



UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE DEPARTAMENTO DE CIENCIAS DE
LA COMPUTACIÓN

DISEÑO Y DESARROLLO DE UNA HERRAMIENTA DE REPRESENTACIÓN Y
VISUALIZACIÓN DE REDES SOCIALES CON CAPACIDADES DISTRIBUIDAS

MEMORIA PARA OPTAR AL TÍTULO DE TÍTULO DE INGENIERO CIVIL EN
COMPUTACIÓN

FELIPE ANIBAL RICARDO ESPINOZA CASTILLO

PROFESOR GUÍA:
CLAUDIO GUTIÉRREZ GALLARDO

MIEMBROS DE LA COMISIÓN:
GONZALO NAVARRO BADINO
LUIS MATEU BRULE

SANTIAGO DE CHILE
JULIO 2013

RESUMEN DE LA MEMORIA
PARA OPTAR AL TÍTULO DE:
INGENIERO CIVIL EN COMPUTACIÓN
POR: FELIPE ANÍBAL RICARDO ESPINOZA CASTILLO
PROFESOR GUÍA: CLAUDIO GUTIÉRREZ GALLARDO
FECHA: 12/07/2013

**DISEÑO Y DESARROLLO DE UNA HERRAMIENTA DE
REPRESENTACIÓN Y VISUALIZACIÓN DE REDES SOCIALES CON
CAPACIDADES DISTRIBUIDAS**

A mi familia, a mi polola, a mis amigos y mi profe guía.

Agradecimientos

Gracias a mí por ser tan genial xD.

Índice general

Introducción	1
1. Descripción del Proyecto	4
1.1. Objetivos	4
1.1.1. Objetivo General	4
1.1.2. Objetivos Específicos	4
1.2. Resultados Esperados	5
1.3. Alcances	5
2. Marco Conceptual	6
2.1. Conceptos de Redes Sociales	6
2.1.1. Definiciones	6
2.2. Conceptos de Desarrollo	6
2.2.1. Desarrollo Web	6
2.2.2. Modelo MVC	8
2.2.3. Frameworks de Desarrollo	8
Conclusión	9

Índice de tablas

Índice de figuras

2.1. Modelo MVC	8
---------------------------	---

Introducción

Existe una red social llamada **LittleSis**[?] que consiste en una base de datos de relaciones de *quien conoce a quien* entre gente política, económica y socialmente poderosa en el mundo de las organizaciones en Estados Unidos cuyo fin es entregar el poder de la información al pueblo.

El origen del nombre de *LittleSis* se relaciona con el personaje *Big Brother* de la novela de George Orwell llamada *Nineteen Eighty-Four*. Este personaje consiste en un dictador que maneja toda la información sobre la población. *Big Brother* posteriormente fue un concepto con el cual se describió un ente poderoso.

De esta forma, al grupo compuesto por las personas poderosas y políticos de un país se les atribuye esta imagen de *Big Brother*, por su poder e influencia en un país. Esta es la motivación para bautizar con el nombre *LittleSis* (*Little Sister*), en oposición a *Big Brother*, a una aplicación cuyo fin es entregar poder de la información a toda la población.

La idea detrás de *LittleSis* tiene la capacidad de ser replicada en muchos países con el mismo fin, esto es, el de entregar poder al pueblo e informarle sobre los posibles conflictos de interés que las autoridades locales puedan tener al lidiar con el ejercicio de su poder, lo que en el caso particular de Chile, existe una iniciativa llamada **Poderopedia**[?] que busca ser una réplica de *LittleSis* para Chile.

En el caso de **Poderopedia**, uno de los líderes de ese proyecto es el ex alumno del DCC, Álvaro Graves, quienes tienen un **modelo RDF publicado en un repositorio en Github***, ellos están desde hace un tiempo cercanos a el lanzamiento de un beta público, de cualquier manera, por lo expresado en las redes sociales y en su página oficial, consiste en esta herramienta de las mismas características de LittleSis, es decir una base de datos centralizada sobre las relaciones del ambiente político chileno.

Con toda sus potencialidades, tanto *LittleSis* como *Poderopedia*, poseen un fin muy particular. En este sentido, sólo tienen la capacidad de cubrir la información sobre las personas muy importantes del mundo político, social y económico poderoso. Pero si una persona quisiera recolectar información sobre grupos de personas (por ejemplo el grupo de poder de una pequeña ciudad) para formar una red sociales, estas aplicaciones no sirven. Tampoco estas

herramientas permiten la creación de redes sociales privadas, que cumplan los objetivos que el usuario desee, pero al mismo tiempo permitirle interactuar con los datos de otras redes sociales pertenecientes a otras personas o comunidades.

El objetivo de esta memoria es llenar ese vacío, diseñando y desarrollando una herramienta para representar redes sociales, y que tengan capacidades distribuidas, donde los usuarios que poseen sus redes sociales públicas o privadas, puedan unirlos a voluntad.

Esta herramienta posee una gran gama de aplicaciones posibles, tanto para estudios de tipo sociológico, histórico, biológico, político, etc. Un caso de ejemplo reciente es el caso de las redes de autoridades e intereses posibles dentro del ambiente educacional chileno, lo que puede dar una visión más informada sobre el entramado de intereses detrás de los problemas denunciados por el conflicto estudiantil que se vive desde el año 2011.

Para lograr desarrollar esta herramienta se deben sortear una serie de desafíos técnicos, que aunque pareciera una aplicación bastante intuitiva, hasta el momento no hemos encontrado una aplicación con estas características.

Entre sus mayores desafíos está el de tener una aplicación de fácil instalación y uso amigable para los usuarios. Por otro lado, con lo que respecta al modelamiento de datos, debe basarse en algún estándar flexible de representación de redes sociales que permita la interoperabilidad. Esta memoria partirá de la base y la experiencia del recientemente titulado doctor en computación Mauro San Martín, cuya tesis de doctorado consistió en diseñar un modelo para el manejo de redes sociales, de esta forma, se aprovechará el conocimiento de su experiencia de investigación, aplicándolo a un trabajo práctico y útil para la investigación y estudio de varias disciplinas.

También habrá que desarrollar el aspecto de la visualización de datos, la integración con otras herramientas enfocadas al análisis de redes sociales, el modelamiento y uso de la información de redes sociales.

Dado que la estructura de redes sociales está fuertemente ligada a grafos, y se necesita un estándar aceptado (para el que existan herramientas de desarrollo disponibles), se ha optado por representarlo en el modelo **RDF**. Este estándar es además es una representación que permitirá todo tipo de usos incorporándose a tecnologías de la **Web Semántica**. Finalmente, además de los anteriores, hay desafíos de negocios que se vinculan con lo técnico, como hacer algunos casos de prueba de usabilidad para entregar una herramienta de valor a usuarios y contactar diversas personas que puedan estar interesadas en la herramienta.

En este informe de avance en el tema de memoria se expondrá los avances hechos en el semestre. En la sección se presenta el objetivo general del trabajo, de manera que el lector

comprenda resumida y claramente la visión y alcance de este trabajo. Luego en la sección de objetivos específicos se descompone el objetivo general en una serie de objetivos medibles incluyendo además objetivos tentativos con respecto al tiempo de lo que se pudiera extender este trabajo. Los avances principales en la memoria de expresan en la sección donde se expone la especificación del sistema. Luego, se discute el enfoque mismo de trabajo de esta memoria en la sección de plan de trabajo, en donde se dará a conocer la metodología general para desarrollar los objetivos, definir los entregables finales del trabajo y haciendo una estimación de tiempos por medio de una carta gantt.

Capítulo 1

Descripción del Proyecto

1.1. Objetivos

A continuación se presentan los objetivos para este trabajo de memoria.

1.1.1. Objetivo General

El objetivo de este trabajo consiste en crear una herramienta por la cual personas que estudian diversos tipos de redes sociales como: sociólogos, periodistas, biólogos, etc; puedan representar, administrar y visualizar redes sociales de mediana escala, contando además con la capacidad de combinar redes sociales con otros usuarios de la herramienta para obtener redes sociales con información más completa ofreciendo esto como un servicio centralizado con la capacidad de ser distribuido

1.1.2. Objetivos Específicos

De lo escrito anteriormente en el objetivo general, se desprenden los siguientes objetivos intermedios:

1. Implementar un modelo de representación de redes sociales en RDF.
2. Hacer una interfaz amigable de ingreso de datos: actores y relaciones.
3. Desarrollar un sistema centralizado que administre la asignación de identificadores únicos a los actores que los usuarios agregan a sus redes sociales.
4. Unir redes sociales, combinando la información y relaciones de actores en común.
5. Visualización
 - (a) De la estructura general de las redes sociales creadas.
 - (b) Específica de elementos de interés en esas redes sociales.

6. Representar algunas redes con el modelo implementado anteriormente, que varíen en tamaño y en complejidad.

1.2. Resultados Esperados

Para llevar a cabo los objetivos expuestos en esta memoria, los resultados que se esperan consisten en: un sistema que permita el modelamiento de redes sociales en donde debe existir una aplicación que permita de manera cómoda crear estas redes sociales, complementar su información para posteriormente unir las redes de un usuario con otro en caso de que este lo quiera. Además de lo anterior, los datos generados por el sistema deben estar en un formato amigable para computadores, que faciliten la posterior interoperabilidad con otras aplicaciones y fuentes de conocimiento.

1.3. Alcances

Es importante mencionar, que con el objetivo de que el trabajo a realizar cumpla con las limitaciones de tiempo correspondientes a una memoria de ingeniería, el problema de la creación y manipulación de redes sociales por personas se acota en los siguientes aspectos:

- El tamaño de redes sociales, se considerará pequeño con un número de alrededor 100 actores por red social, debido a que las redes serán principalmente creadas de forma manual, por tanto debe manejar un número que sea posible de alcanzar por los usuarios de la aplicación, de esta forma, ahorrando problemáticas asociadas con la escalabilidad de la aplicación para redes sociales más grandes.
- Dependiendo de las aplicaciones para las cuales se use el modelamiento de redes sociales, puede requerirse ver los cambios temporales que sufren estas, para analizar la estructura de una red social y su evolución en el tiempo. Dicho problema no será abordado en esta memoria.

Capítulo 2

Marco Conceptual

En este capítulo se describen los conceptos necesarios para que el lector se pueda familiarizar con mayor facilidad con los conceptos usados en el resto del informe, describiendo lo que se entiende por los conceptos usados en términos de redes sociales y conceptos asociados al desarrollo de la solución.

En caso de así preferirlo, el lector puede saltar las secciones que estime convenientes dentro de este capítulo, en caso de dominar los conceptos y usar este capítulo como referencia en caso de necesitarlo.

2.1. Conceptos de Redes Sociales

2.1.1. Definiciones

2.2. Conceptos de Desarrollo

A continuación se explican los términos en lo que a desarrollo que se refiere que tienen relación con el trabajo realizado en esta memoria.

2.2.1. Desarrollo Web

El desarrollo web se denomina al desarrollo de software cuyo fin es la creación de aplicaciones que se ejecuten en computadores que puedan ser accesados vía la *world-wide web*. Este estilo de desarrollo conlleva a que una aplicación que pueda estar ejecutándose en un servidor pueda entregar resultados a clientes provenientes de cualquier parte del mundo, usando una gran diversidad de dispositivos, plataformas de software, etc. La cual es frecuentemente

accesada por las personas vía un navegador web.

Esta propiedad de multiplataforma de las aplicaciones desarrolladas para la web, junto con propiedades de accesibilidad desde diversos medios a las aplicaciones, ha hecho que el desarrollo web sea un enfoque de desarrollo ampliamente usado en la industria en los últimos 10 años.

En términos generales el desarrollo web posee una arquitectura clásica de cliente/servidor, en donde una aplicación corriendo en un servidor proporciona resultados y ejecuta instrucciones proveniente de clientes, que pueden ir desde un navegador web de p.c. o dispositivo móvil, o servir de API para otras aplicaciones. A continuación se adjunta un diagrama que muestra los principales componentes cuando se habla de desarrollo web.

En la imagen se ve un ejemplo simple de una aplicación que posee un servidor de ejecución de la aplicación, con un servidor de base de datos, que son accedidos vía teléfono, tablet o computador por medio de un navegador web o de una interfaz API que use una aplicación nativa.

Dentro del desarrollo web, también se encuentran subcategorías que son expuestas a continuación.

Desarrollo Front-End

Es el desarrollo orientado a todos los componentes con los cuales interactúa un usuario de la aplicación, enfocado en la interfaz y sus componentes gráficos, además de la experiencia usuario, la usabilidad de la aplicación, etc. Es el desarrollo del cliente que el usuario usa para acceder al core de la aplicación.

En términos de tecnologías, se asocia el desarrollo front-end al uso de tecnologías como: HTML, CSS, Javascript, entre otras.

Desarrollo Back-End

Es el desarrollo de la aplicación que almacena y posee la lógica común a todos los clientes de la aplicación, en donde el código de este tipo de desarrollo es que se ejecuta en el servidor. Además generalmente en el desarrollo back-end se incluye todo lo relacionado a persistencia de datos generados a través del uso de la aplicación.

2.2.2. Modelo MVC

El modelo MVC (Modelo, Vista, Controlador)[7], consiste en un modelo para separar la lógica de una aplicación agrupándola en clases u otras unidades modulares, de acuerdo con la responsabilidad que estos módulos cumplan dentro de un sistema. A modo de ejemplo, a fin de ilustrar este concepto, se puede dar un caso de una aplicación que registre compras en un sistema de tienda online, un ejemplo de los módulos asociados a compras en MVC puede ser el siguiente:

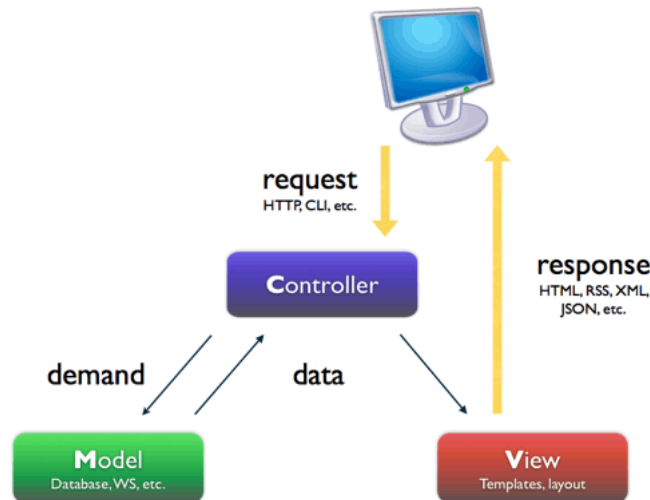


Figura 2.1: Modelo MVC

- **Modelo:** el modelo corresponde a una clase **Compra** que contiene toda la lógica de negocio asociada a las compras, que además se asocia directamente a como se almacena una compra en la base de datos.
- **Vista:** un ejemplo de vista para una compra, puede ser una interfaz **HTML** en la cual el cliente efectúe operaciones sobre la compra, por ejemplo agregar productos, cabe destacar que la vista no ejecuta las acciones, sólo se encarga de recibirlas y enviarlas al último componente del modelo MVC, el controlador.
- **Controlador:** de acuerdo a lo recién expresado, el controlador es el encargado de coordinar uno o más modelos para ejecutar las acciones capturadas en la vista. En el caso de la compra, el controlador es quien efectivamente procesaría el pago de la misma.

2.2.3. Frameworks de Desarrollo

Un framework de desarrollo define un marco en el cual desarrollar una aplicación, el cual provee una gran cantidad de funcionalidad común lista de manera de no desarrollar un proyecto desde cero, pero además de sólo funcionalidad, provee de una estructura lógica con la cual se escribe el código, que está influida por muchas otras personas que han usado el framework en aplicaciones reales.

Frameworks Server Side

Dentro de los frameworks de desarrollo web existe una categoría llamada *Sever Side*, la cual consiste en que el código de la aplicación corre desde un servidor en internet, lo que permite que si la computación es común para muchos clientes, los resultados de esa computación pueden ser usados múltiples veces.

Ejemplos actuales de esta categoría de frameworks son: Ruby on Rails[8], Sinatra[9] que utilizan el lenguaje Ruby; Django[4], Pylons[6] para Python; Spring[10] para Java y CakePHP[3] para PHP entre otros.

Frameworks Client Side

En el último tiempo, surgió una nueva categoría de frameworks de desarrollo web llamada *Client Side*, en la cual el código de la aplicación se ejecuta en el computador del usuario de la aplicación, ahorrando recursos necesarios en un servidor, además de ahorrar el tiempo de latencia entre el cómputo de una respuesta y su transmisión al equipo del usuario.

Comúnmente estos frameworks son escritos para ser usados con el lenguaje Javascript, pues posee la propiedad que todos los navegadores web implementan un motor de Javascript y por lo tanto el usuario no necesita instalar nada más. Ejemplos de frameworks client side son: AngularJS[1], EmberJS[11], Meteor[5] y alternativamente BackboneJS[2] que es una librería más que un framework.

Conclusión

Conclusión content

Bibliografía

- [1] Angularjs home page. <http://angularjs.org/>, June 2013.
- [2] Backbonejs home page. <http://backbonejs.org/>, June 2013.
- [3] Cakephp home page. <http://cakephp.org/>, June 2013.
- [4] Django home page. <https://www.djangoproject.com/>, June 2013.
- [5] Meteor home page. <http://meteor.com/>, June 2013.
- [6] Pylons home page. <http://www.pylonsproject.org/>, June 2013.
- [7] Rails guides, the mvc architecture. http://guides.rubyonrails.org/getting_started.html#the-mvc-architecture, June 2013.
- [8] Ruby on rails, home page. <http://rubyonrails.org/>, June 2013.
- [9] Sinatra home page. <http://www.sinatrarb.com/>, June 2013.
- [10] Spring home page. <http://www.springsource.org/>, June 2013.
- [11] Yehuda Katz. Emberjs home page. <http://emberjs.com/>, June 2013.