. Las personas obtienen retroalimentación que puede incorporarse en el próximo *sprint*. Scrum se centra en el producto de trabajo realmente hecho al final del *sprint* en el caso de *software*, esto significa código que está integrado, totalmente probado y potencialmente entregable (38).

Se utilizó la metodología XP por ser la más adecuada al proyecto teniendo en cuenta que es una metodología ágil utilizada para proyectos pequeños donde los requisitos pueden variar. Además se tuvo en cuenta el tamaño del equipo de desarrollo y su conocimiento de la metodología.

### **XP (*Extreme Programming*)**

XP es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de *software*, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. XP se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico (39).

A continuación presentaremos las características esenciales de XP organizadas en cuatro apartados: historias de usuario, roles, proceso y prácticas.

#### **Historias de usuario**

Son la técnica utilizada para especificar los requisitos del *software*. Se trata de tarjetas de papel en las cuales el cliente describe brevemente las características que el sistema debe poseer, sean requisitos funcionales o no funcionales. El tratamiento de las historias de usuario es muy dinámico y flexible. Cada historia de usuario es lo suficientemente comprensible y delimitada para que los programadores puedan implementarla en unas semanas

#### **Roles XP**

Los roles de acuerdo con la propuesta original de Kent Beck creador de la metodología son:

- Programador. El programador escribe las pruebas unitarias y produce el código del sistema.
- Cliente. Escribe las historias de usuario y las pruebas funcionales para validar su implementación. Además, asigna la prioridad a las historias de usuario y decide cuáles se implementan en cada iteración centrándose en aportar mayor valor al negocio.

- Encargado de pruebas (*Tester*). Ayuda al cliente a escribir las pruebas funcionales. Ejecuta las pruebas regularmente, difunde los resultados en el equipo y es responsable de las herramientas de soporte para pruebas.
- Encargado de seguimiento (*Tracker*). Proporciona realimentación al equipo. Verifica el grado de acierto entre las estimaciones realizadas y el tiempo real dedicado, para mejorar futuras estimaciones. Realiza el seguimiento del progreso de cada iteración.
- Entrenador (Coach). Es responsable del proceso global. Debe proveer guías al equipo de forma que se apliquen las prácticas XP y se siga el proceso correctamente.
- Consultor. Es un miembro externo del equipo con un conocimiento específico en algún tema necesario para el proyecto, en el que puedan surgir problemas.
- Gestor (*Big boss*). Es el vínculo entre clientes y programadores, ayuda a que el equipo trabaje efectivamente creando las condiciones adecuadas. Su labor esencial es de coordinación.

## Proceso XP

El ciclo de desarrollo consiste (a grandes rasgos) en los siguientes pasos:

1. El cliente define el valor de negocio a implementar.
2. El programador estima el esfuerzo necesario para su implementación.
3. El cliente selecciona qué construir, de acuerdo con sus prioridades y las restricciones de tiempo.
4. El programador construye ese valor de negocio.
5. Vuelve al paso 1.

En todas las iteraciones de este ciclo tanto el cliente como el programador aprenden. No se debe presionar al programador a realizar más trabajo que el estimado, ya que se perderá calidad en el *software* o no se cumplirán los plazos. De la misma forma el cliente tiene la obligación de manejar el ámbito de entrega del producto, para asegurarse que el sistema tenga el mayor valor de negocio posible con cada iteración. El ciclo de vida ideal de XP consiste de seis fases: Exploración, Planificación de la Entrega (*Release*), Iteraciones, Producción, Mantenimiento y Muerte del Proyecto.

## Prácticas XP

La principal suposición que se realiza en XP es la posibilidad de disminuir la mítica curva exponencial del costo del cambio a lo largo del proyecto, lo suficiente para que el diseño evolutivo funcione. Esto se consigue gracias a las tecnologías disponibles para ayudar en el desarrollo de *software* y a la aplicación disciplinada de las siguientes prácticas.

**Entregas pequeñas.** Producir rápidamente versiones del sistema que sean operativas, aunque no cuenten con toda la funcionalidad del sistema. Esta versión ya constituye un resultado de valor para el negocio. Una entrega no debería tardar más 3 meses.

**Metáfora.** El sistema es definido mediante una metáfora o un conjunto de metáforas compartidas por el cliente y el equipo de desarrollo. Una metáfora es una historia compartida que describe cómo debería funcionar el sistema (conjunto de nombres que actúen como vocabulario para hablar sobre el dominio del problema, ayudando a la nomenclatura de clases y métodos del sistema).

**Diseño simple.** Se debe diseñar la solución más simple que pueda funcionar y ser implementada en un momento determinado del proyecto.

**Pruebas.** La producción de código está dirigida por las pruebas unitarias. Éstas son establecidas por el cliente antes de escribirse el código y son ejecutadas constantemente ante cada modificación del sistema.

**Refactorización** (*Refactoring*). Es una actividad constante de reestructuración del código con el objetivo de remover duplicación de código, mejorar su legibilidad, simplificarlo y hacerlo más flexible para facilitar los posteriores cambios. Se mejora la estructura interna del código sin alterar su comportamiento externo.

**Programación en parejas.** Toda la producción de código debe realizarse con trabajo en parejas de programadores. Esto conlleva ventajas implícitas (menor tasa de errores, mejor diseño, mayor satisfacción de los programadores,).

**Propiedad colectiva del código.** Cualquier programador puede cambiar cualquier parte del código en cualquier momento.

**Integración continua.** Cada pieza de código es integrada en el sistema una vez que esté lista. Así, el sistema puede llegar a ser integrado y construido varias veces en un mismo día.

**40 horas por semana.** Se debe trabajar un máximo de 40 horas por semana. No se trabajan horas extras en dos semanas seguidas. Si esto ocurre, probablemente está ocurriendo un problema que debe corregirse. El trabajo extra desmotiva al equipo.

**Cliente in-situ.** El cliente tiene que estar presente y disponible todo el tiempo para el equipo. Éste es uno de los principales factores de éxito del proyecto XP. El cliente conduce constantemente el trabajo hacia lo que aportará mayor valor de negocio y los programadores pueden resolver de manera inmediata cualquier duda asociada. La comunicación oral es más efectiva que la escrita.

**Estándares de programación.** XP enfatiza que la comunicación de los programadores es a través del código, con lo cual es indispensable que se sigan ciertos estándares de programación para mantener el código legible.

El mayor beneficio de las prácticas se consigue con su aplicación conjunta y equilibrada puesto que se apoyan unas en otras. La mayoría de las prácticas propuestas por XP no son novedosas sino que en alguna forma ya habían sido propuestas en ingeniería del *software* e incluso demostrado su valor en la práctica. El mérito de XP es integrarlas de una forma efectiva y complementarlas con otras ideas desde la perspectiva del negocio, los valores humanos y el trabajo en equipo.

## 1.7 Conclusiones parciales

Para la selección de la plataforma que facilite el desarrollo de la propuesta de solución se tuvo en cuenta de las herramientas analizadas previamente sus características, ventajas y desventajas. Aunque en un menor grado de incidencia sobre la toma de decisión también se le prestó atención a quienes eran sus colaboradores y sus clientes; como resultado final del análisis de selección los autores definen a Exo platform para el desarrollo de la herramienta que da solución al problema planteado anteriormente.

Teniendo en cuenta la metodología que más se adecua al tamaño del equipo de desarrollo y a la envergadura del proyecto los autores seleccionan XP (*Extreme Programming*) como la metodología de Ingeniería de *Software* a utilizar para obtener un producto final que cumpla las funcionalidades requeridas.

## Capítulo II: Descripción de la propuesta de solución

En el presente capítulo se realiza la descripción de la propuesta de solución a desarrollar. De la misma forma se describen los principales elementos del análisis de esta, con el que se brinda un mayor entendimiento de las características y conceptos relacionados con la propuesta para darle cumplimiento a los objetivos formulados. Entre los aspectos que se detallan se encuentran: el listado de los requisitos funcionales y no funcionales de la plataforma y los patrones de arquitectura y diseño que se utilizan para el proceso de desarrollo.

### 2.1 Red social de microblogging

El desarrollo de este proyecto tiene como objetivo principal proveer a la Universidad de una herramienta que sirva como apoyo para la comunicación interna. Deberá favorecer la socialización de los intereses docentes y personales de estudiantes, profesores y trabajadores, creando múltiples espacios de interacción.

A continuación se describen los diferentes espacios de trabajo que se podrán encontrar en la aplicación, además se ilustran cada una de las secciones con el prototipo de interfaz que deberá tener dicha sección.

#### Página de Principal

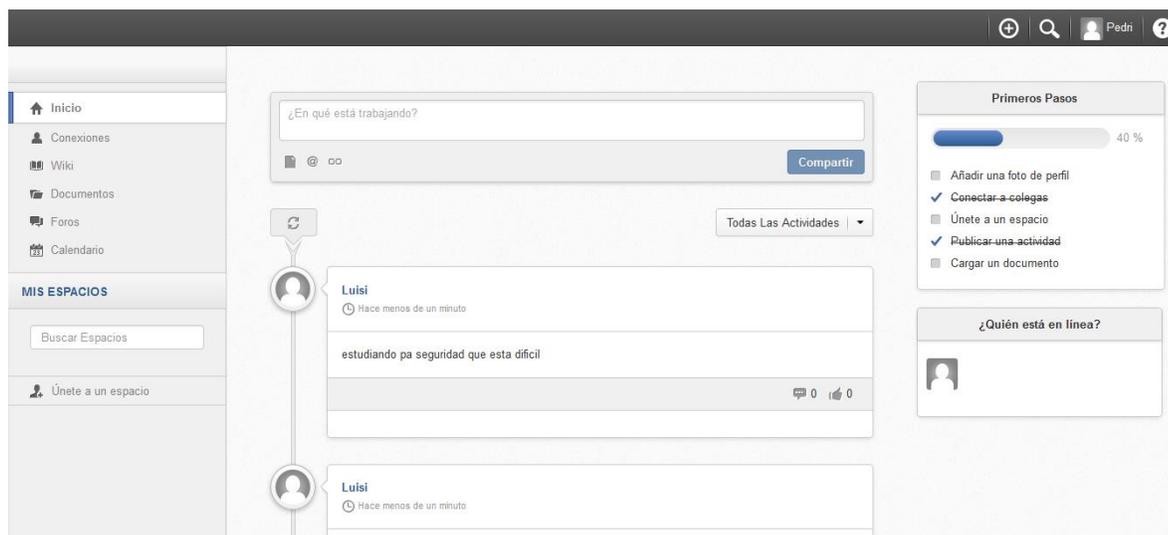


Fig. 1 Interfaz de la página principal

En la página principal se muestra al usuario su flujo de actualización donde puede publicar mensajes, compartir archivos u ofrecer enlaces a otras páginas. Además da la posibilidad de ver los mensajes publicados por él mismo y por otros usuarios así como los comentarios asociados a cada una de estas entradas. Este espacio deberá mostrar además los componentes necesarios que le permitan al usuario crear una entrada y hacerla pública. La sección deberá contar con una barra de menú que permita el acceso rápido a las demás secciones y facilitarle al usuario la navegación por el sistema.

## Mi perfil

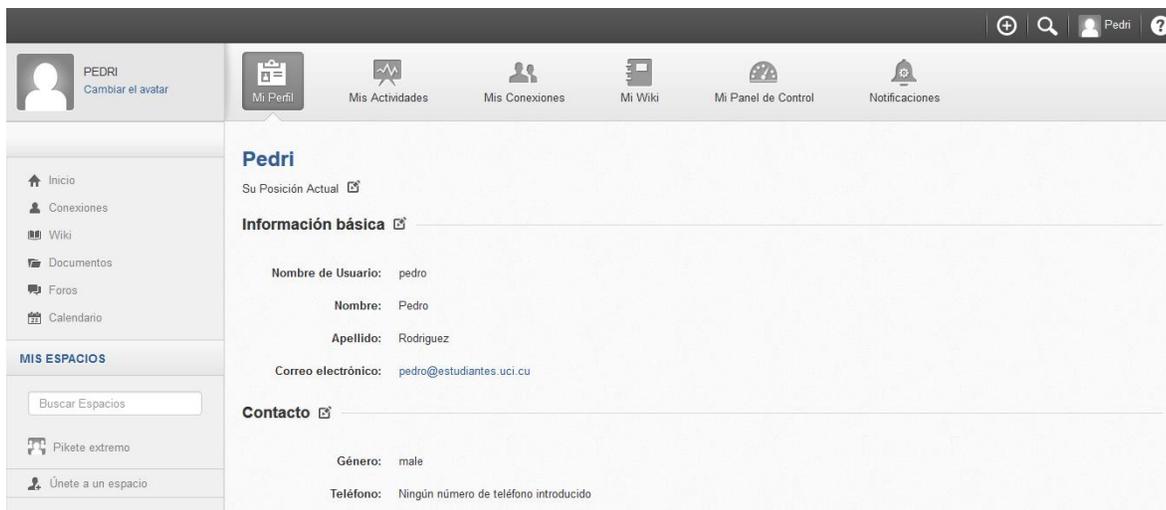


Fig. 2 Interfaz de perfil

El perfil de usuario permite editar toda la información referente a sí mismo. En esta sección puede actualizar información relativa a nombre de usuario, nombre, apellidos, correo electrónico, género y teléfono.

Este espacio deberá mostrar los componentes necesarios que le permitan al usuario modificar un conjunto de preferencias, entre las que se encontrarán: los datos de registro, el sistema de notificaciones y el lenguaje de su entorno.

## Conexiones

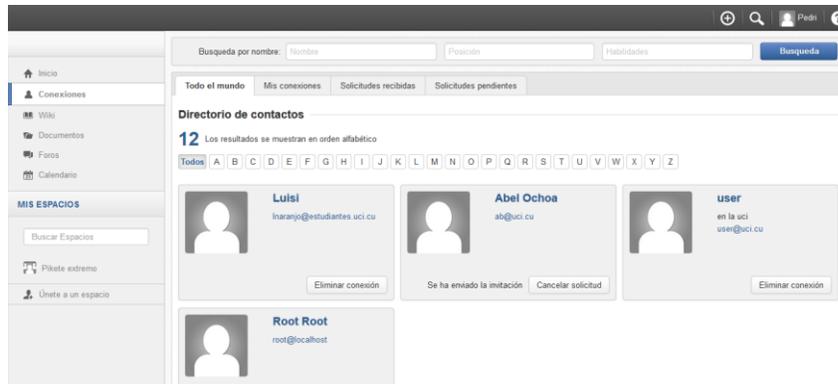


Fig. 3 Interfaz de la sección Conexiones

En la sección conexiones el usuario tendrá la opción de gestionar los usuarios con los que está relacionados. Puede enviar solicitudes de amistad a otros usuarios, cancelar solicitudes enviadas y denegar o aceptar las solicitudes de otros usuarios.

Además se definirán un conjunto de usuarios de alto nivel de importancia como decanos vicerrectores y otros directivos que deben estar incluidos en las cuentas de todos los demás usuario ya que estos difunden información de vital importancia para toda la Universidad favoreciendo de esta manera la comunicación interna.

## Wiki

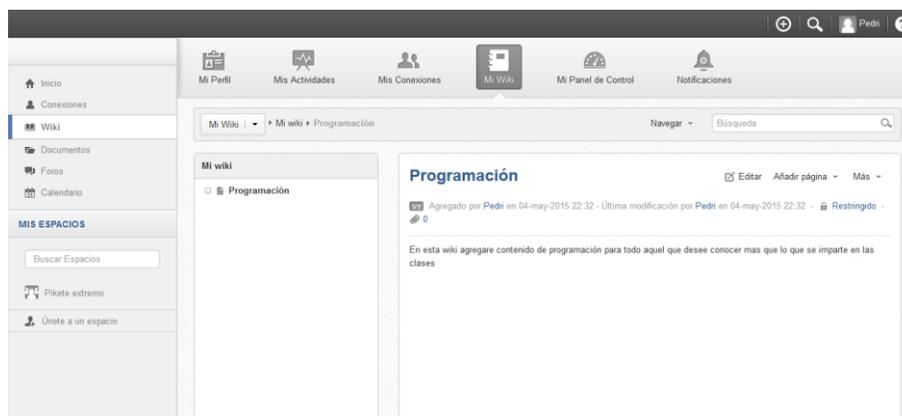


Fig. 4 Interfaz de la sección wiki

La sección de wikis sirve como medio colaborativo ya que un usuario puede invitar a otros a adicionar contenido en su página de forma que se cree una comunidad de usuario que comparten información sobre un mismo tema, enriqueciéndolo y obteniendo conocimiento de este intercambio.

## Documentos

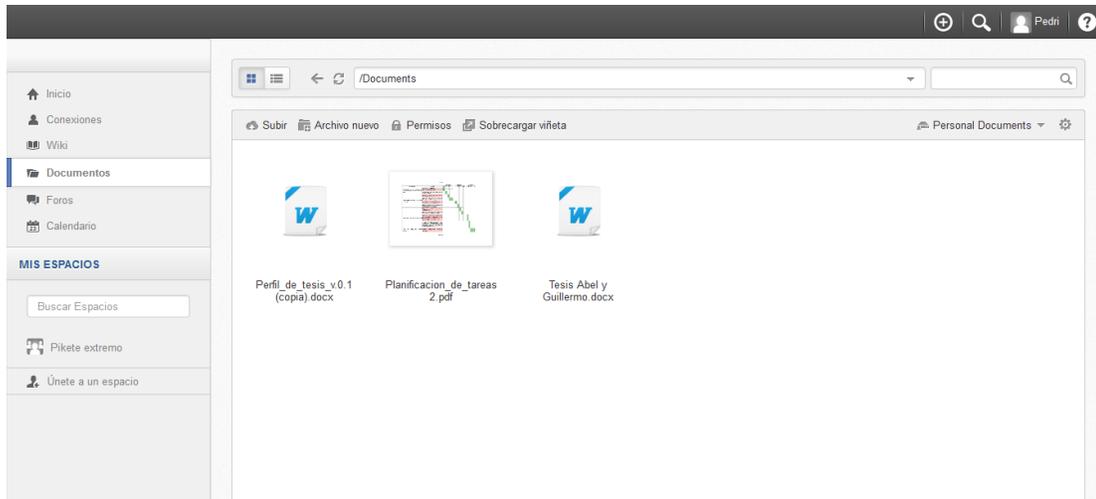


Fig. 5 Interfaz de la sección Documentos

En la sección de documentos el sistema brindará la posibilidad de acceder a todos los archivos que el usuario haya subido. El usuario deberá tener la posibilidad de añadir y editar archivos, además de que podrá incluir comentarios en cada uno. El sistema deberá admitir archivos de cualquier formato y el usuario deberá tener disponible la opción de descargarlo.

## Espacios

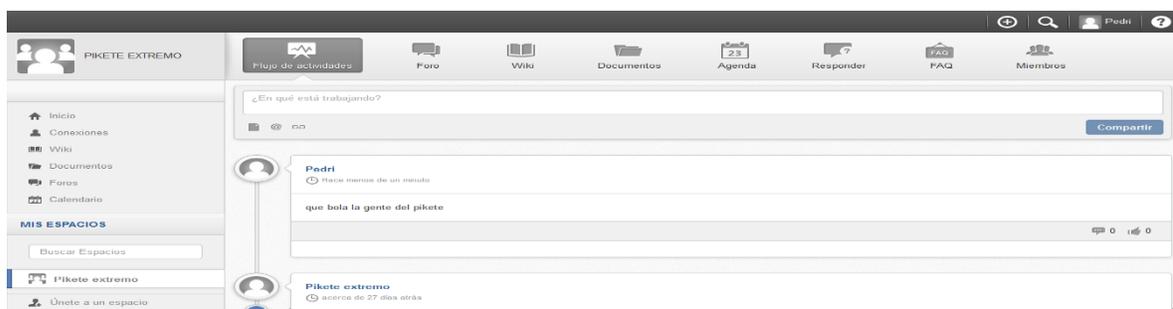


Fig. 6 Interfaz de la sección Espacios

En la sección “Espacios” se deberá mostrar un listado de todos los grupos de la red, así como las opciones necesaria para que el usuario autenticado acceda a una lista con los grupos a los que pertenece, los que administra y las invitaciones que ha recibido.

Los Espacios constituirán el método ideal para estimular el trabajo colaborativo, pues serán conformados a partir de los intereses individuales de sus integrantes. Se podrán subir entradas de blog, archivos, páginas, crear discusiones e incluir enlaces favoritos.

## 2.2 Requisitos funcionales

Los requisitos funcionales de un sistema describen lo que el sistema debe hacer. Estos requisitos dependen del tipo de *software* que se desarrolle, de los posibles usuarios y del enfoque general tomado por la organización al redactar el requisito. Los requisitos funcionales del sistema describen con detalle la función de este, sus entradas, salidas y excepciones (40), además permiten la creación de las historias de usuario descritas en la metodología (ver anexo 3).

Referencia	Requisitos funcionales	Prioridad
RF_1	Acceder al sistema a haciendo uso del servicio de directorio.	media
RF_2	Agregar usuario.	media
RF_3	Eliminar usuario.	media
RF_4	Mostrar listado de usuarios.	media
RF_5	Publicar mensajes en el flujo de actualización.	alta
RF_6	Mostrar mensajes en el flujo de actualización.	alta
RF_7	Cambiar idioma de interfaz.	media
RF_8	Editar perfil de usuario.	baja
RF_9	Crear grupo.	media
RF_10	Eliminar grupo.	media
RF_11	Mostrar listado de grupos creados.	media
RF_12	Asignar roles.	media
RF_13	Enviar solicitud de amistad.	alta

RF_14	Cancelar Solicitud de amistad.	alta
RF_15	Aceptar solicitud de amistad.	alta
RF_16	Denegar solicitud de amistad.	alta
RF_17	Editar notificaciones de actualización.	baja
RF_18	Restablecer notificaciones de actualización.	baja
RF_19	Activar <i>Portlet</i> base para extender portal uci.	alta
RF_20	Cambiar la selección de idioma mediante <i>portlet</i> .	alta
RF_21	Agregar al listado de gadget del portal los avisos del periódico de la Universidad.	alta
RF_22	Activar gadget del portal de noticias relevantes del periódico de la Universidad.	alta
RF_23	Agregar al listado de gadget del portal las principales noticias del periódico de la Universidad.	alta
RF_24	Extender o cambiar los diferentes estilos de la plataforma sin hacer uso de la interfaz gráfica de la aplicación.	alta
RF_25	Persistir los datos mediante un gestor de base de datos	alta

### 2.3 Requisitos no funcionales

Los requisitos no funcionales son aquellos que no se refieren directamente a las funciones específicas que proporciona el sistema, sino a las propiedades emergentes de este como la fiabilidad, el tiempo de respuesta y la capacidad de almacenamiento. Definen las restricciones del sistema como la capacidad de los dispositivos de entrada/salida y las representaciones de datos que se utilizan en las interfaces del sistema (41).

#### Portabilidad

- El sistema será multiplataforma siendo soportado por Windows y Linux.

## **Eficiencia**

- La solución propuesta debe soportar un tiempo de 5 segundos máximo por petición.

## **Escalabilidad**

- El sistema debe permitir la actualización del núcleo de la plataforma sin necesidad de alterar sus demás componentes.

## **Interfaz**

- La comunicación entre el servidor de aplicaciones y la base de datos se lleva a través de la arquitectura TCP/IP.
- La comunicación entre el cliente y el servidor de aplicaciones se realiza a través del protocolo HTTPS.

## **Seguridad**

- La información privada contenida en la plataforma y manejada por los usuarios debe ser protegida de acceso no autorizado haciendo uso de mecanismos de autenticación mediante el uso de usuario y contraseña.
- El sistema en caso de ocurrir algún error debe recuperarse si es posible, realizando un correcto manejo de las excepciones y registrando en trazas los eventos que ocasionan funcionamientos indebidos para un posterior análisis.

## **Hardware:**

- El sistema requerirá un servidor con 4gb de memoria RAM como requerimiento mínimo aunque se recomienda 8gb, procesador *MultiCore* a 2.0 GHz o superior y contar con un almacenamiento mínimo en disco de 10gb. Estos requerimientos están determinados por los requisitos mínimos de hardware de la plataforma.

### Software:

- El sistema requiere del servidor Apache Tomcat 7.0.55 instalado y la Máquina Virtual de Java 7.0, con la variable de entorno JAVA\_HOME definida, para su funcionamiento.
- Un servidor de base de datos PostgreSQL previamente instalado y configurado.

## 2.4 Roles del sistema

En la siguiente tabla se muestran los roles que determinan el nivel de autorización de cada usuario que interactúa con el sistema.

Rol	Permisos
Administrador	Además de contar con todas las funcionalidades de un usuario básico, puede gestionar usuarios registrados en el sistema, los espacios creados por estos y puede asignar permisos de administración a otros usuarios. Además este puede definir tantos roles como sean necesarios y otorgar a estos los permisos requeridos.
Miembro	El usuario básico tiene acceso a su flujo de actualización, puede publicar entradas, y crear espacios.

## 2.5 Descripción de la arquitectura

Según Pressman la arquitectura de *software* de un programa o sistema de cómputo es la estructura o las estructuras del sistema, que incluyen los componentes del *software*, las propiedades visibles externamente de los componentes y las relaciones entre ellos (42). Es un conjunto de patrones que proporcionan un marco de referencia necesario para guiar la construcción de un *software*, permitiendo a los programadores, analistas y todo el conjunto de desarrolladores del mismo compartir una misma línea de trabajo y cubrir todos los objetivos y restricciones de la aplicación.

### 2.5.1 Arquitectura cliente-servidor

Cliente-servidor es una arquitectura informática que separa a un cliente de un servidor y casi siempre se lleva a cabo a través de una red informática. El tipo más básico de la arquitectura cliente-servidor emplea

sólo dos tipos de nodos: clientes y servidores. Este tipo de arquitectura es referida a veces como de dos niveles (ver figura 1).

**Cliente:** permite al usuario formular los requisitos y pasarlos al servidor. Maneja todas las funciones relacionadas con la manipulación y despliegue de datos; por lo que están desarrollados sobre plataformas que permiten construir interfaces gráficas de usuario, además de acceder a los servicios distribuidos en cualquier parte de una red. Algunas de sus funciones son administrar la interfaz de usuario, interactuar con el usuario, procesar la lógica de la aplicación, hacer validaciones locales, generar requisitos de bases de datos, recibir resultados del servidor y formatear resultados.

**Servidor:** encargado de atender a múltiples clientes que hacen peticiones de algún recurso administrado por él. Maneja todas las funciones relacionadas con la mayoría de las reglas del negocio y los recursos de datos. Las funciones del servidor son aceptar los requisitos de bases de datos que hacen los clientes, procesar esos requisitos, formatear datos para transmitirlos a los clientes, procesar la lógica de la aplicación y realizar validaciones a nivel de bases de datos.

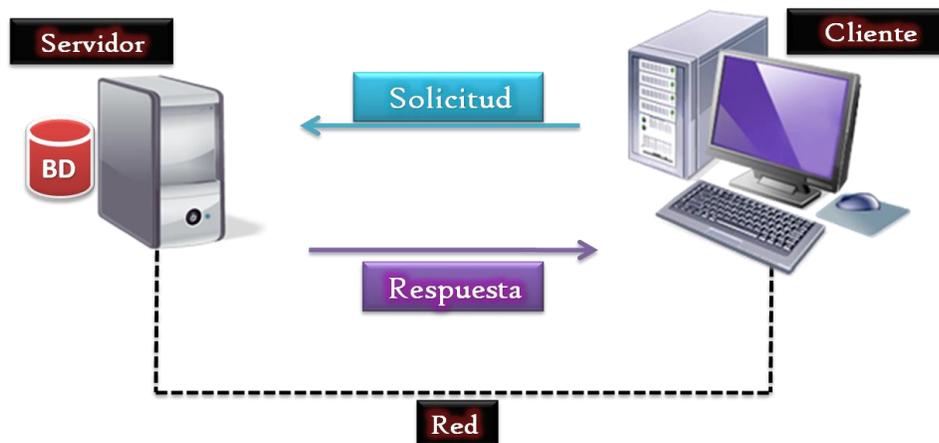


Fig. 1 Arquitectura cliente-servidor

## 2.5.2 Arquitectura de Exo platform

Exo platform utiliza una arquitectura por capas la cual permite la separación de los diferentes componentes que integran el sistema (Ver figura 2).

## **Arquitectura por capas**

La programación por capas es un estilo de programación en el que el objetivo primordial es la separación de la lógica de negocios de la lógica de diseño; un ejemplo básico de esto consiste en separar la capa de datos de la capa de presentación al usuario.

### **Ventajas de la arquitectura por capas**

- Reutilización de capas.
- Facilita la estandarización.
- Dependencias se limitan a intra-capa.
- Contención de cambios a una o pocas capas.

### **Desventajas de la arquitectura por capas**

- A veces no se logra la contención del cambio y se requiere una cascada de cambios en varias capas.
- Pérdida de eficiencia.
- Trabajo innecesario por parte de capas más internas o redundante entre varias capas.
- Dificultad para diseñar correctamente la granularidad de las capas.

## **Componentes de la arquitectura de Exo platform**

A continuación se definen todos los componentes y capas de Exo platform (ver figura 2).

### ***Portlets***

Los *portlets* son componentes modulares de las interfaces de usuario gestionadas y visualizadas en un portal web. Los *portlets* producen fragmentos de código de marcado que se agregan en una página de un portal. Típicamente, siguiendo la metáfora de escritorio, una página de un portal se visualiza como una colección de ventanas de *portlet* que no se solapan, donde cada una de estas muestra un *portlet*. Por lo

tanto un *portlet* (o colección de *portlets*) se asemeja a una aplicación web que está hospedada en un portal. Por ejemplo, un *portlet* de aplicación puede ser para el correo, el parte meteorológico, un foro, noticias, etc.

### **Exo platform modules (Módulos)**

Esta capa se encarga de la interacción con el usuario. Contiene los *portlets* que el administrador defina para su portal.

### **Portal container (Contenedor de portal)**

El Contenedor de portal gestiona la información referente a todos los sitios creados desde la misma plataforma y permite la administración de cada uno por separado aunque estén hospedados sobre la misma base.

### **Portlet container (Contenedor de portlets)**

Contiene todo el conjunto de *portlets* por defecto y los que fueron creados para el diseño del sitio. Se crearon tres *portlets* fundamentales:

uciGadgets: Este *portlet* contiene todas las funcionalidades adicionales que se implementaron para agregar a la herramienta.

uciPortal: Contiene la información de los *portlets* activos del sistema y su disposición en el sitio.

uciSkin: Permite modificar el diseño del sitio y contiene las hojas de estilo que dan formato al contenido de la página.

**Core service (Servicios):** Esta capa provee a los demás componentes todas las funcionalidades relativas a los servicios como autenticación, seguridad, base de datos, así como la conexión con el servicio LDAP<sup>20</sup>.

---

<sup>20</sup> LDAP (*Lightweight Directory Access Protocol*): Protocolo Ligero de Acceso a Directorios. Es un protocolo a nivel de aplicación que permite el acceso a un servicio de directorio ordenado y distribuido para buscar diversa información en un entorno de red.

## Kernel

El *kernel* es la base de todos los módulos de la plataforma, se encarga de inicializar todos los servicios y componentes, además de contener toda la lógica del negocio.

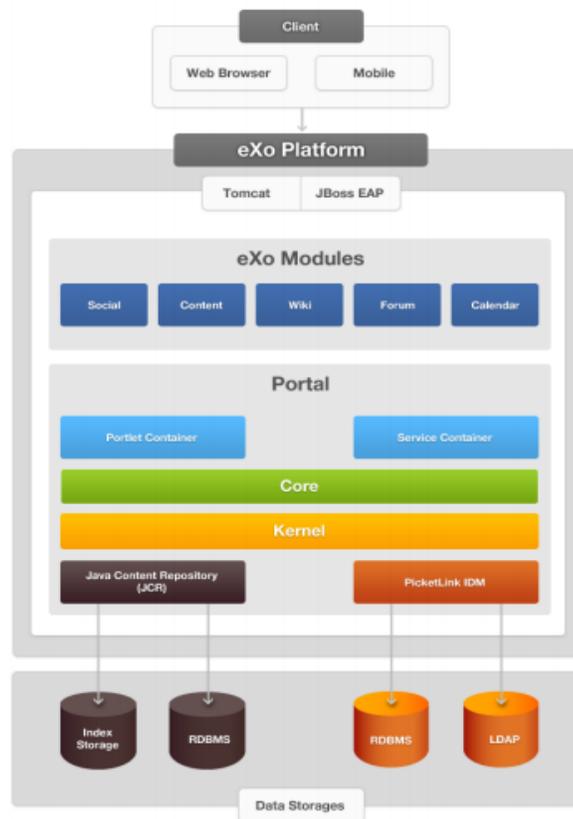


Fig. 2 Arquitectura de Exo platform (43).

### 2.5.3 Diagrama de despliegue

El Diagrama de despliegue es un diagrama estructurado que muestra la arquitectura del sistema desde el punto de vista del despliegue (distribución) de los artefactos del *software* en los destinos de despliegue. El diagrama de despliegue contiene instancias de nodos conectados por enlaces de comunicación. Las instancias de nodo pueden contener instancias de tiempo de ejecución, como instancias de componentes y objetos (ver figura 3).

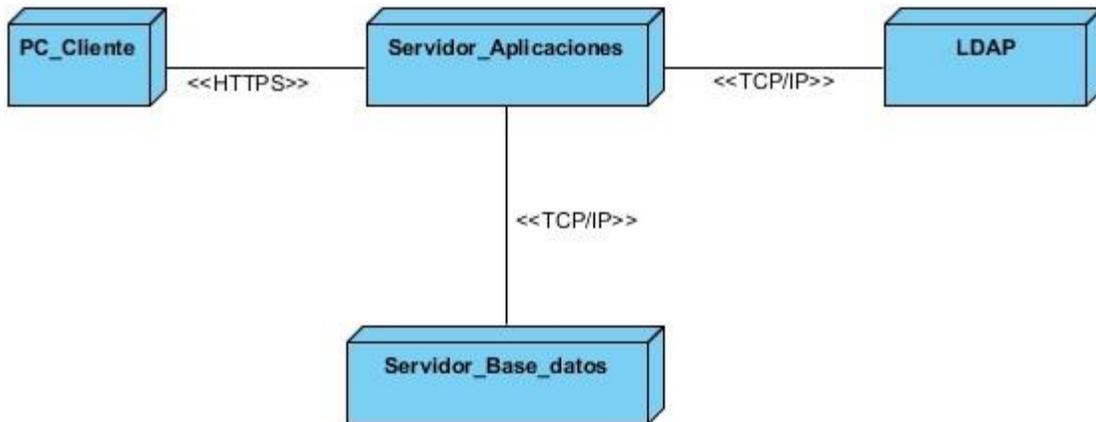


Fig. 3 Diagrama de despliegue

### Descripción de los componentes del diagrama

**Ciente:** El cliente cumple la función de acceder al sistema a través de un navegador haciendo uso del protocolo http.

**Servidor de aplicaciones:** El servidor de aplicaciones contiene toda la lógica del negocio así como el sistema y todas sus funcionalidades, es además, el encargado de atender todas las solicitudes del cliente y de realizar el acceso a la base de datos.

**Servidor de base de datos:** En este se encuentran todos los datos y la información que el cliente genera a través del sistema y posee protocolos de autenticación y autorización para la protección de esta información.

**Servidor LDAP:** En este servidor se encuentra el sistema de autenticación el cual provee al sistema de los nombres de usuario y las contraseñas, garantizando la seguridad de esta información.

**Protocolo HTTPS:** El Protocolo seguro de transferencia de hipertexto (*Hypertext Transfer Protocol Secure*), está destinado a la transferencia segura de datos de Hipertexto, es decir, es la versión segura de HTTP

**Familia de protocolos TCP/IP:** El modelo TCP/IP describe un conjunto de guías generales de diseño e implementación de protocolos de red específicos para permitir que un equipo pueda comunicarse en una red. Mediante este protocolo el servidor de aplicaciones se conecta con el servidor de base de datos.

## 2.6 Descripción del diseño

El diseño de *software* juega un papel importante en el desarrollo de *software* lo cual permite al ingeniero de *software* producir varios modelos del sistema o producto de que se va a construir el mismo que forman una especie de plan de la solución de la aplicación. Estos modelos pueden evaluarse en relación con su calidad y mejorarse antes de generar código, de realizar pruebas y de que los usuarios finales se vean involucrados a gran escala. El diseño es el sitio en el que se establece la calidad del *software*.

### 2.6.1 Patrones de diseño GoF

#### **Singleton (Instancia única)**

Restringe la creación de objetos pertenecientes a una clase o el valor de un tipo a un único objeto (ver figura 4).

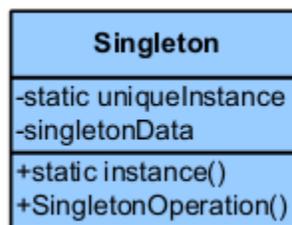


Fig. 4 Patrón *Singleton*

El *Exo platform Container* es un objeto responsable de la carga de los servicios / componentes. La clase *Exo platformContainer* es heredada por todos los recipientes, incluido *RootContainer* y *PortalContainer*. Esta clase hereda de un marco *PicoContainer* que permite a *Exo platform* aplicar los principios de Inversión de Control (también conocida como IOC).

El contenedor de servicios se basa en *Pico* con el fin de proporcionar manejo de Inversión de Control. Por lo tanto, los servicios no son responsables de la creación de instancias de los componentes de los que dependen; la creación de objetos se delega a un objeto *singleton: ServiceManager*. Esta arquitectura proporciona un diseño de acoplamiento flexible, donde los servicios dependientes se pueden intercambiar de forma transparente en tiempo de ejecución.

```

public OrganizationService {
    private CMSService service;

    public OrganizationService(CMSService service) {
        this.service = service;
    }

    //métodos que usa la interfaz CMSService
}

```

### **Facade**

Provee de una única interfaz para acceder a un sistema completo, que actúa como único punto de acceso al mismo, y hace que éste sea más fácil de utilizar. Se aplica este patrón cuando para una aplicación surge de la necesidad de desacoplar un sistema de sus clientes y de otros subsistemas, haciéndolo más independiente, portable y reutilizable (ver figura 5).

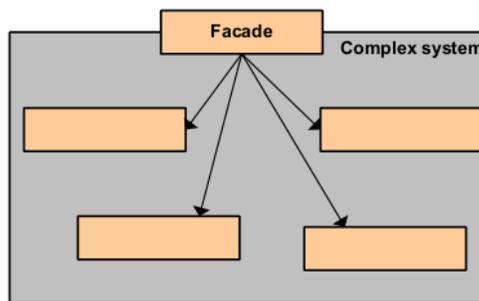


Fig. 5 Patrón *Facade*

El *Exo platformPortletContainerFacade*, una implementación del patrón de diseño *facade*, es el principal punto de entrada al contenedor. La fachada hace referencia a una agregación de objetos *PortletApplicationProxy*. Cuando se ha registrado en la aplicación, la fachada llamará a los métodos protegidos del proxy de aplicación para cargar, registrarse y crear instancias de los *portlets*.

## Composite

Este patrón define una estructura en forma de árbol, y permite construir objetos complejos a partir de otros más sencillos de una forma recursiva. En el mundo real podemos encontrar este patrón en la estructura organizativa de muchas empresas. Cuando el director le solicita al gestor un informe de progreso, éste solicita un informe a cada trabajador subordinado y combina los resultados para formar su informe (ver figura 6) (44).

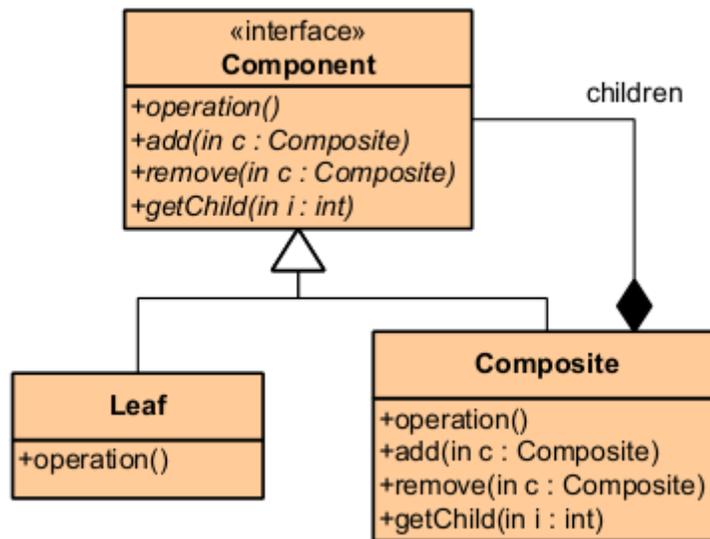


Fig. 6 Patrón Composite

Se utilizó este patrón para tener un repositorio único para todos los tipos de contenido y para proporcionar al desarrollador con un pequeño conjunto de interfaces de gran alcance y capacidades extensibles. La combinación de este patrón con programación orientada a aspectos permiten al *Exo platform CMS* ofrecer todas las características de un sistema de gestión de contenidos, tales como el almacenamiento en caché, control de versiones, bloqueo, indexación y búsqueda. Para la indexación, se utiliza el motor *Lucene* y para el control de caché el proyecto *Apache JCS*. Así, el desarrollador puede acceder, añadir, eliminar y modificar cualquier objeto de contenido en el depósito a través de los métodos *getNodeByUri()*, *addChild()*, *removeChild()*, *appendChild()* y *getParentNode()*.

## 2.6.2 Patrones de base de datos

El diseño y construcción de una base de datos requiere del mayor esfuerzo y análisis posible ya que a partir de este diseño es que se crea la base de datos. El siguiente modelo de datos representa como se relacionan las entidades de la plataforma (ver figura 7).

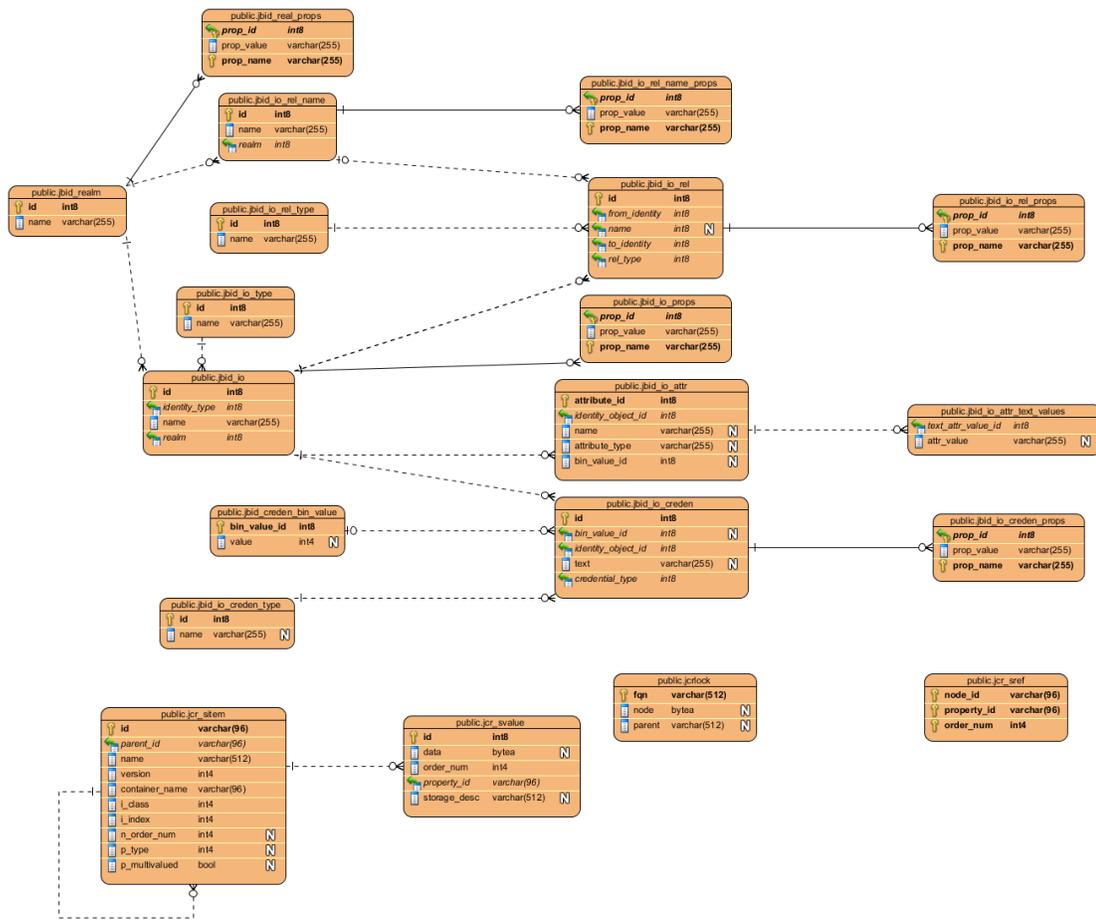


Fig. 7 Modelo de datos

En la actualidad las bases de datos suelen ser muy grandes y a veces el trabajo con los patrones de diseño hacen que el trabajo sea más fácil además asegura un resultado correcto. Estos patrones son utilizados mayoritariamente en los almacenes de datos.

## Llaves subrogadas

Este patrón se utilizó para generar una llave primara única para cada entidad en vez de usar un atributo identificador en el contexto dado. Normalmente se usa enteros en columnas *identity* o GUID (*Global UniqueIdentifier*) que están demostrados que no se repiten o con una probabilidad extremadamente baja. Permite que las tablas sean más fáciles de consultar por el identificador dado que se conoce el mismo tipo de todos en cada tabla.

## Árboles fuertemente codificados (*Hardcoded tree*)

A cada nodo se le asocia una entidad. En términos de base de datos las relaciones que existen entre las tablas pueden ser de uno a uno, de uno a muchos, de muchos a muchos, etc. en este caso las relaciones son de uno a muchos (n). El árbol fuertemente codificado es utilizado para representar jerarquías donde es bien conocida la estructura, es importante representar la correspondencia, por ejemplo las estructuras organizacionales (45).

Este patrón fue utilizado en la base de datos de modo que todas las relaciones existentes entre tablas son de uno a muchos eliminando las relaciones de muchos a muchos con la creación de una nueva tabla que contiene la información que converge en las entidades relacionadas.

## 2.7 Conclusiones Parciales

Los artefactos generados en este capítulo permitieron obtener una visión general de la propuesta de solución. Con la identificación de los requisitos se obtuvo las características que debe cumplir el sistema, logrando un entendimiento de lo que realmente se desea implementar. La especificación de requisitos, apoyado por la utilización de una arquitectura por capas y diversos patrones de diseño permitieron dar al *software* una estructura entendible y con un alto nivel de escalabilidad e interconexión.

## Capítulo III: Validación de la propuesta de solución

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 y los tipos de pruebas realizadas al sistema.

### 3.1 Estándares de codificación

#### Nombres de clases

Los nombres de clases deben ser mezclas de mayúsculas y minúsculas, con la primera letra de cada palabra interna en mayúsculas (*CamelCase*). Se debe intentar mantener los nombres de clase simples y descriptivos.

```
public abstract class AccessManager

public class Exo Container
extends org.Exo.container.management.ManageableContainer
```

#### Métodos

Los métodos deberán ser verbos (en infinitivo), en mayúsculas y minúsculas con la primera letra del nombre en minúsculas, y con la primera letra de cada palabra interna en mayúsculas (*lowerCamelCase*). No se permiten caracteres especiales. El nombre ha de ser lo suficientemente descriptivo, no importando a priori la longitud del mismo.

```
hasPermission(AccessControlList acl, String[] permission, Identity user)

setContext(InvocationContext context)
```

## Variables

Se evitará en la medida de lo posible la utilización de caracteres especiales, así como nombre sin ningún tipo de significado funcional.

Se estructurarán en mayúsculas y minúsculas con la primera letra del nombre en minúsculas, y con la primera letra de cada palabra interna en mayúsculas (*lowerCamelCase*).

```
String contextPath = request.getContextPath() ;  
  
String username = request.getParameter("username");
```

Las excepciones son las variables utilizadas en bucles *for*, para esos casos se permite utilizar i, j, k, l y siempre en ese orden de anidamiento. El primer bucle siempre será el que tenga la variable i como iterador.

## Comentarios

Los comentarios serán utilizados para dar información adicional al desarrollador sobre la implementación del diseño de la clase. Se tiene, por tanto, que evitar referencias al diseño funcional de la misma. Se tienen que evitar el uso de caracteres especiales dentro de los comentarios así como el uso de cajas u otro tipo de gráfico creado mediante códigos ASCII.

## Declaraciones

Las variables de avance de bucles *for* no podrán ser modificadas de ninguna manera fuera de la propia sentencia del bucle.

La duplicidad de los nombres de variables en diferentes niveles dentro de la misma clase se tiene que evitar.

## Sentencias

Normas básicas:

- Una sentencia por línea de código.
- Todo bloque de sentencias entre llaves, aunque sea una sola sentencia después de un *if*.

La declaración de los bucles *for* será usualmente de la forma:

```
for (int i = 0; i < condición ; i++)
```

Son obligatorias las tres condiciones del bucle *for*: inicialización, condición de finalización y actualización del valor de la variable de avance. La variable de avance del bucle nunca podrá ser modificada dentro del propio bucle.

## **Buenas prácticas**

### Constantes

Como norma general todas las constantes numéricas no deberían codificarse directamente, salvo la excepción de -1, 0 y 1.

### Propiedades

El acceso/modificación de las propiedades de una clase (no constantes) siempre mediante métodos de acceso *get/set*.

No utilizar el operador asignación en sitios donde se pueda confundir con el operador igualdad. Ni dentro de expresiones complejas.

### Métodos

Como norma general no se debe acceder a un método estático desde una instancia de una clase, debemos utilizar la clase en sí misma.

## **3.2 Validación**

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

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

### 3.2.1 Pruebas de *software*

El instrumento adecuado para determinar el estado de la calidad de un producto de *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 requisitos. Las pruebas de *software* constituyen un conjunto de herramientas, técnicas y métodos que evalúan el desempeño de un programa (42). 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:

- 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.

### Pruebas funcionales

Dentro de las pruebas funcionales se encuentra el método de caja negra, el cual está diseñado para validar los requisitos funcionales sin fijarse en el código fuente del programa. Las pruebas de caja negra permiten al ingeniero de *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 el método de caja blanca. Las pruebas de caja negra intentan encontrar errores de las siguientes categorías:

- 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.

## Descripción de casos de prueba

A continuación se muestran algunos de los casos de prueba realizados al sistema con el objetivo de detectar no conformidades (ver anexo 5).

Escenario	Descripción	Respuesta del sistema	Flujo central
Enviar solicitud	Mediante este escenario el usuario envía una solicitud a otro.	El sistema envía la solicitud al usuario deseado.	<ul style="list-style-type: none"> <li>El usuario una vez autenticado en el sistema selecciona la opción Conexiones en la barra lateral derecha.</li> <li>Puede realizar una búsqueda o buscar en el directorio de contactos el usuario al que desea enviar la solicitud.</li> <li>En la ficha del usuario selecciona la opción conectar</li> <li>El sistema enviara la solicitud al usuario deseado el cual puede aceptar o denegar.</li> </ul>
Cancelar solicitud enviada.	Mediante este escenario el usuario cancela una solicitud enviada.	El sistema cancela la solicitud previamente enviada.	<ul style="list-style-type: none"> <li>El usuario una vez autenticado en el sistema selecciona la opción Conexiones en la barra lateral derecha.</li> <li>Puede realizar una búsqueda o buscar en el directorio de contactos el usuario al que desea enviar la solicitud.</li> <li>En la ficha del usuario selecciona la opción conectar</li> <li>El sistema enviará la solicitud al usuario deseado.</li> <li>Luego de enviada la solicitud el usuario selecciona la opción cancelar solicitud.</li> <li>El sistema cancela la solicitud previamente enviada</li> </ul>

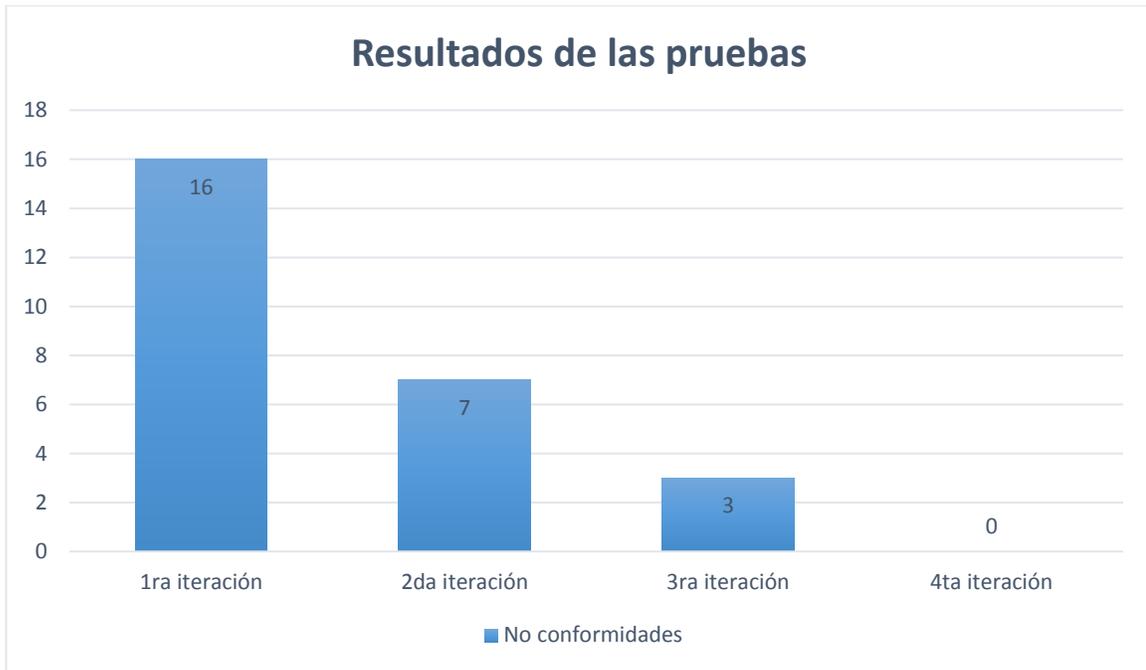
Descripción de caso de Prueba Enviar Solicitud de amistad

Escenario	Descripción	Respuesta del sistema	Flujo central
Cambiar el idioma de la interfaz	Mediante este escenario el usuario cambia el idioma de la interfaz.	El sistema cambia el lenguaje al definido por el usuario.	<ul style="list-style-type: none"> <li>El usuario una vez autenticado en el sistema selecciona la opción cambiar idioma en el menú de opciones de usuario.</li> <li>El sistema muestra un listado de los idiomas disponibles</li> <li>El usuario elige el idioma y selecciona la opción aplicar.</li> <li>El sistema cambiar el idioma del sitio al definido por el usuario.</li> </ul>

Descripción de caso de prueba Cambiar idioma de la interfaz

Mediante estos casos de prueba se pudo detectar errores en el funcionamiento del sistema los cuales fueron corregidos para alcanzar el grado de calidad esperado por el cliente. Se realizaron 4 iteraciones de pruebas de las cuales en la primera se detectaron 16 no conformidades que fueron corregidas. En la segunda

iteración se identificaron 7 no conformidades no detectadas en la primera iteración. En la tercera se detectaron 3 no conformidades y en la cuarta iteración no se detectaron errores en las funcionalidades. En el siguiente gráfico se ilustran los resultados:



### Pruebas de carga y estrés

Este tipo de pruebas se realiza con el objetivo de determinar la rapidez del sistema para realizar una tarea en condiciones extremas de trabajo. También son usadas para determinar la solidez de la aplicación en los momentos de carga extrema y ayudar a los administradores a determinar si la aplicación rendirá lo suficiente en caso de que la carga real supere a la carga esperada. Para ejecutar este tipo de prueba se utiliza la herramienta Jmeter en su versión 2.10 en la cual se emplea los siguientes parámetros:

- No. de usuarios: Cantidad de usuarios (hilos) conectados concurrentemente.
- Cantidad iteraciones: Cantidad de ciclos de prueba que se realizan con la misma cantidad de usuarios.
- Media: Promedio de respuestas obtenidas de la muestra.

- Mínimo: Menor número de respuesta obtenido de la muestra
- Máximo: Mayor número de respuesta obtenido de la muestra.
- 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.
- Tiempo de respuesta por petición: Tiempo que utiliza el sistema para dar respuesta a una petición.

A continuación se muestran los resultados luego de aplicar la prueba a la página de inicio del sistema

No. de usuarios	Cantidad de iteraciones	Media	Mínimo	Máximo	Error	Rendimiento (peticiones/seg)	Tiempo de respuesta por petición (ms)
1	4	25	25	28	0.00%	33.1/seg	30.1
20	4	153	25	427	0.00%	49.5/seg	20.4
50	4	981	479	1305	0.00%	41.2/seg	24.2
100	4	1807	50	2723	0.00%	46.6/seg	21.7
200	4	3280	1096	5216	0.00%	48.1/seg	20.7
400	2	6511	1499	9541	0.00%	45.9/seg	21.7
1000	2	13561	4935	21882	0.00%	37.5/seg	26.6

Estos resultados demuestran que el sistema es capaz de mantener una carga de hasta 1000 usuarios concurrentemente sin sufrir problemas de rendimiento.

### Entorno de Pruebas

Para estas pruebas se utilizó un equipo con los siguientes requisitos:

- Microprocesador *Intel Core i3* – 3120 a 2.5 GHz
- Memoria *RAM* de 8Gb
- Disco duro de 500 GB
- Sistema Operativo *Windows* 8.1

### **3.2.2 Pruebas de aceptación**

Estas pruebas las realiza el cliente. Son básicamente pruebas funcionales, sobre el sistema completo, y buscan una cobertura de la especificación de requisitos y del manual del usuario. Estas pruebas no se realizan durante el desarrollo, pues sería impresentable al cliente; sino que se realizan sobre el producto terminado e integrado o pudiera ser una versión del producto o una iteración funcional pactada previamente con el cliente. La experiencia muestra que aún después del más cuidadoso proceso de pruebas por parte del desarrollador, quedan una serie de errores que sólo aparecen cuando el cliente comienza a usarlo.

Se realizaron las pruebas de aceptación para comprobar que el sistema satisfacía las necesidades del cliente y contaba con las funcionalidades especificadas. Como constancia de su realización queda archivada el acta de aceptación (ver anexo 4).

### **3.3 Conclusiones parciales**

El análisis de los estándares de codificación utilizados en la plataforma permitió verificar que esta posee un código organizado, entendible y de fácil modificación. De ser extendida la plataforma se debe ser consecuente con estos estándares. A partir de la definición de casos de prueba y la realización de pruebas funcionales y de carga y estrés se logró detectar no conformidades que fueron corregidas permitiendo así un producto final con la calidad requerida por el cliente. El uso de pruebas de aceptación permitió verificar que el cliente estaba satisfecho con el sistema creado.

## **Conclusiones generales**

El análisis de diferentes herramientas de comunicación en la web, así como la selección de las tecnologías de desarrollo sentaron las bases para el desarrollo de la propuesta de solución.

La utilización de pruebas de aceptación, funcionales y de carga y estrés permitieron obtener un producto final con la calidad requerida y las funcionalidades definidas por el cliente.

Al término de este trabajo la comunidad universitaria cuenta con un espacio de interacción en línea y en tiempo real que aprovecha las potencialidades de las herramientas de la Web 2.0.

## **Recomendaciones**

Para este trabajo se proponen las siguientes recomendaciones:

- Actualizar sistemáticamente el núcleo del sistema para que este cuente con las nuevas mejoras de la plataforma.
- Analizar las potencialidades que puede presentar esta herramienta en otros centros educativos del país.

## Bibliografía

1. **Rivera, Arnulfo Perez.** La importancia de la comunicación humana. *Diario de Xalapa*. 18 de abril de 2011.
2. **Pérez Martinto, Pedro Carlos.** Entorno Virtual de Aprendizaje. [En línea] [Citado el: 3 de abril de 2015.] [http://eva.uci.cu/file.php/104/Tema\\_2/Recursos\\_bibliograficos/Diseno\\_metodol\\_de\\_la\\_invest-poblacion\\_y\\_muestra-\\_Metodos\\_y\\_diseno\\_experimental\\_-\\_Tema\\_3.pdf](http://eva.uci.cu/file.php/104/Tema_2/Recursos_bibliograficos/Diseno_metodol_de_la_invest-poblacion_y_muestra-_Metodos_y_diseno_experimental_-_Tema_3.pdf).
3. **Gonzalez, Fernando Rey.** *Comunicación, desarrollo y personalidad*. La Habana : Pueblo y Educación, 1995.
4. **Fonseca Yerena, Maria del Socorro.** *Comunicación Oral Fundamentos y Práctica Estratégica*. Primera. México : Pearson Educación, 2000. pág. 4.
5. **Chiavenato, , Idalberto.** *Introducción a la Teoría General de la Administración*. Séptima. s.l. : McGraw-Hill Interamericana, 2006. pág. 110.
6. **Fernández Collado, Carlos.** *La Comunicación en las organizaciones*. Mexico : Trillas, 1997. págs. 27-31. ISBN: 968-24-6586-9.
7. *El impacto de la tecnología en la comunicación empresarial: reflexiones y análisis.* **Lopez Jimenez, Iliá E.** 79, Mexico : s.n., julio de 2012, Primera Revista Electrónica en América Latina Especializada en Comunicación.
8. **Real Academia Española.** Real Academia Española. [En línea] [Citado el: 17 de febrero de 2015.] <http://lema.rae.es/drae/?val=blog>.
9. *Redes Sociales.* **Royero, Jaime.** 2007.
10. **Fernández Muñoz, Cristobal.** *LAS RELACIONES PÚBLICAS ANTE EL RETO DIGITAL Y EL FENÓMENO TWITER : ESTUDIO SOBRE LA RESPUESTA DE LAS AGENCIAS DE RRPP Y DEL SUBSECTOR EMPRESARIAL DE LAS BEBIDAS FERMENTADAS EN ESPAÑA*. Departamento de Comunicación Audiovisual y Publicidad, Universidad Complutense de Madrid. Madrid : s.n., 2012. págs. 69-82, 86-95, Tesis doctoral.
11. **Wales, Jimmy y Sanders, Larry.** Wikipedia. [En línea] Fundación Wikimedia, 15 de enero de 2001. [Citado el: 11 de marzo de 2015.] <http://es.wikipedia.org/>.
12. **Club Planeta.** Club Planeta. [En línea] [Citado el: 17 de febrero de 2015.] [http://www.cad.com.mx/historia\\_de\\_twitter.htm](http://www.cad.com.mx/historia_de_twitter.htm).
13. **GigaTecno.** GigaTecno. [En línea] enero de 2012. [Citado el: 19 de febrero de 2015.] <http://gigatecno.blogspot.com/2012/02/ventajas-y-desventajas-de-twitter.html>.
14. **Club Planeta.** Club Planeta. [En línea] [Citado el: 17 de febrero de 2015.] [http://www.cad.com.mx/historia\\_de\\_facebook.htm](http://www.cad.com.mx/historia_de_facebook.htm).
15. **GigaTecno.** GigaTecno. [En línea] enero de 2012. [Citado el: 19 de febrero de 2015.]

16. **Camacho, Herminio.** Desarrollan en Cuba servicio de microblogging. [ed.] Ricardo Ronquillo y Yailin Orta. *Juventud Rebelde. digital*, 2 de octubre de 2013.
17. **Consejo Editorial.** Cubadebate. [En línea] 5 de agosto de 2003. [Citado el: 11 de marzo de 2015.] <http://www.cubadebate.cu/editores/>.
18. **Ecured.** Ecured. [En línea] 14 de diciembre de 2010. [Citado el: 11 de marzo de 2015.] <http://www.ecured.cu/index.php/EcuRed>.
19. **Mahara ePortfolio System.** Mahara. [En línea] [Citado el: 2 de Mayo de 2015.] <https://mahara.org/>.
20. **Elgg Foundation Project.** Elgg Introducing a powerful open source. [En línea] [Citado el: 2 de Mayo de 2015.] <https://elgg.org/>.
21. **Exo Platform.** Exo. [En línea] [Citado el: 22 de febrero de 2015.] <http://www.exoplatform.com/company/es/products/features>.
22. **Groovy.** Groovy. [En línea] [Citado el: 3 de marzo de 2015.] <http://groovy-lang.org/>.
23. **Eguíluz Pérez, Javier.** *Introducción a JavaScript*. 2009. pág. 5.
24. —. *Introducción a xhtml*. 2007. págs. 6,7.
25. **W3C Working Group.** World Wide Web Consortium. [En línea] [Citado el: 3 de marzo de 2015.] <http://www.w3.org/standards/xml/core>.
26. **Object Management Group.** Unified Modeling Language. [En línea] 9 de octubre de 2014. [Citado el: 27 de marzo de 2015.] [http://www.omg.org/gettingstarted/what\\_is\\_uml.htm](http://www.omg.org/gettingstarted/what_is_uml.htm).
27. **Eguíluz Pérez, Javier .** *Introducción a CSS*. 2007. págs. 7,8.
28. **Álvarez, Miguel Angel.** Manual de JQuery: Desarrolloweb. *Desarrolloweb*. [En línea] 2015. <http://www.desarrolloweb.com/manuales/manual-jquery.html>.
29. **Eguíluz Pérez, Javier.** *Introducción a ajax*. 2008. pág. 5.
30. **Oracle.** NetBeans. [En línea] [Citado el: 4 de marzo de 2015.] <https://netbeans.org/features/index.html>.
31. **Apache Software Foundation.** Apache. [En línea] [Citado el: 4 de marzo de 2015.] url: [http://httpd.apache.org/ABOUT\\_APACHE.html](http://httpd.apache.org/ABOUT_APACHE.html).
32. **The PostgreSQL Global Development Group .** PostgreSQL-es. [En línea] 2 de octubre de 2010. [Citado el: 4 de Abril de 2015.] [http://www.postgresql.org.es/sobre\\_postgresql](http://www.postgresql.org.es/sobre_postgresql).
33. **pgAdmin Development Team.** pgAdmin. [En línea] [Citado el: 4 de marzo de 2015.] <http://www.pgadmin.org/docs/dev/introduction.html>.

34. **Pressman, Roger S.** *Ingeniería de Software un enfoque práctico*. Quinta. s.l. : McGraw-Hill Companies, 2002. ISBN: 8448132149. .
35. **Apache Software Foundation.** Apache Jmeter. [En línea] [Citado el: 17 de mayo de 2015.] <http://jmeter.apache.org/>.
36. **Larman, Craig.** *Agile & Iterative Development. A Manager's Guide*. [ed.] Addison Wesley. 2005, 2 y 3, págs. 7-9, 9-39, 109-210.
37. **Eclipse.** Eclipse. [En línea] 2011. <http://epf.eclipse.org/wikis/openup/>.
38. **Sutherland, Jeff.** *The Scrum Papers: Nut, Bolts, and Origins of an Agile Framework*. 2012.
39. **Beck, Kent.** *Extreme Programming Explained. Embrace Change*. [ed.] Addison Wesley. s.l. : Pearson Education, 1999.
40. **Sommerville, Ian.** *Ingeniería de Software*. [ed.] Miguel Martín Romo. 7ma. Edición. Madrid : PEARSON EDUCACION, S.A., 2005. ISBN: 84-7829-074-5.
41. —. *Ingeniería de software*. 6ta. Edición. Prentice-Hall : s.n., 2002. ISBN 970-26-0206-8.
42. **Pressman, Roger.** *Ingeniería de Software: Un Enfoque Práctico*. 6ta. Edición. s.l. : McGraw-Hill, 2005. pág. 900. ISBN: 9701054733.
43. **Exo Platform** . Exo Platform Documentation. [En línea] [Citado el: 4 de abril de 2015.] <http://docs.exoplatform.com/public/index.jsp?topic=%2FPLF42%2Fhome.html>.
44. **García Mondaray, Sergio.** GodTIC Un mundo binario. [En línea] 15 de Noviembre de 2012. [Citado el: 31 de marzo de 2015.] <http://www.godtic.com/blog/2012/11/15/patrones-de-diseno-gof/>.
45. **Blaha, Michael.** *Patterns of Data Modeling*. Primera. 2010. ISBN-13: 978-1439819890.
46. **Ansedo Espiñeira, Pablo.** *La comunicación en las organizaciones en la sociedad del conocimiento*. Escuela Universitaria de Relaciones Laborales da Coruña. 2010. Tesis de Grado.
47. **Rodríguez Guerra, Ingrid.** Comunicación organizacional: teorías y puntos de vista. *Comunicacion organizacional*. La Habana, Cuba : s.n., febrero de 2005.
48. **López Jimenez, Daniel.** *Comunicación Empresarial*. Primera. s.l. : ECOE EDICIONES LTDA., 2008. págs. 50-51. ISBN 9586484599.
49. **Paz, Jerónimo Perez.** Jerónimo Perez Paz - Aficiones y experiencias. *Jerónimo Perez Paz - Aficiones y experiencias*. [En línea] 15 de marzo de 2008. [Citado el: 17 de febrero de 2015.] <http://www.jeronimoperez.com/blog/que-es-un-microblog/>.

50. **Kioskea group.** kioskea. [En línea] marzo de 2015. [Citado el: 24 de marzo de 2015.] <http://es.kioskea.net/faq/4146-el-microblogging-privado-en-la-empresa#el-microblogging-en-la-empresa>.
51. **Vázquez Garcés, Sandra.** Redes sociales corporativas: una tendencia empresarial. 13 de octubre de 2013.
52. **GuiaSalud.** Slideshare. [En línea] 26 de septiembre de 2010. [Citado el: 19 de febrero de 2015.] <http://es.slideshare.net/GuiaSalud/2-web-20microblogging>.
53. **Rodríguez Sánchez, Elena y Menéndez Menéndez, Sonia.** Microblogging18. [En línea] digital, 2011. [Citado el: 19 de febrero de 2015.] <http://microblogging18.blogspot.com/2012/01/para-que-utilizar-las-herramientas-de.html>.
54. **Postgres Global Development Group.** *Tutorial de PostgreSQL*. [ed.] Thomas Lockhart. 1996.
55. **Sánchez Méndez, Alelí.** *Proceso de desarrollo de software DAC*. La Habana : s.n., 2013.