**Indignado Framework**

Vanessa Revetria

[lavarece@hotmail.com](mailto:lavarece@hotmail.com)

Juan Manuel Miraballes

[jmg216@hotmail.com](mailto:jmg216@hotmail.com)

Andrés Aldao

[andrew5900@hotmail.com](mailto:andrew5900@hotmail.com)

Maximilian Silvera

[maxi.silveraf@gmail.com](mailto:maxi.silveraf@gmail.com)

Gonzalo Castro

elgonzacastro@hotmail.com

**RESUMEN**

Debido a que los últimos años se ha incrementado la cantidad de grupos sociales que ejercen su derecho de protestar sobre temas sociales que no están de acuerdo, se decidió hacer un portal donde cada grupo pueda unificar su información y sus integrantes.

Para ellos se realizaran un sistema para administración, realizada en la plataforma .NET, pensado como una solución de escritorio, y para el usuario final del sistema, se realizara un sistema web utilizando también tecnología .NET.

[En el resumen se presentan, de forma breve, la motivación, contribución y resultados del trabajo. No extenderse demasiado. Por ejemplo: En los últimos años ha cobrado relevancia global el movimiento de protesta de los indignados .……. En este artículo se presenta el diseño e implementación de una plataforma informática de soporte a estos movimientos de protesta……. utilizando la plataforma .NET.] [Un par de oraciones describiendo los principales resultados y contribuciones del trabajo].

**Palabras Clave**

Catástrofes, .NET, Servicios Web, WCF, WPF, MVC 3, Razor, Movimientos de protesta, [, otras…]

**INTRODUCCIÓN**

[Motivación más extendida que en el resumen.]

[Describir resumidamente las particularidades de la solución propuesta, destacando la importancia del trabajo.]

[Se puede mencionar que el trabajo se plantea en el marco de la edición 2012 de la asignatura Taller de Sistemas de Información 1.]

Tal como detallamos en el resumen, en el último tiempo se ha incrementado la cantidad de grupos que sienten la libertad de protestar sobre un tema por el cual sienten el deseo de aportar su opinión.

Debido a que los grupos pueden estar extendidos a nivel mundial, se hace de suma importancia la implementación de un sistema donde cada integrante del grupo pueda acceder para conocer qué tipos de movimientos se están haciendo, en que lugares, que información se esta publicando en la web sobre dicho movimiento o sobre algún tema de interés de éste, etc.

En el marco de la edición 2012 de la asignatura Taller de Sistemas de Información 1, se plantea implementar una solución para dicho interés.

Para este sistema, surgió la necesidad de la implementación de dos sistemas, ya que surge la necesidad de desarrollar un sistema de tipo back office, donde se manejará la gestión del sistema web, de ahora en adelante llamado Front Office, y claro está la implementación del Front Office.

El resto del documento se organiza de la siguiente manera. La sección 2 presenta […]. En la sección 3 […]. Por último, en la sección […] se presentan las conclusiones del trabajo.

**MARCO CONCEPTUAL**

[En esta sección se introducen resumidamente los conceptos que puedan ser necesarios para la comprensión del artículo. En este caso se podría, por ejemplo, describir el concepto de multi-tenant, web services, etc.]

En esta sección se describen varios conceptos sobre los cuales se deben tener conocimientos para la implementación del sistema. Entre ellos la principal complejidad del sistema, que es la implementación del multi-tenant.

Otro concepto que se debe tener claro es la idea del manejo de información por web services donde......

También debemos tener en cuenta la arquitectura seleccionada, que separa el sistema en capas.

A continuación detallamos dichos conceptos.

**Multi Tenant**

Multi Tenant se refiere a un principio de arquitectura de software donde una única instancia de una aplicación se despliega en un servidor sirviendo a multiples clientes (tenants). **Multi Tenant** contrasta con una arquitectura **multi-instancia** en la cual son instancias independientes de software (o sistemas de hardware) que establecen para las organización de distintos clientes.  Con una arquitectura **multi-tenant**, una aplicación de software está diseñada para la  partición de sus datos y la configuración de manera que cada cliente trabaje con una instancia de la aplicación virtual personalizada. Este tipo de arquitecturas se denominan**multi-tenant** o **multipropietario**, en las que sobre un único recurso operan múltiples usuarios que son dueños, por así decirlo, del mismo.

**Web Service**

Los web services son funcionalidad que puden ser utilizadas por una aplicación en particular. Estas funcionalidades puden ser descriptas, publicadas, localizadas e invocadas a través de una red (generalmente Internet). Una de las características principal de los servicios es que éstos se encuentran en una espcecie de “caja negra”, lo que nos permite reutilizarlos sin importarnos como fueron implementados. La forma de accederlos es utlizando protoclos web como HTTP y XML, esto permite la interoperablidad entre las aplicacaciones, por lo cual, los consumidores de los web services pueden ser implementados en cualquier plataforma y en cualquier lenguje de programación.

**Arquitectura en capas**

La arquitecutra en capas es un estilo de programación donde su objetivo es separar los diferentes aspectos del desarrollo de una aplicación, como por ejemplo cuestiones de la presentación, lógica de negocio y mecanismos de almacenamiento. Comunmente la comunicación entre las diferentes capas se da mediante interfaces bien definidas que nos permiten la correcta separacion de las mismas. Además, este tipo de diseño permite la instalacion distribuida de una aplicación, mejorando su escalabilidad, disponibilidad y utilización de recursos.

**DESCRIPCIÓN DEL PROBLEMA**

Como indicamos la motivación del sistema reside en el diseño y construcción de un sistema para que los grupos de personas que desean centrar toda la información de su sitio de protesta puedan contar con una página web donde podrá publicar los contenidos deseados,

Las principales funcionalidades que debe proveer el sistema son la gestión de usuarios, gestión de sitios, gestión de variables y de reportes por parte del administrador y por parte del usuario final, se desarrollan las funcionalidades de gestión de usuarios, manejo de convocatorias, gestión de recursos compartidos y novedades y el chat. Aclaramos que el front office estará implementado para cada grupo independientemente. Esto indica que el usuario será un entidad independientemente para cada grupo.

Además, se plantean como funcionalidades opcionales de integración con el Facebook y Twitter, integración entre servidores, realizar un cliente Windows phone, detección de dispositiovos móviles e interfaz HTML5 y el deploy en Windows azure.

Por último, el sistema debe implementarse utilizando la plataforma .NET, con todas sus tecnologías integradas.

**SOLUCIÓN PLANTEADA**

Nuestra solución se basa en dos sistemas, como ya explicamos un Back Office y un Front Office, desarrollados con distintas tecnologías.

El Back Office se desarrollo utilizando Windows Presentatioon Foundation y el Front Office utilizando MVC 3 con Razor.

El Back Office contará con tres menues basados en la gestión del sistio, estos 5 menues son: Gestión de Administradores, Gestión de usuarios, Gestión de Grupos, Gestión de Variables y Gestión de Reportes. Cada menú de estos, tendrá la responsabilidad de gestionar lo que su nombre indica.

Luego en el Front Office tendremos una Home del sitio donde se desplegarán todas las funcionalidades de dicho sistema. Dicha home desplegará toda su información sólo para usuarios registrados. Los usuarios no registrados, sólo podrán registrarse o ver la información pública del sitio.

A continuación presentamos los casos de uso identificados en el sistema, donde cada uno de ellos, se detallará….

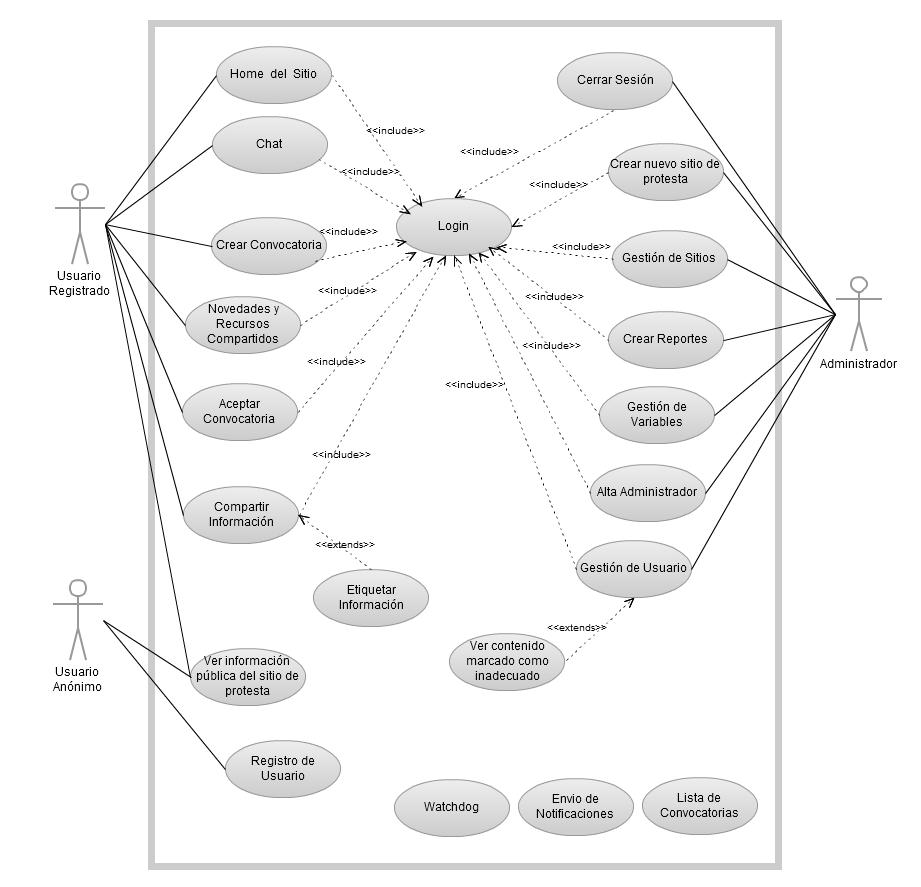


Figura 1: Detalle de casos de uso de la solución

Breve descripción de cada caso de uso:

***Front Office***

*Home del sitio*: Desplegará todas las opciones detalladas a continuación

Login: Única manera de que un usuario peuda ingresar al sitio, habilitandole todas las opciones

Chat: Sistema que se realizará para que los usuarios registrados puedan chatear entre ellos.

Crear Convocatoria: En esta sección se podrá ingresar todos los datos necesarios para crear una convocatoria. La convocatoria será visible para todos los usuarios registrados, donde ellos podran marcar su asistencia

Novedades y recursos compartidos: Se detallarán todos los recursos compartidos por los usuarios registrados al sitio. El recurso puede tener la visibilidad habilitada para todos los usuarios o sólo para usuarios registrados

Aceptar Convocatoria: Es la opción donde los usuarios registrados pueden indicar que asistirán a la convocatoria

Compartir Información: En esta sección se podrá subir y compartir la información deseada

Etiquetar Información: Toda información publicada podrá ser etiquetada, la idea es que el usuario tenga una notificación indicando que fue etiquetado.

Ver información pública del sitio: Se listará a los usuarios no registrados la información del sitio que fue publicada con visibilidad habilitada para todos los usuarios.

Cerrar sesión: Es la opción para que el usuario cierre su sesión

Crear nuevo sitio de protesta: Se creará un nuevo sitio, en una nueva url, con un diseño distinto.

Gestión de sitios: Podrá editar otros sitios creados

Crear Reportes: Se podrán crear reportes para sacar estadisiticas

Gestión de variables: Habrá ciertas variables que controlarán el sitio. Las variables podrán ser editadas aquí.

Alta Administrador: Se podrá dar de alta a otros administradores

Gestión de usuarios: Se controlará la actividad de los usuarios

Ver contenido marcado como inadecudo: Se permitirá ver todo el contenido marcado de manera inadecuada, para evaluar decisiones.

Watchdog: Proceso que dará de baja contenidos inadecuados

Envio de notificaciones: Es la notificación que se envía luego de crear una convocatoria

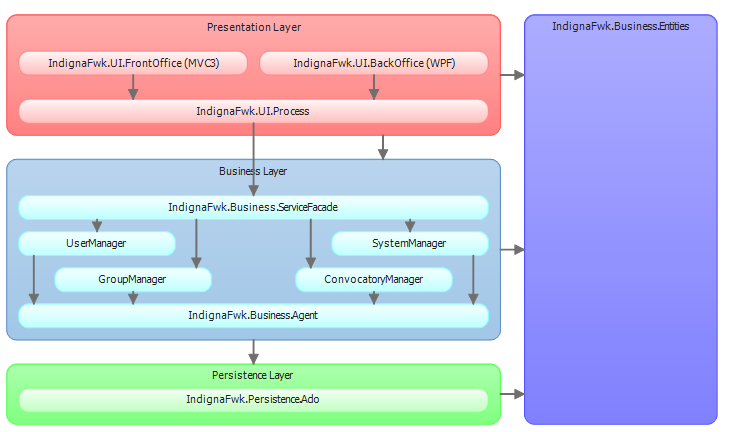
Lista de convocatorias: Se listarán todas las convocatorias creadas en el sitio.

[En esta sección se describe de forma general lo que se hizo, o sea, de todo lo planteado en la sección , a qué se llegó, qué opcionales se eligieron, etc. Sería conveniente incluir una figura o diagrama que diera una visión global de la solución.]

[Se podrían incluir sub-secciones para describir los puntos más relevantes de la solución, por ejemplo, requerimientos que se resolvieron de forma novedosa, algún punto que necesite más detalle para su comprensión, etc. Un punto interesante a destacar sería cómo se logró la integración con otros grupos.]

**ARQUITECTURA DEL SISTEMA**

Aquí presentamos la arquitectura del sistema.



**Aquí una explicación**

**IMPLEMENTACIÓN**

Se utiliza el IDE de Microsoft Visual Studio 2010 proffesional, integrado con el sistema de repositorio web Ankhsvn. Para la creación de la base de datos se utiliza Microsoft SQL Server 2008, con el IDE de SQL Server Management Studio.

Se utilizó, en la parte de FrontOffice, el patrón MVC con tecnologías ASP MVC 3 utilizando las vistas Razor.

Para la parte de BackOffice se utilizo Windows Presentation Fundation (WPF) utilizando C# realizando una aplicación de “escritorio”.

**Productos y Herramientas**

[En esta sub-sección se describen los productos y herramientas que se evaluaron y/o utilizaron, presentando brevemente una evaluación de los mismos. Se sugiere incluir una tabla para resumir la evaluación, incluyendo por ejemplo: puntos fuertes, puntos débiles, evaluación general, etc.]

**Tabla . Evaluación de Productos y Herramientas**

|  |  |  |  |
| --- | --- | --- | --- |
| **Producto** | **Puntos Fuertes** | **Puntos Débiles** | **Evaluación General** |
| [Producto 1] |  |  |  |
| [Producto 2] |  |  |  |

**Problemas Encontrados**

El mayor problema que se presento fue la implementación del multi-tenant donde…

**EVALUACIÓN DE LA SOLUCIÓN**

Los puntos fuertes de la solución fue la implementación del multi tenant,

[En esta sección se evalúa críticamente la solución identificando y mencionando brevemente sus puntos fuertes y puntos débiles.]

**DESARROLLO DEL PROYECTO**

[En esta sección se describe cómo fue el desarrollo del proyecto, porcentaje de tiempo dedicado a cada tarea, desviación con respecto al cronograma inicial y análisis de los posibles motivos que llevaron a que el proyecto se desarrollara de la forma descripta.]

**CONCLUSIONES Y TRABAJO A FUTURO**

[En esta sección se presentan las conclusiones y trabajo a futuro.]

**REFERENCIAS**

* Laboratorio de Integración de Sistemas. Taller de Sistemas de Información 1 – Trabajo Obligatorio. Año 2011. <http://www.fing.edu.uy/inco/cursos/tsi/TSI1/2011/laboratorio/Obligatorio2011.pdf> [Consulta: Mayo 2011]
* Microsoft .NET Framework

<http://www.microsoft.com/net/> - [Consulta: Mayo 2010]

* ACM Proceedings Templates  
  <http://www.acm.org/chapters/policy/toolkit/template.html>   
  [Consulta: Mayo 2010]