Tabla de contenido

[Presentación 2](#_Toc329750938)

[Descripción del problema 2](#_Toc329750939)

[Solución planteada 2](#_Toc329750940)

[Arquitectura de la solución 3](#_Toc329750941)

[Soluciones particulares 4](#_Toc329750942)

[Mejoras a la solución 7](#_Toc329750943)

[Problemas encontrados 7](#_Toc329750944)

[Desarrollo del proyecto 7](#_Toc329750945)

[Evaluaciones 8](#_Toc329750946)

[Demo 8](#_Toc329750947)

[Back-office 8](#_Toc329750948)

[Crear grupo 8](#_Toc329750949)

[Chequear variables del sistema 9](#_Toc329750950)

[Front-office 9](#_Toc329750951)

[Registro 9](#_Toc329750952)

[Login con facebook 9](#_Toc329750953)

[Login convencional 9](#_Toc329750954)

[Contenidos 9](#_Toc329750955)

[Compartir contenido 9](#_Toc329750956)

[Mapa full 10](#_Toc329750957)

[Crear convocatoria 10](#_Toc329750958)

[Chat 10](#_Toc329750959)

[Integración 11](#_Toc329750960)

[Windows Phone 11](#_Toc329750961)

# Presentación

## Descripción del problema

Motivación

* Rápido crecimiento de los movimientos de protesta.
* Expansión geográfica de estos movimientos.
* Generar un espacio para que estos grupos puedan expresar sus ideas, organizar actividades y compartir información (videos, imágenes, links, etc).

Objetivos principales

* Administración y coordinación de varios de estos movimientos.
* Coordinación de actividades (Convocatorias).
* Un único portal web por cada movimiento con diferente diseños (Distintos estilos).

## Solución planteada

Asdasd

* Desarrollo de una plataforma con el fin de centralizar toda información relevante a estos grupos.
* Implementación de dos módulos: FrontOffice y BackOffice.

**FrontOffice (Actor: Usuario)**

* Portal web de acceso público con autenticación de usuarios.
* Los usuarios podrán compartir información, realizar convocatoria a diferentes actividades, sumarse a las mismas y chatear.
* Mapa integrado para realizar las convocatorias.

**BackOffice (Actor: Administrador)**

* Aplicación de escritorio que gestiona los sitios de protesta con acceso restringido.
* Gestión de Grupos, usuarios y contenidos

## Arquitectura de la solución

**Enfoque principal**

Arquitectura de 3 capas que nos permitó separar los diferentes aspectos de la aplicación logrando una solución esclalable y mantenible.

**Presentation Layer**

* IndignaFwk.Web.FrontOffice  
   Módulo encargado de mostrar los datos al usuario final. Se ha implementado un proyecto web desarrollarlo.
* IndignaFwk.WPF.BackOffice   
  Aquí se implementa el módulo de BackOffice desarrollado como aplicación de escritorio.
* IndignaFwk.WP7.WindowsPhoneApplication  
  Módulo que contiene la aplicación nativa de Windows Phone.
* IndignaFwk.UI.Process   
  Módulo que consume los web services (WCF) expuestos en la capa de negocio. Se desarrolló una fachada encapsular las diferentes clases UI.Process.

**Business Layer**

Capa que contiene las reglas de negocio de la aplicación.

* IndignaFwk.Business.Services   
   Aquí están ubicados los web services (WCF), que exponen las funcionalidades hacia el exterior de la capa de negocio. Estos servicios son consumidos por la capa de presentación, en particual por el módulo IndignaFwk.UI.Process.
* IndignaFwk.Business.Managers   
  Este módulo resuelve parte de la lógica de negocio del framework. Se han creado diferentes managers y un fábrica para la correcta separación de las funcionalidades y su lógica.  
    
   Se comunica con la capa de acceso a datos, los agents y los servicios medianteintefaces.
* IndignaFwk.Business.Agents   
   Módulo utilizado para la integración con otros grupos y para la integración con fuentes de datos.

**Access Layer**

Capa de acceso a datos que se comunica directamente a la base de datos.

* IndignaFwk.Persistence.DataAccess   
  Único módulo dentro de esta capa el cual contiene todos los objetos ADOs (DataAccessObjects) .  
    
  Los objetos ADOs son los encargados de crear, editar, borrar y obtener los datos de la base.   
    
  Creación de interfaces para la correcta comunicación con la capa de negocio.

**Capa Vertical**

* IndignaFwk.Common   
  Capa que será referenciadas por las restantes (Presentation, Business, Persistence).  
    
  Contiene entidades de negocio, enumerados y funciones utilitarias.

## Soluciones particulares

**Aspectos Funcionales**

* MultiTenant

Arquitectura que permite proveer servicios a multiples clientes mediante una unica instancia.

Es posible crear un nuevo sitio (tenant) con solo crear un nuevo grupo en la BD.

Se generaron hasta 12 sitios gracias a la potencialidad de este enfoque.  
  
Para resolver este problema se utilizó el patrón Inversión de Control e Inyección de Dependencias.

*Detalle multitenant*

La aplicación web desarrollada permite reutilizar la implementación de la solución, para que cada sitio creado de una sensación de aplicación única, es decir, le sea transparente al usuario que por mas que ingrese a diferentes sitios el mismo esta siempre dentro de la misma aplicación web.

El problema que se planteo requería que cuando un usuario ingresara una URL la aplicación automáticamente detectara al sitio que se intentaba acceder y que todas las funcionalidades del sistema manipularan la información solamente del sitio que se ha accedido, o sea, el universo de datos de cada sitio es único dentro de la misma base de datos e independiente de los datos de los demás sitios existentes en la aplicación.

Para solucionar esto técnicamente decidimos hacer uso de la herramienta StructureMap, esta permite el uso de los patrones de inversión de control e inyección de dependencias, los cuales entre muchas otras cosas permiten desacoplar al máximo las dependencias entre si y la forma en la que se utilizan los objetos para no atarlos entre si pudiendo modificar uno de los objetos sin que los otros que usan de el se vean afectados, veamos un ejemplo sencillo:

Tengo una clase A que consume una operación de la clase B:

Class A  
{  
private B b;  
public A() {  
b = new B();  
}  
public void doSomething(){  
b.someMethod();  
}  
}

Si vemos el codigo, el realizar un cambio en el constructor de los elementos de B implicaria necesariamente realizar cambios en la implementacion de A para adecuarla a los nuevos cambios de B, este problema es solucionado por la inversión de control de la siguiente forma:

public class A  
{  
private B b;  
public A(B objetob)  
{  
this.b = objetob;  
}  
public void hago\_algo()  
{  
b.hacer\_algo();  
}  
}

Modifico el constructor de A para que reciba una instancia de B, donde B es una interface que puede tener varias implementaciones, ahora al recibirla y no crearla dentro del constructor el código no queda atado a ninguna implementación en particular y el método hago algo delegara a la instancia de B que se desee.

En nuestra solución se creo una clase que reemplaza el ControllerFactory original de MVC para generar instancias de los controllers pasándole como parámetro el sitio recuperado a partir de la URL ingresada, de esta forma toda la implementación estará orientada al uso del grupo particular.

* Integración entre Servidores

La integración permite compartir contenido publicado en otro servidor.

En nuestro caso se integró con otra instancia de nuestra misma aplicación. Generando una nueva base de datos para la nueva instancia.  
  
El servicio ServiceExpose, expone las operaciones a ser consumidas por el otro servidos y el InG4Agent es el componente que las consume.

* Chat Embebido

Chat similar al de facebook que permite simultaneas conversaciones con integrantes del sitio.

* Chat Tipo Sala

Funcionalidad aislada al resto que permite una unica conversación activa.

* Filtro de Búsqueda de Convocatorias

Funcionalidad que busca de forma dinámica las convocatorias de un sitio dependiendo de los parámentros ingresados en el filtro.

* Encriptación de Contraseñas  
     
  Encriptación de contraseñas de usuarios y administradores.  
    
   Se utilizó el método de encriptación Rijndael.  
    
  Encriptación desde la capa de presentación. Los datos viajan encriptados por web services y se guardan de la misma manera en la base de datos.
* Edición de Perfil de Usuario (Agregado Simple)  
    
  Funcionalidad simple que se agregó para que el usuario pudiera editar sus datos principales.
* Mapa   
    
  Creación de mapa full screen para mejorar la experiencia de usuario

## Mejoras a la solución

* Controles de seguridad previstos
* Recuperacion Usuario y contaseña
* Confirmación correo electrónico
* Control de Captcha
* Comentarios en contenidos y convocatorias
* Mejoras de diseño en el BackOffice
* Mayor integración con fuentes de datos
* Autenticación con Twitter, etc.
* Implementar más funcionalidades para Windows Phone
* Desplegar en Azure

## Problemas encontrados

* **Login con Facebook**

Debido a politicas de Facebook, se requiere un Id por URL, por lo tanto diseñamos que al registrar un nuevo grupo se permita ingresar dicho id.

* + **Chat**

Dificultad para testear el correcto funcionamiento.

* **Cierre de Sesión Imprevisto**

Se implementó un mecanismo que cuando la sesión del web server caduca, se marca el usuario como desconectado.

## Desarrollo del proyecto

Se adoptó un mecanismo de implementación horizontal (desde el back-end hacia el front-end). Esto consiste en realizar todas las tareas del mismo tipo en una misma etapa avanzando capa por capa (Entidades - > ADOs -> Managers -> etc.)

* **Ventajas** 
  + Optimización del tiempo y el trabajo de cada recurso.
  + Prevención de errores.
  + Mecanismo de implementación único en los elementos del mismo tipo.
  + Código facilmente mantenible por cualquier recurso de equipo.
* **Desventajas** 
  + Recodificación masiva por errores encontrados tardiamente.
  + Sensación de avance lento al comienzo del proyecto.
  + Menor distribución del conocimiento por centrarse en una única tarea.

*Frente a tantos temas a investigar y tan poco tiempo para crear la solución, se decidió utilizar esta técnica de desarrollo que nos permitió tener un avance constante.*

## Evaluaciones

* Tecnologías usuadas
* Curso
* Trabajo en Grupo
* Docente

# Demo

## Back-office

### Crear grupo

Autenticarse en el backoffice e ir directo a la creación de grupo, ingresar los siguientes datos:

* Nombre: Sistema operativo android
* Descripcion: Ventajas y desventajas del sistema operativo android
* Mapa: posicionarlo en sao-paulo

*Comentar que la ubicación del grupo es utilizada para posicionar los mapas del front, por ejemplo cuando se abre el mapa selector paqra indicar la ubicación de los usuarios y la de nuevas convocatorias, el mapa aparece en las coordenadas del grupo. Tambien sucede en el mapa Full screen.*

* AppId de facebook: 504123102936837

*Explicar que si el campo queda vacio quiere decir que el sitio web no tendrá habilitada la autentiicacion con facebook, por el contrario si. Recordar que este AppId es proporcionado por facebook y debe ser valido ya que si no la funcionalidad en el front no funcionara correctamente.*

* Fuentes externas:
  + YouTube: 3 elementos con keyword: Android
  + Wikipedia: 2 elementos con keyword: Android
* Layout: 1
* Tematica: Tecnologia (4)
* Url: so.android

Ir al archivo hosts de Windows y mostrar como el administrador debe agregar el DNS mapeado a 127.0.0.1 que es donde en este momento está iniciado el servidor web, agregar la línea: 127.0.0.1 so.android

### Chequear variables del sistema

Ir a la pantalla de parámetros del sistema y comentar que seria aconsejable el administrador haga un chequeo de las variables del sistema, principalmente la que afectan los contenidos que se muestran en el front, es decir la que filtra los “N” contenidos mas ranqueados.

*Abrir el Chrome y levantar sitio con la url* [*http://so.android/FrontOffice*](http://so.android/FrontOffice)*. Resaltar que se visualizan las fuentes externas que se ingresaron en el back. Contarselas mostrar que son 3 y 2.*

*Ir al acceder y mostrarles que se ve el login con facebook.*

## Front-office

Levantar el sitio http://zona.panda/FrontOffice , comentar que se va a utilizar este sitio ya que tiene contenido precargado a mano el cual puede calificarse y usuarios ingresados.

Mostrar como en el home se pueden ver los contenidos públicos, que no es posible calificar ya que no estoy autenticado.

### Registro

Ir a la pantalla de registro y mostrar cómo se da de alta un usuario, al abrir el mapa recordar lo comentado en el back con respecto a la posición en la que aparece. No dar de alta un usuario.

### Login con facebook

Ir a la pantalla de login y presionar el botón de facebook, autenticarse con una cuenta de real y presionar aceptar, aparecerá una pantalla solicitando las coordenadas ya que es el único dato faltante obligatorio. Comentar esto sucede ya que es necesario crear en el sistema un usuario con los mismos datos facebook para que el mismo pueda realizar las tareas como cualquier otro usuario.

### Login convencional

Cerrar la sesión iniciada con facebook y autenticarse de manera convencional con un usuario registrado a mano.

### Contenidos

Mostrar cómo se ven en el home más contenidos, el orden en el que aparecen es:

1. Ingresados a mano ordenados por calificación (mas ranqueados) y luego fecha (muestra los “N” variables del sistema).
2. Luego los contenidos de fuentes externas.
3. Los contenidos de sitios de integración sin ningún orden.

Resaltar que aparecen la cantidad de me gusta y la cantidad de inadecuados y que es posible modificar la votación de una a otra pero no eliminarla.

El ranking es en orden ascendente y se determina pos los que tienen más “megusta” y si hay varios con la misma cantidad se utiliza la fecha como segundo parámetro de ordenamiento.

### Compartir contenido

Se ingresa a la opción Compartir Contenido, comentar que tipos de contenido se pueden compartir y que hay una ayuda para el usuario. Decir que aparecerá un preview de los tipos de contenido validos o error si no es valido, una vez guardado mostrar como queda en la vista inicial si es que la cantidad “N” lo considera.

### Mapa full

Presionar el link del mapa full y mostrar que el mismo estará disponible al usuario desde cualquier lugar del sitio, el mismo muestra las convocatorias cercanas al grupo y un link de acceso a las mismas. Ingresar a una y mostrar la pantalla, de ahí crear una nueva.

### Crear convocatoria

Indicar que en esta página es donde se listan todas las convocatorias, luego de crear la convocatoria, aparecerá en esta grilla.

Ir a crear una nueva y comentar que el mapa de selección aparece en las coordenadas del grupo, que las fechas deben ser consecutivas y la inicial no puede ser menor a la actual, finalizar creación.

Mostrar que la convocatoria quedo ingresada en la página de convocatorias.

Comentar sobre la notificación creada y decir que la mismas se muestra ya que la convocatoria fue creada a menos de 50km de la ubicación del usuario, la notificación quedara ahí hasta que el usuario la marque como leída.

Resaltar que las notificaciones les llegan a todos los usuarios registrados que cumplan la condición que se encuentran a menos de 50 km.

Mostrar que en la descripción de la convocatoria aparecen las asistencias y el quórum y que es posible marca como que asisitire o no asistiré a las que quiera.

### Chat

Comentar como funcionan las dos soluciones de chat propuestas:

* Chat embebido: por un lado esta el chat de el sitio que estamos viendo, la idea de este es permitir al usuario que permanentemente este visualizando los usuarios que se conectan y se desconectan con la posibilidad de chatear simultanemante con varios, al abrir una ventana de chat se mostrara la conversación y se podrá continuar con la misma, si un usuario recibe mensajes y la ventana no esta abierta se mostrar una notificación de los mensajes recibidos.
* Chat tipo sala: Por otro lado diseñanos un chat tipo sala, el cual es una funcionalidad aparte en el sistema, para acceder al mismo se debe hacer click en la opción de menú y este no permite conversaciones simultaneas, solo una a la vez.

No consideramos que uno sea mejor al otro ya que cada sitio dependendiendo de sus necesidades (contenidos, tematica, etc.) puede requerir un tipo de chat u otro para que la dinámica del sitio sea más eficiente.

## Integración

Explicar que la integración se realizara contra nosotros mismos, para lo cual desplegamos una nueva aplicación web y aplicación de servicios en el mismo servidor pero distinto puerto.

Comentar que se va a acceder a esta aplicación la cual esta bajo la URL <http://abuso.animal/FrontOfficeInt>.

Decir que por el momento las aplicaciones integran con los Contenidos y las Convocatorias de la misma tematica, es decir, al ingresar a el sitio abuso animal que tiene como tematica “Naturaleza” el sitio recuperara todos los contenidos y convocatorias de la aplicación FrontOffice original donde la tematica de esos datos sea “Naturaleza”. Mostrar como dicho sitio carga los mismos contenidos (Ingresados a mano, o sea, no los extraidos de fuentes externas obvio) que se podían ver en el sitio zona panda de la aplicación original.

## Windows Phone

Se cambiara la pc a la de Gonzalo y el habla sobre la aplicación nativa desarrollada.