

PLAN DE TRABAJO - PROYECTO ANONYMSG



Versión del documento	5.0
Descripción de la versión	Entrega Final. Modificación de datos para publicación en O2.
Fecha	03 de julio de 2013
Consultor	Jordi Ceballos Villach
Autora	Deyris Drake Neninger

TABLA DE CONTENIDOS

1. INTRODUCCIÓN	4
1. DEFINICIÓN GENERAL DEL PROYECTO	4
1.1. OBJETIVOS	5
1.2. FUNCIONALIDADES PRINCIPALES	5
2. CALENDARIO DEL PROYECTO.....	6
2.1. DETALLE DEL CALENDARIO DE TRABAJO	6
2.2. RECURSOS E INFRAESTRUCTURA	8
2.2.1. RECURSOS HARDWARE	8
2.2.2. RECURSOS SOFTWARE.....	8
3. RIESGOS DEL PROYECTO	9
4. REQUERIMIENTOS FUNCIONALES	9
4.1 FUNCIONES PARA USUARIOS EN DISPOSITIVOS WEB O MOBILE	10
4.2 FUNCIONES PARA USUARIOS ADMINISTRADORES	10
5. REQUERIMIENTOS NO FUNCIONALES	10
5.1 USABILIDAD	11
5.2 AGILIDAD EN LAS BUSQUEDAS	11
5.3 ACCESIBILIDAD	11
6. FUNCIONALIDADES Y USUARIOS DEL SISTEMA	11
6.1 USUARIOS DEL SISTEMA	11
6.2 FUNCIONALIDADES DE USUARIOS	12
6.3 FUNCIONALIDADES DE USUARIOS ADMINISTRADORES	12
7. CASOS DE USO	12
7.1 DESCRIPCIÓN DE CASO DE USO [CU00] – INICIAR SESIÓN	14
7.2 DESCRIPCIÓN DE CASO DE USO [CU000] – VERIFICAR SESIÓN	14
7.3 DESCRIPCIÓN DE CASO DE USO [CU01] –BÚSQUEDA	14
7.4 DESCRIPCIÓN DE CASO DE USO [CU010] – INFORMACIÓN DE BÚSQUEDA	15
7.5 DESCRIPCIÓN DE CASO DE USO [CU011] – GEOLOCALIZAR MENSAJES.....	15
7.6 DESCRIPCIÓN DE CASO DE USO [CU02] – ORDENAR MENSAJES ENCONTRADOS	16
7.7 DESCRIPCIÓN DE CASO DE USO [CU03] – LISTAR MENSAJES	17
7.8 DESCRIPCIÓN DE CASO DE USO [CU04] – CALIFICAR	17
7.9 DESCRIPCIÓN DE CASO DE USO [CU05] – PUBLICAR	18
7.10 DESCRIPCIÓN DE CASO DE USO [CU050] – GEOLOCALIZAR	18
7.11 DESCRIPCIÓN DE CASO DE USO [CU06] – FINALIZAR SESIÓN	19
7.12 DESCRIPCIÓN DE CASO DE USO [CU07] – LOGIN.....	19
7.13 DESCRIPCIÓN DE CASO DE USO [CU08] – LISTAR STOP WORDS.....	19
7.14 DESCRIPCIÓN DE CASO DE USO [CU080] – ADMINISTRAR STOP WORDS/	20
7.15 DESCRIPCIÓN DE CASO DE USO [CU09] – DENUNCIAR.....	20
7.16 DESCRIPCIÓN DE CASO DE USO [CU10] – LISTAR DENUNCIAS.....	21
7.17 DESCRIPCIÓN DE CASO DE USO [CU100] – ADMINISTRAR DENUNCIAS	21
7.18 DESCRIPCIÓN DE CASO DE USO [CU11] – FINALIZAR SESIÓN ADMIN	22
8. DISEÑO TÉCNICO.....	22
8.1 PARTE SERVIDOR.....	22
8.2 PARTE CLIENTE.....	22
8.3 DISEÑO LÓGICO.....	23

9.	ARQUITECTURA DE BASE DE DATOS	24
9.1	DESCRIPCIÓN DE TABLAS	25
9.2	DESCRIPCIÓN DE ENTIDADES MÁS IMPORTANTES	25
9.2.1	Entidad PLACE	26
9.2.2	Entidad MESSAGE	26
9.2.3	Entidad USER_ADMIN	26
9.2.4	Entidad ADMIN_ACTION	27
10.	DIAGRAMA DE CLASES.....	27
10.1	DESCRIPCIÓN DE LOS SERVICIOS Y RELACIÓN CON LAS CLASES	28
10.2	DIAGRAMA DE CLASES DE LOS SERVICIOS.....	28
11.	DIAGRAMA DE SECUENCIA.....	30
11.1	DIAGRAMA DE SECUENCIA – CASO: MOSTRAR MENSAJES GEOLOCALIZADOS EN UN LUGAR ESPECÍFICO.....	30
12.	PROTOTIPOS	31
12.1	PROTOTIPOS PARA MOBILE	32
12.2	PROTOTIPOS PARA WEB.....	36
12.3	PROTOTIPOS DEL PORTAL DE ADMINISTRACION	38
13.	IMPLEMENTACION	39
13.1	ORGANIZACIÓN DE LA ESTRUCTURA DEL PROYECTO	40
13.2	REPOSITORIO DE SOFTWARE	41
13.3	TEMPLATES HTML5 BOILERPLATE	42
13.4	ORGANIZACIÓN DE CSSs.....	42
13.5	GEOLOCALIZACIÓN.....	44
13.6	JSON Y JQUERY	44
13.7	SINCRONIZACIÓN CLIENTE-SERVIDOR	45
13.7.1	DESARROLLO PARA LA VISUALIZACIÓN MOBILE	46
13.8	DESARROLLO PARA ANDROID	48
13.9	EJECUCION DE LA APLICACIÓN DURANTE LA ETAPA DE IMPLEMENTACIÓN	50
13.9.1	Para instalar el proyecto	50
13.9.2	Pruebas orgánicas de la aplicación.....	51
14.	DESCRIPCIÓN DE LA APLICACIÓN	51
14.1	MOVILIDAD A TRAVÉS DE LAS OPCIONES DE LA APLICACIÓN	52
14.2	FUNCIONAMIENTO MOBILE	52
14.2.1	Listar mensajes	52
14.2.2	Publicación de los mensajes:.....	54
14.2.3	Búsqueda de los mensajes:.....	55
14.3	FUNCIONAMIENTO WEB.....	57
14.4	HERRAMIENTA DE ADMINISTRACION	59
15.	CONCLUSIONES	62
14.5	CUMPLIMIENTO DE OBJETIVOS.....	62
14.6	DESDE LO QUE SE PLANTEÓ HASTA LO QUE SE DESARROLLÓ	63
14.7	PLANTEAMIENTO DE MEJORAS.....	63
14.8	DESARROLLOS FUTUROS.....	63
14.9	OPINIÓN PERSONAL	64
16.	FUENTES DE INFORMACIÓN.....	64

1. INTRODUCCIÓN

La iniciativa que persigue el proyecto *AnonyMSG* es dar movimiento a mensajes anónimos que se quedan en la web sin ningún fin, es decir, encontrar un espacio para dichos mensajes cuyo elemento común es el lugar en el que se han originado.

No existen usuarios ni login; como una pared de paso donde un transeúnte siente la necesidad de expresar algo y lo deja sin más, sin el peligro, por llamarlo de cierta forma, de ser visto o descubierto debido a la existencia de nics o identificadores que lo delaten.

AnonyMSG es una aplicación sin ánimo de lucro, y solo persigue quizá el fin un tanto literario de recoger mensajes que nunca llegan a ningún destino ya que no tienen ni remitente, ni destinatario cierto.

1. DEFINICIÓN GENERAL DEL PROYECTO

La idea es conseguir que existan páginas donde usuarios puedan dejar mensajes anónimos, los lugares no tienen necesariamente que existir, pueden ser lugares de nombre descriptivo. Sólo tiene como objetivo recopilar mensajes que se dejan anónimamente en un lugar, aprender HTML5 y JQueryMobile y vincularlo con indexadores para hacer las búsquedas más eficientes y útiles.

La aplicación no hará uso de login ni se registrarán usuarios.

El lugar seleccionado por el usuario para dejar su mensaje, podrá tener localización real del cual se guardará la ubicación geolocalizada, o podrá ser un sitio imaginario, que solo contará con el nombre. Dicho nombre, se podrá normalizar para poder asegurar más coincidencia de mensajes sobre un mismo lugar.

Ejemplo:

	Lugar registrado por el usuario	Valor normalizado	Geolocalización	Mensaje
1	En la Sagrada Familia	SAGRADA FAMILIA	Localización A	Mensaje para la chica de la falda rosa...
2	En el séptimo cielo de la Sagrada Familia	SAGRADA FAMILIA		Anónimo para el que cierra la puerta...
3	En el metro, línea 5	METRO LINEA 5		Mensaje para el conductor que pasa por Sagrada Familia a las 9:00 y nunca me espera...
4	Línea 5, viajando hacia Sagrada Familia	METRO LINEA 5, SAGRADA FAMILIA		Mensaje para ese turista que seguramente no entiende el castellano...

En la búsqueda correspondiente a la Sagrada familia, que aparece geolocalizado por el primer usuario, aparecerán los mensajes de los registros 1,2 y 4 con la localización del registro 1.

En la búsqueda correspondiente al Metro, Línea 5, aparecerán los registros 3 y 4.

1.1. OBJETIVOS

Dado el auge de los dispositivos móviles y la necesidad de ampliar hacia dicha línea las aplicaciones web existentes, surgen posibilidades como las aplicaciones multiplataforma con frameworks usados para encapsular una aplicación web y simulando una aplicación mobile.

Dentro de los frameworks existentes, se selecciona jQueryMobile framework, basado en el popular jQuery, y un magnífico aliado para el desarrollo de sitios web orientados para al segmento de consumidores web en movilidad.

El objetivo principal del proyecto es adquirir un conocimiento de las tecnologías implicadas en este tipo de desarrollos.

1.2. FUNCIONALIDADES PRINCIPALES

La aplicación consta básicamente de las siguientes pantallas que estarán presentes o no en dependencia del tamaño del dispositivo:

Tamaño del dispositivo	Pantallas	Descripción
Pequeño (Smartphones)	Inicio	Dividida en dos acciones: - Publicar mensajes - Buscar mensajes
Pequeño	Publicar mensajes	Diseñada para la publicación de los mensajes.
Pequeño	Buscar mensajes	Diseñada para la búsqueda de mensajes con una caja de texto abierta donde el usuario puede especificar cualquier palabra para su búsqueda.
Web y Tablet	Dashboard	Pantalla con tres widgets: - Publicar mensajes - Buscar mensajes - Conjunto de mensajes resultados de la búsqueda en caso de que se haya hecho alguna y en caso contrario, los últimos publicados.
Todos	Thickbox de Geolocalización	Pantalla de Google maps donde el usuario geolocaliza el lugar.

Dichas pantallas recogen las funciones que podrá hacer AnonyMSG:

- Generación y publicación de mensajes en tiempo real.
- Listado de los mensajes existentes por fecha de publicación, mostrando texto, hora, día y lugar.
- Geolocalización del lugar que el usuario quiera registrar.
- Búsqueda por indexación de mensajes dado un lugar o palabra en el texto del mismo.

2. CALENDARIO DEL PROYECTO

Según la programación de entregas propuesta en el curso, se muestra la siguiente tabla con los entregables ajustándolas al desarrollo del proyecto:

Fecha de entrega	Evaluación continuada	Entregable
11 de marzo	PAC1	Introducción, presentación y plan de trabajo.
8 de abril	PAC2	Análisis funcional, diseño técnico y prototipo.
20 de mayo	PAC3	Implementación
10 de junio	Entrega final	Memoria y video con la presentación del proyecto.

2.1. DETALLE DEL CALENDARIO DE TRABAJO

Dadas la fecha inicial y final del proyecto (28 de febrero al 10 de junio) se presenta el siguiente diagrama de Gantt con 102 días de planificación. Como la imagen se presenta muy reducida, se adjunta documento pdf.

PAC 3 IMPLEMENTACION ANONYMSG APP – Memoria

Nombre de la tarea	Fecha de inicio	Fecha de fin	Duración	Jan	Feb	Mar	Abr	May	Jun	Jul	Ag	Sep	Oct	Nov	Dic	Ene	Feb	Mar	Abr	May	Jun	Jul	Ag	Sep	Oct	Nov	Dic
1 PFC Anonymsg App	28/02/13	10/06/13	103																								
2 PAC 1 - Plan de trabajo	28/02/13	11/03/13	12																								
3 Búsqueda de información	28/02/13	03/03/13	4																								
4 Preparación de informe previo	04/03/13	06/03/13	2																								
5 Encuentro virtual Google Hangout	06/03/13	06/03/13	0																								
6 Preparación del Plan de Trabajo	06/03/13	10/03/13	5																								
7 Revisión y entrega del Plan de trabajo	11/03/13	11/03/13	1																								
8 PAC 2 - Análisis funcional y diseño técnico	12/03/13	08/04/13	26																								
9 Preparación del entorno de trabajo	12/03/13	13/03/13	2																								
10 Instalación y configuración del Software	12/03/13	13/03/13	2																								
11 Análisis funcional de la plataforma	14/03/13	18/03/13	5																								
12 Estado de requerimientos funcionales	14/03/13	18/03/13	3																								
13 Especificación de las funcionalidades	17/03/13	17/03/13	1																								
14 Generación de diagramas UML	18/03/13	18/03/13	1																								
15 Diseño funcional	18/03/13	31/03/13	13																								
16 Definición de la arquitectura del sistema	18/03/13	27/03/13	9																								
17 Preparación del diseño funcional	28/03/13	30/03/13	3																								
18 Generación de diagrama de BBDD	30/03/13	30/03/13	1																								
19 Generación de scripts de BBDD	30/03/13	31/03/13	2																								
20 Generación de Prototipos	01/04/13	08/04/13	8																								
21 Preparación de prototipos	01/04/13	02/04/13	2																								
22 Maquetación (Responsive Web Design)	02/04/13	08/04/13	7																								
23 Entrega del Análisis funcional y diseño técnico	08/04/13	08/04/13	1																								
24 PAC 3 - Implementación	08/04/13	21/04/13	13																								
25 Implementación de estructuras comunes	08/04/13	21/04/13	13																								
26 Creación de BBDD	08/04/13	10/04/13	2																								
27 Implementación de capa lógica y persistencia	11/04/13	13/04/13	3																								
28 Implementación de SaaS	14/04/13	21/04/13	8																								
29 Implementación de plataforma web	21/04/13	28/04/13	9																								
30 Implementación de plataforma móvil	30/04/13	09/05/13	10																								
31 Preparación de juegos de pruebas	10/05/13	11/05/13	2																								
32 Pruebas de integración	11/05/13	15/05/13	5																								
33 Preparación de juegos de pruebas de integración	11/05/13	13/05/13	3																								
34 Pruebas de integración web/móvil	13/05/13	15/05/13	3																								
35 Despliegue del proyecto	15/05/13	16/05/13	5																								
36 Estado de repositorio (dólar, cómo y cuánto)	15/05/13	17/05/13	3																								
37 Subida al repositorio	16/05/13	18/05/13	1																								
38 Pruebas críticas	18/05/13	19/05/13	2																								
39 Entrega de la implementación	20/05/13	20/05/13	1																								
40 PAC 4 - Preparación de la memoria	21/05/13	18/06/13	21																								
41 Preparación del documento formal	21/05/13	25/05/13	5																								
42 Preparación del material de la presentación	24/05/13	02/06/13	8																								
43 Grabación del vídeo	02/06/13	06/06/13	7																								
44 Entrega del material audiovisual	10/06/13	10/06/13	1																								

Las jornadas se quedan prefijadas como de 4 horas por día laborable y 8 horas por día festivo.

2.2. RECURSOS E INFRAESTRUCTURA

Se persigue diseño multiplataforma, con lo que se usará un framework para que la web se comporte como una aplicación mobile real. Nos basaremos en las plantillas del proyecto HTML5 Boilerplate (<http://html5boilerplate.com/>) para conseguir que un mismo diseño sea aplicado tanto para web como para dispositivos móviles.

De esta forma conseguiremos que accediendo a una misma url, nuestra aplicación sea visible con un diseño responsable en cualquier navegador.

También se desarrollará una web UI para Android, con la que podremos acceder al play store.

El objetivo fundamental del proyecto es incorporar elementos de Responsive Web Design para conseguir adaptar el sitio web al entorno del usuario intentando seguir el concepto de "One Web" o "Web for All" accediendo desde cualquier tipo de dispositivo.

2.2.1. RECURSOS HARDWARE

A continuación se detallan los elementos que intervendrán en el sistema desde el punto de vista técnico:

Componente	Características técnicas	Función
Ordenador de desarrollo	Intel® core(TM) Dúo CPU 1,86GHz, 2 GB de RAM	Tareas de diseño y desarrollo. Maquetación
Ordenador Servidor BBDD , Servidor Web privado y Repositorio	Intel® Dual Core, 4 GB de RAM	Hosting, BBDD y repositorio
Teléfono móvil	HTC 800 MHz, 384 RAM Conexión 3G /Wifi	Test y depuración de errores
Tablet	1GHz, 512 de RAM Conexión 3G /Wifi	Test y depuración de errores

2.2.2. RECURSOS SOFTWARE

Se detallan los elementos de software que, a grandes rasgos, se necesitarán durante el desarrollo del proyecto:

Software	Función
Maven	Herramienta de software para la gestión y construcción de proyectos JEE.
Eclipse Springsource	IDE para desarrollo especializado en Spring MVC.
Perforce	Software de repositorio.
MySQL	Software de gestión de BBDD.
Hibernate	Framework de persistencia.
Spring MVC	Framework para desarrollo de aplicaciones.
Contenedor Apache Tomcat	Servidor web con servicio para Servlets y JSPs + Servidor web

	HTTP.
Apache Solr	Software indexador para búsquedas.
Selenium	Test de integración.

3. RIESGOS DEL PROYECTO

Tendremos en cuenta los siguientes acontecimientos que pueden ser considerados incidencias e influirán en la consecución del resultado final. De esta forma intentamos reducir el impacto sobre el proyecto.

Riesgo	Probabilidad	Impacto	Acciones a llevar a cabo
Avería de HW	Baja	Crítico	<ul style="list-style-type: none"> - HW de contingencia - Copias de seguridad - Repositorio habilitado en perfectas condiciones.
Falta de conocimientos en Solr, Responsive Design.	Media	Media	<ul style="list-style-type: none"> - Búsqueda de documentación y manuales. - Tiempo extra para dichas actividades que se ha tenido en cuenta en la planificación del diagrama de Gantt. - Sustitución de dichas tareas por otras con menos necesidad de conocimientos repercutiendo en la calidad del producto pero garantizando su entrega.
Falta de conocimientos en JQueryMobile y HTML5.	Media	Alta	<ul style="list-style-type: none"> - Búsqueda de documentación y manuales. - Tiempo extra para dichas actividades que se ha tenido en cuenta en la planificación del diagrama de Gantt.
Planificación del proyecto incorrecta.	Media	Media	<ul style="list-style-type: none"> - Seguimiento estricto del calendario. - Reprogramación de las tareas. - Replanteamiento de las soluciones.
Enfermedad del desarrollador.	Inevitable	Media	<ul style="list-style-type: none"> - Seguimiento propio del caso según proceda.
Impacto de otras asignaturas y trabajo no tenido en cuenta durante la planificación.	Inevitable	Media	<ul style="list-style-type: none"> - Seguimiento estricto del calendario.

4. REQUERIMIENTOS FUNCIONALES

A continuación se detallan las diferentes funcionalidades que se esperan desarrollar para conformar AnonyMSG como un sencillo sistema de mensajería con remitente anónimo y lugar especificado.

4.1 FUNCIONES PARA USUARIOS EN DISPOSITIVOS WEB O MOBILE

- 1- Listar mensajes publicados: Acceso al listado de todos los mensajes publicados ordenados por fecha y hora con lugar especificado.
- 2- Búsqueda de mensajes publicados por lugar: Búsqueda de mensajes, dados varios parámetros de búsqueda (lugar, hora, texto en particular o fecha) obteniéndose como resultado una lista de los mensajes que coincidan con los criterios de búsqueda.
- 3- Búsqueda de mensajes por geolocalización: Dado un mapa como google maps, aparecerán iconos que indiquen lugares con mensajes de AnonyMSG publicados.
- 4- Publicar un mensaje: El usuario, sin necesidad de login o identificación, podrá publicar un mensaje especificando un lugar. Dicho lugar podrá estar geolocalizado o ser un lugar imaginario.
- 5- Dar un voto a un mensaje publicado: Se podrá dar un voto de 'Me gusta' o 'No me gusta' a cada mensaje publicado. Solo se podrá efectuar un voto sobre un mismo mensaje una vez por cada sesión, ya que la sesión del usuario es el único identificador que tendremos.

4.2 FUNCIONES PARA USUARIOS ADMINISTRADORES

- 1- Login del administrador: Dado usuario y password, acceso a la herramienta de administración del aplicativo.
- 2- Listar mensajes publicados: Lista de mensajes publicados.
- 3- Conocer la existencia de mensajes que presenten contenido ofensivo dada una lista de stop words: Al publicar un mensaje, se revisarán con patrones campo título y cuerpo del mensaje para conocer la existencia de mensajes con contenido ofensivo o poco acorde con la ética del aplicativo.
- 4- Administrar lista de stop words: Posibilidad de acceso a lista de stop words y agregar o eliminar palabras de dicha lista.
- 5- Eliminar un mensaje de un lugar: Posibilidad de eliminar un mensaje del aplicativo.
- 6- Administrar lugares con mensajes: Acceso a lista de todos los lugares dados de alta el aplicativo y posibilidad de modificar o quitar dichos lugares.

5. REQUERIMIENTOS NO FUNCIONALES

Destacamos algunas características importantes a tener en cuenta como parte de los requisitos:

- Usabilidad
- Agilidad en las búsquedas
- Accesibilidad

5.1 USABILIDAD

Se persigue un diseño lo más usable y óptimo posible. El contenido a mostrar será sencillo, ya que es una aplicación que básicamente se usa para publicar mensajes. En la pantalla no aparecerá contenido que no se necesite para la funcionalidad en cuestión.

La navegación será muy sencilla, la versión mobile de la aplicación, solo cuenta con tres pantallas: publicación, búsqueda y geolocalización.

Hay que tener en cuenta tanto las pantallas en orientación horizontal como vertical y tratar de mantener a los usuarios en la misma sección de la página al cambiar la orientación.

Los botones han de ser grandes, cuadrados y con simulación de volumen. Los links y los checks han de tener el área de respuesta ampliada.

La maquetación se ha de hacer para diferentes dispositivos.

La versión web y la versión móvil tienen que ser similares y ofrecer la misma experiencia.

Es absolutamente esencial detectar si un usuario accede a la web desde móvil y redireccionarle a la versión del sitio diseñada para móviles. Pero una buena práctica hay que ofrecer la opción de volver al sitio para web y al mismo tiempo facilitar el regreso al sitio para mobile. Los usuarios podrán elegir qué versión de la aplicación quieren usar.

5.2 AGILIDAD EN LAS BUSQUEDAS

Las búsquedas han de ser lo más rápidas posibles, y no causar problemas de rendimiento en el motor de búsquedas de MySQL. Se han de poder realizar búsquedas geolocalizadas, es decir, dado un mapa, poder saber en qué lugares hay mensajes publicados.

5.3 ACCESIBILIDAD

La aplicación deberá ser accesible vía web y mobile y para ello, se instalará en un servidor web al que se podrá acceder a través de un servidor web con una dirección provisional tipo anonymsg.dyndns.com.

6. FUNCIONALIDADES Y USUARIOS DEL SISTEMA

6.1 USUARIOS DEL SISTEMA

Usuarios identificados: Usuarios identificados por el número de sesión. En cuanto se desconectan de la aplicación, ya sea web o mobile, desaparecen del sistema. El mensaje sí se quedará identificado por el número de la sesión, solamente podrán ser dueños de los mensajes que crean durante esa sesión, una vez abandonada la misma, la sesión se destruye y ya no podrán eliminar sus mensajes.

Usuarios administradores: Usuarios que hacen login en la herramienta para su administración.

6.2 FUNCIONALIDADES DE USUARIOS

Todas estas funcionalidades se desarrollarán para ambas plataformas (Web y Mobile) procurando que las mismas se comporten de forma similar.

- 1- Iniciar sesión
- 2- Listar mensajes ordenados por fechas
- 3- Listar mensajes ordenados por fecha dado un lugar o texto seleccionado
- 4- Listar mensajes por geolocalización
- 5- Ordenar búsquedas
- 6- Publicar un mensaje
- 7- Dar o quitar voto de un mensaje
- 8- Finalizar sesión

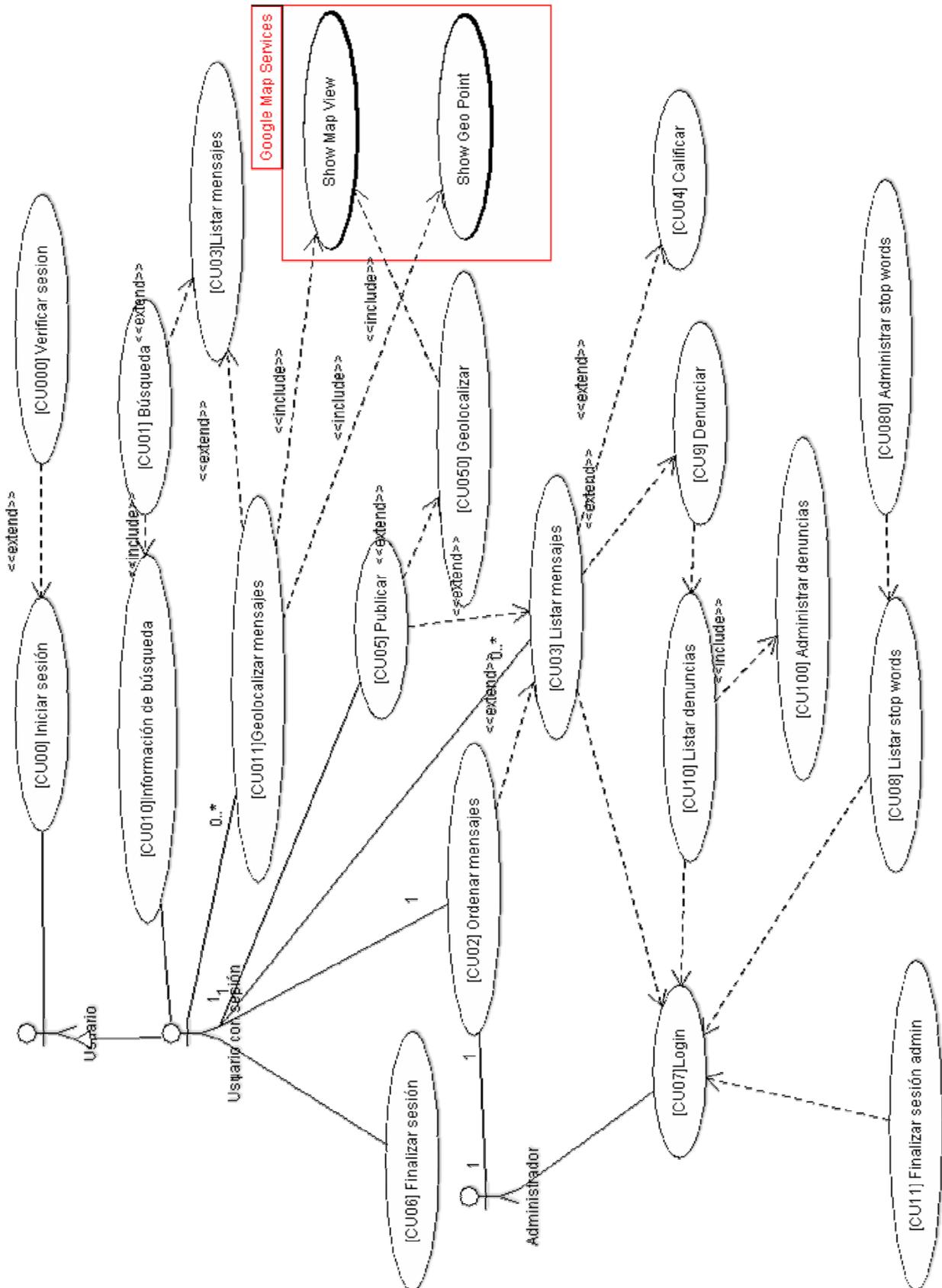
6.3 FUNCIONALIDADES DE USUARIOS ADMINISTRADORES

Todas estas funcionalidades son desarrolladas para la aplicación administrativa, que será únicamente una aplicación Web:

- 1 - Login
- 2 - Listar todo el contenido de mensajes
- 3 - Recibir alerta contenido ofensivo
- 4 - Listar stop words
- 5 - Agregar/modificar/eliminar stop words
- 6 - Eliminar mensajes
- 7 - Eliminar lugares y mensajes asociados
- 8 – Finalizar sesión

7. CASOS DE USO

Presentamos un diagrama de los casos de uso para ambos tipos de usuario y una descripción más detallada de los mismos a continuación.



7.1 DESCRIPCIÓN DE CASO DE USO [CU00] – INICIAR SESIÓN

Identificador	CU00
Nombre	Iniciar sesión
Resumen	Representa el inicio de sesión de un usuario. El sistema reconocerá de donde viene la petición, si web o mobile y ofrecerá en ambos casos la posibilidad de acceso a cualquiera de las dos plataformas.
Actores	Usuario
Precondiciones	Ninguna
Postcondiciones	El usuario queda identificado con un identificador de sesión que le acompañará durante toda la misma.
Flujo normal	<ol style="list-style-type: none"> 1. El usuario hace una petición http a la url de la aplicación o el usuario accede a su aplicación ya descargada previamente. 2. El sistema le da la bienvenida ofreciendo la posibilidad de un cambio de plataforma si es que así el usuario lo desea. 3. Se muestra la home de la aplicación.
Flujos alternativos	1a) En caso de error en el reconocimiento del dispositivo con el que se está intentando conectar el usuario, se ofrecerá la plataforma web con un mensaje que lo informe.
Inclusiones	Ninguna
Extensiones	[CU000] Verificar sesión

7.2 DESCRIPCIÓN DE CASO DE USO [CU000] – VERIFICAR SESIÓN

Identificador	CU000
Nombre	Verificar sesión
Resumen	Se busca la existencia de un identificador de sesión en el objeto usuario y si no lo tiene, se le añade uno.
Actores	Usuario
Precondiciones	Usuario sin identificador de sesión inicia la sesión.
Postcondiciones	Usuario con identificador de sesión
Flujo normal	1. El usuario accede a la aplicación a través de la url en alguna de las dos plataformas.
Flujos alternativos	1a) En caso de que ocurra un error de internet o de registro en BBDD del identificador de sesión, se le informará al usuario con un mensaje que finaliza el caso de uso.
Inclusiones	Ninguna
Extensiones	Ninguna

7.3 DESCRIPCIÓN DE CASO DE USO [CU01] –BÚSQUEDA

Identificador	CU01
Nombre	Búsqueda
Resumen	El usuario ejecuta una búsqueda dados unos criterios de su interés obteniendo un listado de los mensajes que cumplen con dichos criterios.
Actores	Usuario con sesión
Precondiciones	Usuario con sesión establece criterios para realizar la búsqueda

Postcondiciones	El usuario obtiene un listado de mensajes
Flujo normal	1. El caso comienza cuando el usuario pulsa en el botón buscar de la home.
Flujos alternativos	1ª) No hay ningún mensaje que cumpla con los criterios y se le informará al usuario con un mensaje.
Inclusiones	[CU010] Información de búsqueda
Extensiones	[CU02] Ordenar mensajes

7.4 DESCRIPCIÓN DE CASO DE USO [CU010] – INFORMACIÓN DE BÚSQUEDA

Identificador	CU010
Nombre	Información de búsqueda
Resumen	Dado un formulario el usuario rellena los datos correspondientes con su búsqueda.
Actores	Usuario con sesión
Precondiciones	Usuario con sesión.
Postcondiciones	El usuario tendrá acceso a realizar una búsqueda.
Flujo normal	<ol style="list-style-type: none"> 1. El caso comienza con la presentación de los campos posibles para generar una búsqueda. 2. El usuario rellenará los campos que desee
Flujos alternativos	<p>1ª) En caso de rellenar algún campo mal, aparecerá un mensaje de error en la pantalla indicando cuál es el campo erróneo para dar la opción de corrección.</p> <p>1b) El usuario puede decidir no rellenar ningún campo de criterio y pulsar en botón de buscar, en ese caso se listarán todos los mensajes del sistema ordenados por fecha de publicación pasando al caso CU03 Listar mensajes.</p>
Inclusiones	Ninguna
Extensiones	Ninguna

7.5 DESCRIPCIÓN DE CASO DE USO [CU011] – GEOLOCALIZAR MENSAJES

Identificador	CU011
Nombre	Geolocalizar mensajes
Resumen	El usuario puede establecer la localización física de su mensaje a través de un mapa de google.
Actores	Usuario con sesión
Precondiciones	Existe posibilidad de geolocalización
Postcondiciones	Aparecerá un listado de mensajes correspondientes con la geolocalización establecida ordenados por fecha de aparición.
Flujo normal	<ol style="list-style-type: none"> 1- El usuario pulsa el icono de acceso a los mapas. 2- Aparecerá una ventana de mapas. 3- Aparecerá por defecto la zona en la que se encuentra ubicado el dispositivo físicamente. 4- Se mostrarán puntos en el mapa que indican los lugares donde hay publicados mensajes de la aplicación. 5- El usuario selecciona uno de los puntos en los que aparecen mensajes para establecerlo como geolocalización de su búsqueda. 6- El usuario también puede escribir qué geolocalización quiere.

Flujos alternativos	1a) Hay problemas en el acceso a los mapas, el botón de acceso a la página de geolocalización aparecerá inhabilitado y se mostrará un mensaje de información que lo indique. 3a) No se puede ubicar físicamente el dispositivo. Aparecerá una localización por defecto.
Inclusiones	Show Map View (de Google Map Services) Show Geo Point (de Google Map Services)
Extensiones	[CU03] Listar mensajes

7.6 DESCRIPCIÓN DE CASO DE USO [CU02] – ORDENAR MENSAJES ENCONTRADOS

Identificador	CU02
Nombre	Ordenar mensajes
Resumen	Se le da al usuario la posibilidad de ordenar mensajes de un listado.
Actores	Usuario con sesión Usuario administrador
Precondiciones	Existe en pantalla un listado de mensajes.
Postcondiciones	El listado aparece ordenado según el criterio de ordenación seleccionado por el usuario. Siempre existirá posibilidad de ordenación por cualquier criterio.
Flujo normal	<ol style="list-style-type: none"> 1. El caso empieza desde el listado de mensajes ([CU03]) 2. En el encabezado del listado, aparecerán unos iconos de flechas que orientan la ordenación y el caso de uso comienza cuando el usuario pulsa alguno de estos iconos. 3. Cuando ya los mensajes están ordenados, el icono correspondiente con dicha ordenación, aparecerá inhabilitado. 4. Existirá siempre la opción de volver al listado original que da por cerrado el caso de uso.
Flujos alternativos	1a) Si ocurre algún error de internet o de bbdd, se le informará con un mensaje al usuario.
Inclusiones	Ninguna
Extensiones	[CU03] Listar mensajes

7.7 DESCRIPCIÓN DE CASO DE USO [CU03] – LISTAR MENSAJES

Identificador	CU03
Nombre	Listar mensajes
Resumen	El usuario podrá acceder a una lista de mensajes con la descripción de los mismos.
Actores	Usuario con sesión Usuario administrador
Precondiciones	Usuario con sesión.
Postcondiciones	Aparece un listado de mensajes con sus detalles y sus calificaciones.
Flujo normal	<ol style="list-style-type: none"> 1. El caso de uso comienza cuando el usuario accede a la aplicación. 2. A este caso de uso también se puede llegar a través de una búsqueda [CU01] normal o por geolocalización [CU011]. 3. Aparecerán los mensajes del día de hoy en primer lugar y a continuación el resto de mensajes ordenados por fecha.
Flujos alternativos	1a) No aparecen mensajes en la aplicación, se le indicará al usuario con un aviso.
Inclusiones	Ninguna
Extensiones	[CU04] Calificar

7.8 DESCRIPCIÓN DE CASO DE USO [CU04] – CALIFICAR

Identificador	CU04
Nombre	Calificar
Resumen	Se podrá dar o quitar puntuación a un mensaje
Actores	Usuario con sesión
Precondiciones	El usuario ha iniciado un proceso de listado de mensajes
Postcondiciones	En el listado de los mensajes se verá reflejada la puntuación actual del usuario.
Flujo normal	<ol style="list-style-type: none"> 1. Comienza con el listado de mensajes 2. Aparecerán dos iconos, uno para dar un punto y otro para quitar un punto. El usuario podrá pulsar cualquiera de los dos botones. 3. El tile de los iconos avisará al usuario que sólo se votará una vez por sesión. 4. El botón pulsado se inhabilitará para impedir que ese usuario en esa sesión vuelva a calificar un mismo mensaje.
Flujos alternativos	2a) En caso de que exista error de internet o de BBDD, se le mostrará un mensaje al usuario en pantalla invitándole a calificar en otro momento.
Inclusiones	Ninguna
Extensiones	Ninguna

7.9 DESCRIPCIÓN DE CASO DE USO [CU05] – PUBLICAR

Identificador	CU05
Nombre	Publicar
Resumen	El usuario podrá publicar mensajes una vez rellenado el formulario para dicha publicación.
Actores	Usuario con sesión
Precondiciones	Rellenar formulario de datos de publicación correctamente.
Postcondiciones	El mensaje publicado, se agregará al listado de mensajes de la home con su hora de publicación.
Flujo normal	<ol style="list-style-type: none"> 1. El usuario con sesión accede a la home. 2. Rellena el formulario con los datos necesarios 3. Pulsa el botón de publicar 4. Aparecerá el mensaje publicado en el listado.
Flujos alternativos	<p>2a) El usuario puede optar por la opción de Geolocalizar [CU050].</p> <p>2b) En caso de que algún dato presente algún error, se le mostrará un mensaje al usuario con la posibilidad de corrección.</p> <p>3a) En caso de que ocurra algún error de internet o de BBDD, se le mostrará un mensaje al usuario orientándole la posibilidad de volver a publicar.</p> <p>3b) En caso de que el texto del mensaje presente una Stop word, también se le comunicará al usuario y se le imposibilitará publicar el mensaje hasta que no lo cambie.</p>
Inclusiones	[CU050] Geolocalizar
Extensiones	[CU03] Listar mensajes

7.10 DESCRIPCIÓN DE CASO DE USO [CU050] – GEOLOCALIZAR

Identificador	CU050
Nombre	Geolocalizar
Resumen	El usuario puede establecer una ubicación en google map para su lugar del mensaje como un dato más.
Actores	Usuario con sesión
Precondiciones	Posibilidad técnica de geolocalización. Haber iniciado un caso de publicación de mensaje (CU05).
Postcondiciones	El mensaje aparecerá con un acceso en el listado a su geolocalización una vez publicado el mensaje.
Flujo normal	<ol style="list-style-type: none"> 1 – El usuario está en la ventana de publicación de mensaje y pulsa en el acceso a Geolocalizar. 2- Ubicará el lugar ya sea tecleando el nombre en el buscador o ubicándolo con un click en el mapa y pulsando ok a continuación. 3- Estos mensajes aparecerán en el listado con un link al mapa para ubicarlos y este es el fin de este caso de uso.
Flujos alternativos	2a) En caso de error producto del acceso a la herramienta de mapas, aparecerá un mensaje que lo informa.
Inclusiones	Show Map View (Google Map Service)
Extensiones	Ninguna

7.11 DESCRIPCIÓN DE CASO DE USO [CU06] – FINALIZAR SESIÓN

Identificador	CU06
Nombre	Finalizar sesión
Resumen	El usuario sale de la sesión destruyéndose su identificador temporal y su posibilidad de modificación o eliminación de mensajes publicados.
Actores	Usuario con sesión
Precondiciones	El usuario está conectado a la aplicación.
Postcondiciones	El usuario cierra la sesión pulsando en el cierre del navegador o pulsando en Salir.
Flujo normal	<ol style="list-style-type: none"> 1.El usuario pulsa el botón Salir. 2.Aparecerá, antes de salir, un mensaje aclaratorio de su pérdida de posibilidad de modificación o eliminación de sus mensajes y preguntando si realmente desea salir. 3.En caso afirmativo, se da por cerrada la sesión y el caso de uso.
Flujos alternativos	2a) En caso de que el usuario seleccione su deseo de permanencia en el sistema, se interrumpe el caso de uso.
Inclusiones	Ninguna
Extensiones	Ninguna

7.12 DESCRIPCIÓN DE CASO DE USO [CU07] – LOGIN

Identificador	CU07
Nombre	Login
Resumen	El usuario administrador hace entrada de usuario y contraseña en el sistema.
Actores	Usuario administrador
Precondiciones	Usuario con login y password del sistema
Postcondiciones	Usuario logado y con acceso a todas las funciones de administrador.
Flujo normal	<ol style="list-style-type: none"> 1- El usuario pone su usuario y su password. 2- El usuario hace click en el botón Entrar. 3- Si está todo bien, se le da acceso a la ventana con las funciones correspondientes. 4- Se le da sesión de administrador.
Flujos alternativos	3a) Usuario y/o password incorrectos, se le comunica al usuario y se le deja en la ventana de login.
Inclusiones	Ninguna
Extensiones	Ninguna

7.13 DESCRIPCIÓN DE CASO DE USO [CU08] – LISTAR STOP WORDS

Identificador	CU08
Nombre	Listar stop words
Resumen	El usuario administrador puede listar palabras no permitidas para administrarlas.
Actores	Usuario administrador
Precondiciones	Usuario administrador logado
Postcondiciones	Aparecerá una lista de stop words con un botón al lado para eliminar o modificar ordenadas alfabéticamente.

Flujo normal	<ol style="list-style-type: none"> 1- El usuario administrador ha pulsado en la opción de Administración de stop words. 2- Aparecerá un listado de las mismas con posibilidad para modificar o eliminar.
Flujos alternativos	1a) En caso de que el listado aparezca vacío, se le comunicará al usuario con un mensaje.
Inclusiones	Ninguna
Extensiones	[CU080] Agregar/Eliminar stop words

7.14 DESCRIPCIÓN DE CASO DE USO [CU080] – ADMINISTRAR STOP WORDS/

Identificador	CU080
Nombre	Administrar stop words
Resumen	A partir de un listado de stop words (CU08), el usuario administrador puede agregar, eliminar o modificar palabras no permitidas.
Actores	Usuario administrador
Precondiciones	Usuario administrador logado.
Postcondiciones	El listado quedará modificado según la acción seleccionada por el usuario.
Flujo normal	<ol style="list-style-type: none"> 1- El listado presenta la posibilidad de agregar palabras ya que la primera fila es una fila en blanco con un botón Agregar. 2- El listado presenta en cada registro, un botón Eliminar que dará la posibilidad de Eliminar la palabra. 3- Todas las palabras presentadas son editables. 4- Una vez modificado el listado según conveniencia, se pulsará Ok. 5- El sistema actualizará los mensajes según la lista actual de stop words eliminando los mensajes que contentan dichas palabras e informando de los mensajes que han sido eliminados.
Flujos alternativos	4a) En caso de presentar algún error en la modificación del listado, se le mostrará un mensaje con el mismo al usuario.
Inclusiones	Ninguna
Extensiones	Ninguna

7.15 DESCRIPCIÓN DE CASO DE USO [CU09] – DENUNCIAR

Identificador	CU09
Nombre	Denunciar
Resumen	El usuario puede denunciar un mensaje por contenido que considere ofensivo.
Actores	Usuario.
Precondiciones	El usuario ha iniciado el caso de uso de la lista de mensajes [CU03] y pulsa en el icono Denunciar.
Postcondiciones	El mensaje queda marcado con la denuncia y se incorporará a la lista de mensajes denunciados para ser revisado por el administrador del sistema.
Flujo normal	<ol style="list-style-type: none"> 1- El usuario accede a la lista de mensajes. 2- Pulsa en el icono Denunciar 3- Aparece un texto informándole de que el mensaje será revisado por el administrador del sistema y que se tomarán

	las medidas pertinentes. 4- El mensaje aparecerá en el listado como 'Denunciado'.
Flujos alternativos	2a) En caso de error de internet o de BBDD, se le informará al usuario del mismo.
Inclusiones	Ninguna
Extensiones	Ninguna

7.16 DESCRIPCIÓN DE CASO DE USO [CU10] – LISTAR DENUNCIAS

Identificador	CU10
Nombre	Listar denuncias
Resumen	Se le da la posibilidad al usuario de listar todos los mensajes con denuncias y los detalles de los mismos.
Actores	Usuario administrador
Precondiciones	Usuario administrador logado
Postcondiciones	Aparece un listado con todos los detalles de los mensajes denunciados.
Flujo normal	<ol style="list-style-type: none"> 1- El caso comienza cuando el usuario administrador pulsa la opción de su menú de Administrar denuncias. 2- Aparecerá un listado de mensajes denunciados con la posibilidad de eliminar los mismos.
Flujos alternativos	1a) No existen mensajes denunciados en el sistema y se le informa al usuario.
Inclusiones	[CU100] Administrar denuncias
Extensiones	Ninguna

7.17 DESCRIPCIÓN DE CASO DE USO [CU100] – ADMINISTRAR DENUNCIAS

Identificador	CU100
Nombre	Administrar denuncias
Resumen	Posibilidad que tiene el administrador del sistema de eliminar los mensajes denunciados por los usuarios.
Actores	Usuario administrador
Precondiciones	Usuario administrador logado
Postcondiciones	Mensaje eliminado del sistema
Flujo normal	<ol style="list-style-type: none"> 1- A partir del listado de mensajes denunciados [CU10] se pulsa en Eliminar Mensaje denunciado. 2- El mensaje se eliminará del sistema automáticamente.
Flujos alternativos	1a) Aparece un mensaje de error en caso de que exista algún problema de Internet o BBDD.
Inclusiones	Ninguna
Extensiones	Ninguna

7.18 DESCRIPCIÓN DE CASO DE USO [CU11] – FINALIZAR SESIÓN ADMIN

Identificador	CU11
Nombre	Finalizar sesión admin
Resumen	El usuario administrador se desloga del sistema
Actores	Usuario administrador
Precondiciones	Usuario administrador logado
Postcondiciones	Usuario administrador con sesión cerrada
Flujo normal	El usuario administrador hace click en Salir de la herramienta de administración.
Flujos alternativos	Ninguno
Inclusiones	Ninguna
Extensiones	Ninguna

8. DISEÑO TÉCNICO

La arquitectura general del proyecto que se va a desarrollar, está basada en una arquitectura Cliente-Servidor de tres capas:

- Capa de presentación
- Capa de negocio
- Capa de acceso a datos

8.1 PARTE SERVIDOR

La parte de servidor estará comprendida por la capa de gestión y la capa de persistencia. Atenderá las peticiones de todos los clientes que podrán venir desde diferentes plataformas (web o mobile).

Como **servidor web**, utilizaremos un contenedor Apache Tomcat. Los contenedores proveen un entorno de ejecución para componentes Java y brindan los servicios de seguridad, transacciones, administración del ciclo de vida, caches, persistencia y comunicación en la red. Apache se encargará de servir las páginas estáticas y Tomcat las páginas dinámicas. Nos podremos servir además de los otros módulos de apache como mod_rewrite y mod_authz.

Como **servidor de datos**, utilizaremos un MySQL server. MySQL es una [base de datos](#) muy rápida en la lectura cuando utiliza el motor no transaccional [MyISAM](#), pero puede provocar problemas de integridad en entornos de alta concurrencia en la modificación. En aplicaciones web hay baja concurrencia en la modificación de datos y en cambio el entorno es intensivo en lectura de datos, lo que hace a MySQL ideal para este tipo de aplicaciones.

Como **servidor de búsquedas**, utilizaremos Apache solr, que es un servidor de búsqueda construida sobre la librería Lucene que nos permite realizar peticiones HTTP para indexar o hacer consultas sobre la indexación de documentos, base de datos, etc.

8.2 PARTE CLIENTE

La parte que inicia el proceso en el modelo, es la parte cliente enviando una solicitud de servicio a nuestra aplicación, el proceso ocurre de la siguiente manera:

- El cliente solicita una petición vía URL.
- El servidor recibe la petición.
- Si es la primera, se usa el motor de Servlets para cargarlo y llamar al init()
- Si no es la primera, se convierte en un hilo.

- Un servlet puede manejar múltiples peticiones.
- El cliente, una vez lanzada la petición se queda a la espera de la respuesta del servidor.

Nuestro cliente lanzará peticiones tanto al servidor AnonyMSG como a Google Maps para toda la parte de geolocalización.

8.3 DISEÑO LÓGICO

El diseño lógico va a estar conformado por un conjunto de frameworks, APIs y diferentes metodologías de desarrollo.

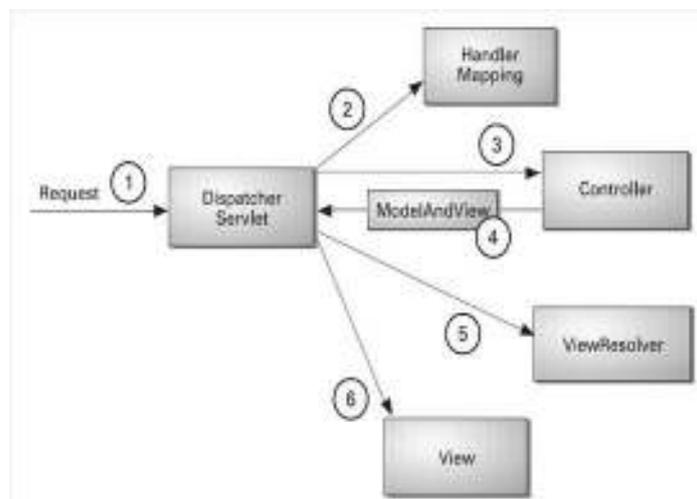
A continuación haremos una breve reseña de las metodologías de desarrollo a utilizar **a medida que el proyecto se vaya desarrollando**:

Maven:

Emplearemos Maven para montar la estructura del proyecto, descargar librerías, compilar y generar informes, es decir, utilizaremos Maven para gestionar nuestro proyecto y hacerlo reutilizable.

Spring MVC:

Proyecto de Springsource que nos permite crear una aplicación con patrón Modelo Vista Controlador.



Hibernate:

Herramienta de [Mapeo objeto-relacional](#) (ORM) para la [plataforma Java](#) que facilita el mapeo de atributos entre una [base de datos](#) relacional tradicional y el modelo de [objetos](#) de una aplicación, mediante archivos declarativos ([XML](#)) o anotaciones en los [beans](#) de las entidades que permiten establecer estas relaciones.

Gracias a este framework, solucionaremos las diferencias entre el modelo de datos y el modelo de clases.

Apache Solr:

Solr es un motor de búsqueda de [código abierto](#) basado en la [biblioteca Java](#) del proyecto [Lucene](#), con APIs en [XML/HTTP](#) y [JSON](#), resaltado de resultados, [búsqueda facetada](#), [caché](#), y una interfaz para su administración.¹

Corre sobre un contenedor de [servlets](#) Java como [Apache Tomcat](#).

Geolocalización:

Es una nueva API [Javascript](#) que nos permite conocer de forma fácil nuestras coordenadas desde un navegador, de esta forma tendremos almacenados **la latitud y longitud de una posición en concreto**.

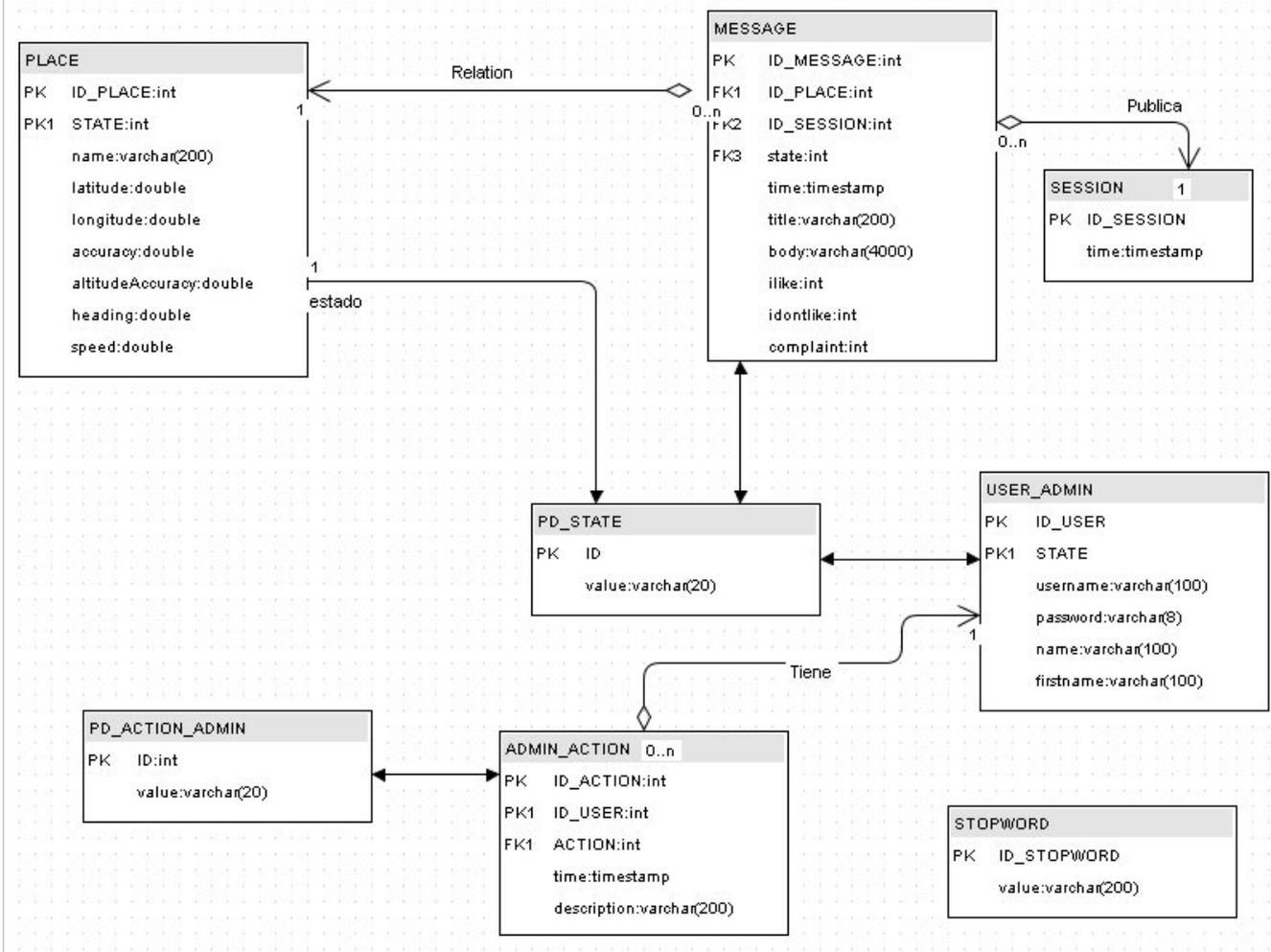
Responsive Design:

Framework basado en el ya conocido framework de jquery. Conjunto de **herramientas que simplificará el proceso de crear páginas para móviles**, desde la escritura del propio código HTML, la maquetación con CSS y la creación de efectos dinámicos con Javascript.

9. ARQUITECTURA DE BASE DE DATOS

Definimos a continuación la estructura del modelo de persistencia de datos. El mismo ha de mantener la información de usuarios administradores, mensajes, lugares y geolocalizaciones.

9.1 DESCRIPCIÓN DE TABLAS



9.2 DESCRIPCIÓN DE ENTIDADES MÁS IMPORTANTES

A continuación describimos brevemente cada una de las tablas y su relación con las demás entidades:

ENTIDAD LÓGICA	DESCRIPCIÓN	CARDINALIDAD	ENTIDAD RELACIONADA
PD_STATE	Estados a tener en cuenta con relación a mensajes y usuarios (activos, borrados, etc.).	1:1	USER_ADMIN
		1:1	MESSAGE
		1:1	PLACE
PD_ACTION_ADMIN	Valores de las posibles acciones que puede hacer un administrador para dejar histórico (eliminar mensaje, eliminar lugar, etc.).	1:1	ADMIN_ACTION
PLACE	Mantiene la persistencia de los lugares.	1:N	MESSAGE
MESSAGE	Mantiene la persistencia de los mensajes.		
SESSION	Mantiene la persistencia de las sesiones.	1:N	MESSAGE

USER_ADMIN	Mantiene la persistencia de los usuarios administradores.	1:N	ADMIN_ACTION
ADMIN_ACTION	Mantiene el histórico de las acciones llevadas a cabo por los administradores.	N:1	USER_ADMIN
STOPWORD	Mantiene la persistencia de la lista de stop words que verifican la calidad de los mensajes publicados.		

A continuación describimos las tablas más importantes con los detalles de sus atributos.

9.2.1 Entidad PLACE

ATRIBUTO	DESCRIPCIÓN	TIPO	LONGITUD	CLAVE
ID_PLACE	Identificador del lugar	Autonumérico		SI
STATE	Estado del lugar (activo, borrado, inactivo)	Dígito	1	SI
name	Nombre en texto del lugar	Texto	200	NO
latitude	Dato para geolocalización	Decimal	-	NO
longitude	Dato para geolocalización	Decimal	-	NO
accuracy	Dato para geolocalización	Decimal	-	NO
altitudeAccuracy	Dato para geolocalización	Decimal	-	NO
heading	Dato para geolocalización	Decimal	-	NO
speed	Dato para geolocalización	Decimal	-	NO

9.2.2 Entidad MESSAGE

ATRIBUTO	DESCRIPCIÓN	TIPO	LONGITUD	CLAVE
ID_MESSAGE	Identificador del mensaje	Autonumérico	-	SI
ID_PLACE	Identificador del lugar	Numérico	-	SI
ID_SESSION	Identificador de la sesión, nutable	Texto	200	SI
STATE	Identificador del estado (activo, borrado, inactivo)	Dígito	1	SI
time	Fecha y hora en la que se publica	Fecha	-	NO
title	Título del mensaje	Texto	200	NO
body	Cuerpo del mensaje	Texto	4000	NO
ilike	Número de calificaciones positivas	Numérico	-	NO
idontlike	Número de calificaciones negativas	Numérico	-	NO
complaint	Número de denuncias del mensaje	Numérico	-	NO

9.2.3 Entidad USER_ADMIN

ATRIBUTO	DESCRIPCIÓN	TIPO	LONGITUD	CLAVE
ID_USER	Identificador de usuario	Autonumérico	-	SI
STATE	Estado en el que se encuentra (activo, borrado, inactivo)	Dígito	1	SI
username	Nombre de usuario, único	Texto	100	NO
password	Contraseña	Texto	8	NO

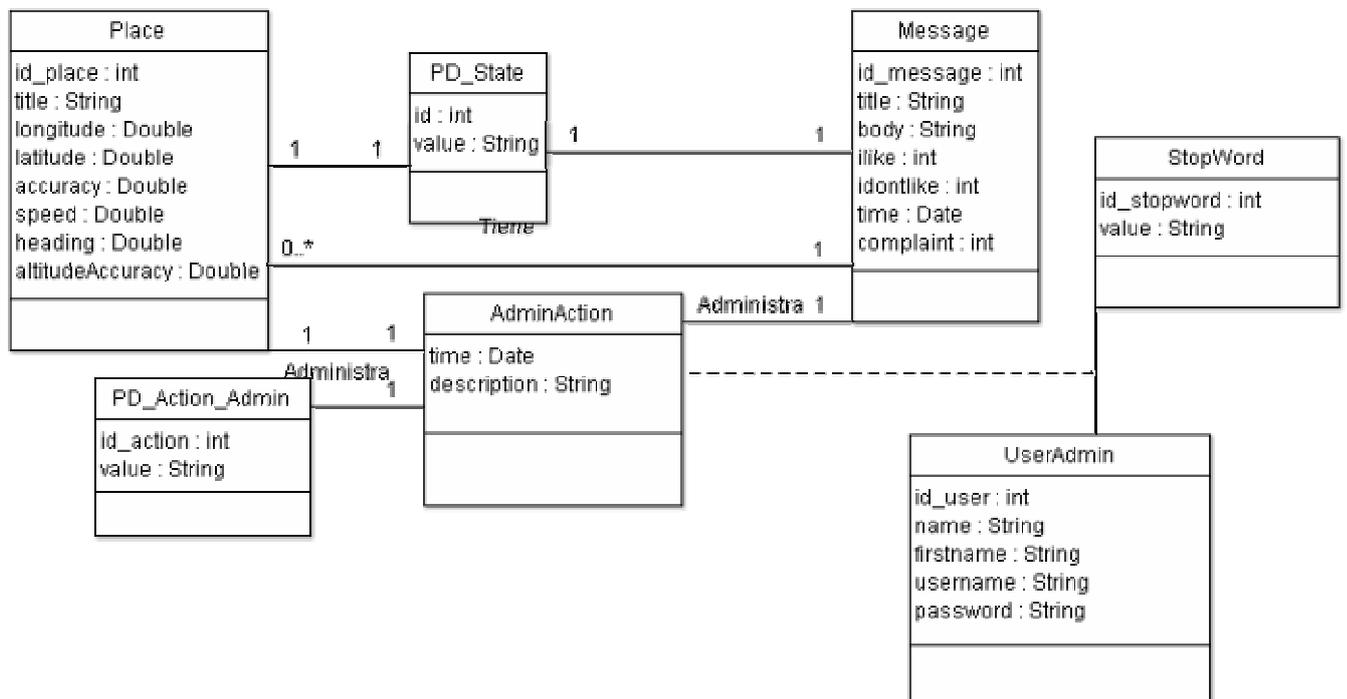
name	Nombre del administrador	Texto	100	NO
firstname	Apellido del administrador	Texto	100	NO

9.2.4 Entidad ADMIN_ACTION

ATRIBUTO	DESCRIPCIÓN	TIPO	LONGITUD	CLAVE
ID_ACTION	Identificador de la acción.	Autonumérico	-	SI
ID_USER	Identificador del usuario administrador	Numérico	-	SI
ACTION	Tipo de acción (Borrar mensaje, borrar lugar, etc.).	Numérico	1	SI
time	Fecha y hora de la acción.	Fecha	-	NO
description	Descripción de la misma.	Texto	200	NO

10. DIAGRAMA DE CLASES

El diagrama de clases se encuentra muy relacionado con el diagrama de entidades que hemos mostrado en el apartado anterior.



10.1 DESCRIPCIÓN DE LOS SERVICIOS Y RELACIÓN CON LAS CLASES

Los servicios son elementos importantes a la hora de realizar el diagrama de clases, ya que a partir de ellos podremos identificar previamente las clases que necesitaremos para implementar el diseño vista controlador con Spring MVC. Los servicios que se prevén son los siguientes:

- Publicar (PublicService)
- Búsqueda (SearchService)
- Geolocalizar (GeolocService)
- Administrar mensajes (AdminMessageService)
- Administrar lugares (AdminPlaceService)

Cada uno de los servicios, llevará implementadas las siguientes clases:

- Clase DAO para persistencia.
- Interfaz DAO para su persistencia.
- Clase Service que se encarga de la lógica y de la llamada a la interfaz DAO.
- Interfaz Service para publicar su servicio.

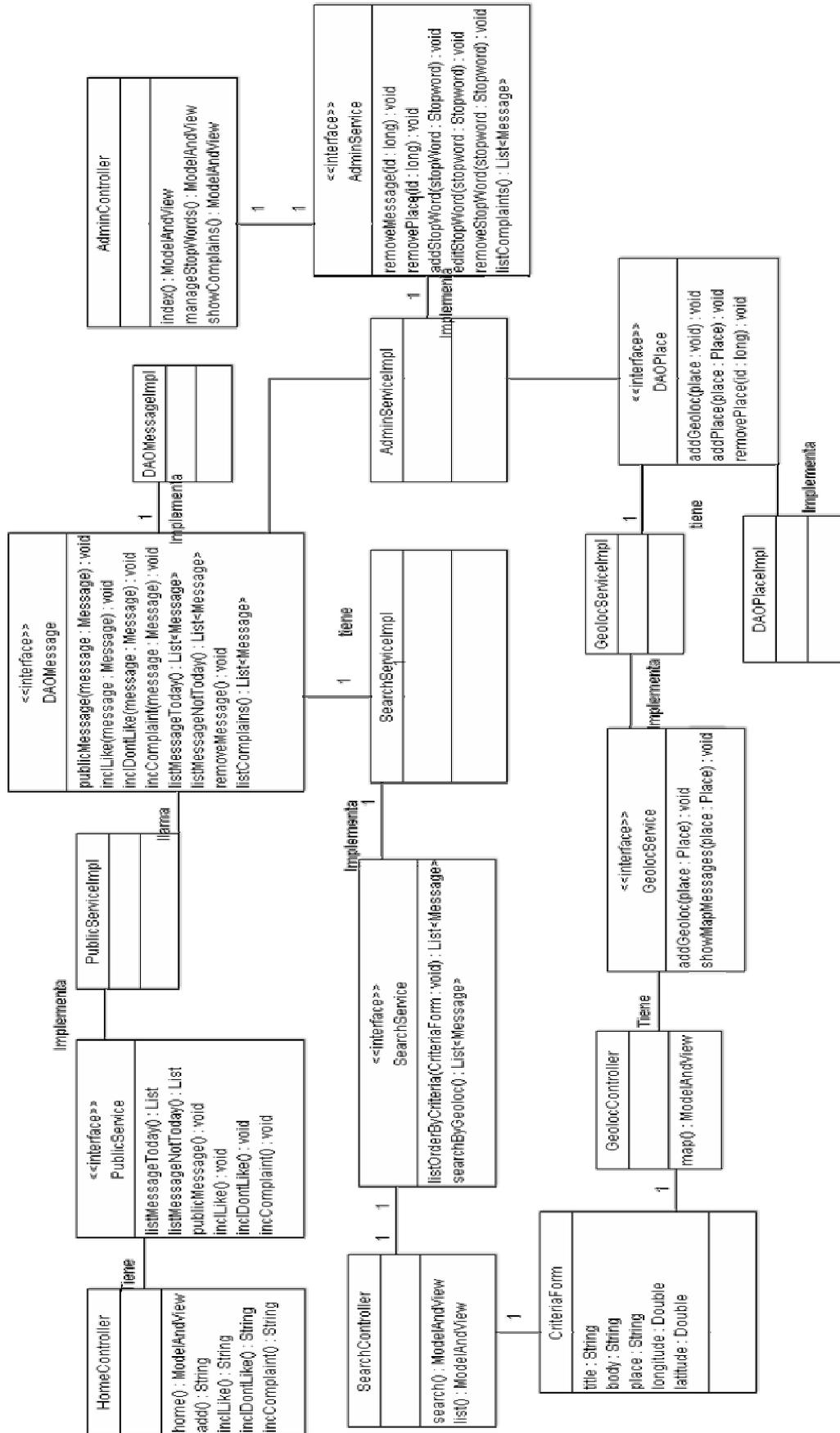
Por otra parte, cada una de las tablas de la BBDD, tendrá una clase correspondiente encargada de su persistencia.

Los controladores serán aquellas clases que estén detrás de cada una de las ventanas que se elaborarán. Existirán los siguientes:

Nombre del controlador	Descripción	Servicios que implementa
HomeController	Encargado del dashboard donde aparece la lista de mensajes ordenados por fecha	PublicService
SearchController	Encargado del formulario de búsqueda	SearchService
GeolocController	Encargado de la ventana de los mapas.	GeolocService
AdminController	Encargado de la página de administración	AdminMessageService AdminPlaceService

10.2 DIAGRAMA DE CLASES DE LOS SERVICIOS

A continuación se refleja el diagrama de clases relacionado con Spring MVC e Hibernate. Para facilidad del modelo, los métodos solo se agregarán a la interfaz y no se pondrán en las clases que implementan la interfaz.



11. DIAGRAMA DE SECUENCIA

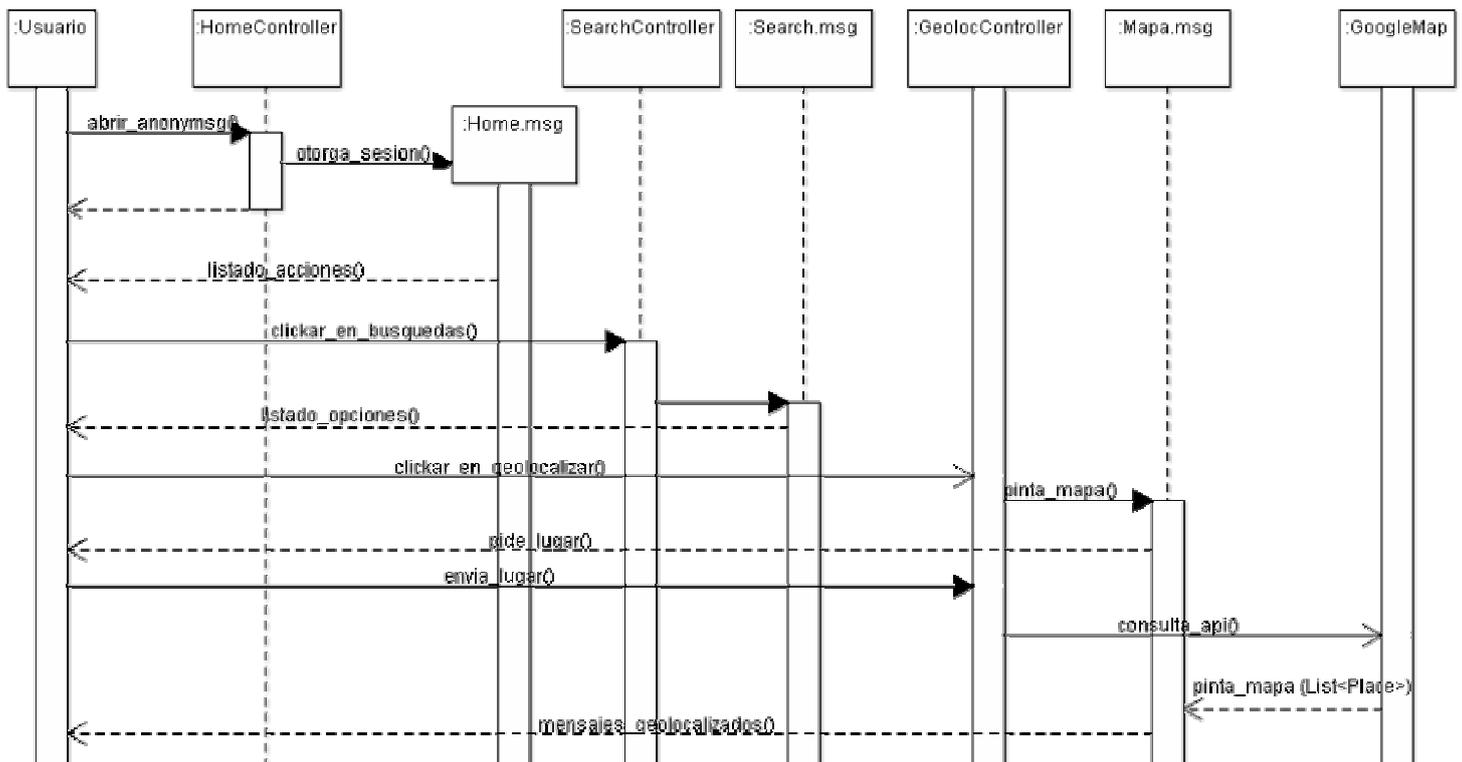
Para representar la interacción entre los elementos del sistema, empleamos un diagrama de secuencias. Como esta interacción es similar entre casi todas las operaciones menos en las que presentan un elemento externo (Google maps), tomaremos un ejemplo de estas últimas.

A continuación veremos un ejemplo de como mostrar mensajes geolocalizados en un lugar específico a través de AnonyMSG.

11.1 *DIAGRAMA DE SECUENCIA – CASO: MOSTRAR MENSAJES GEOLOCALIZADOS EN UN LUGAR ESPECÍFICO.*

El usuario se conecta a AnonyMSG, el sistema le proporciona un identificador de sesión. A continuación, HomeController proporciona la home al usuario, ya que es la primera página a la que se llega. En home.msg se muestran todas las opciones que hay y el usuario elige ir a búsquedas. En la página de búsquedas, search.msg, el usuario llama a la opción de geolocalización y en la pantalla del mapa, map.msg, teclea el nombre del lugar de su interés. El sistema le devolverá, mediante una consulta a la api de google, los mensajes en forma de puntos situados en el mapa presentes en el sistema y los de alrededor del lugar en cuestión.

Se describe gráficamente la secuencia descrita:



12. PROTOTIPOS

La aplicación consta básicamente de las siguientes pantallas que estarán presentes o no en dependencia del tamaño del dispositivo:

Tamaño del dispositivo	Pantallas	Descripción
Pequeño (Smartphones)	Inicio	Dividida en dos acciones: - Publicar mensajes - Buscar mensajes
Pequeño	Publicar mensajes	Diseñada para la publicación de los mensajes.
Pequeño	Buscar mensajes	Diseñada para la búsqueda de mensajes con una caja de texto abierta donde el usuario puede especificar cualquier palabra para su búsqueda.
Web y Tablet	Dashboard	Pantalla con tres widgets: - Publicar mensajes - Buscar mensajes - Conjunto de mensajes resultado de la búsqueda en caso de que se haya hecho alguna y en caso contrario, los últimos publicados. - Calificar y denunciar
Todos	Thickbox de Geolocalización	Pantalla de google maps donde el usuario geolocaliza el lugar.

Nota: Los prototipos se han realizado con Diagrama.ly, y son estrictamente orientativos para apoyar en el trabajo de la determinación de la navegación, interacciones y usabilidad.

12.1 PROTOTIPOS PARA MOBILE

PANTALLA DE INICIO



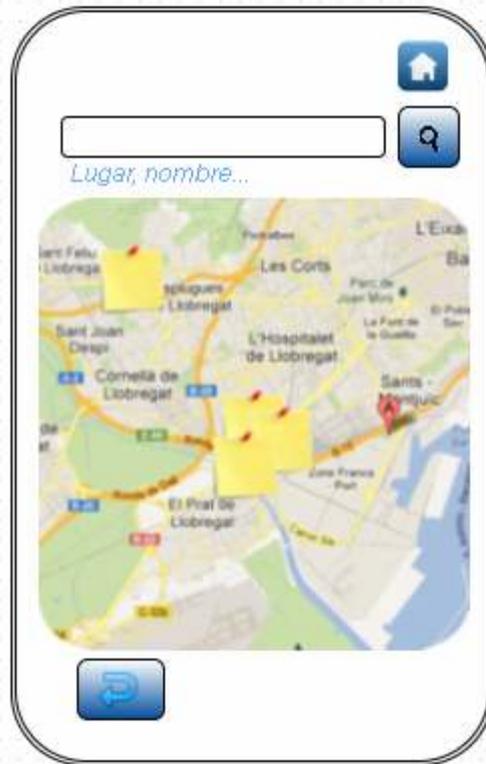
PANTALLA DE PUBLICAR



Si pulsamos en 'Geolocalizar mensaje', nos vamos a la siguiente ventana que conserva el buscador.

Los mensajes que aparezcan en el lugar indicado, se mostrarán señalados en el mapa.

PANTALLA DE GEOLOCALIZACION



PANTALLA DE LISTADO (home)



PANTALLA DE RESULTADOS



12.2 PROTOTIPOS PARA WEB

Al entrar en AnonyMSG, arribaremos a la home:



En dicha página podremos hacer la mayoría de las acciones.

Como nota cabe mencionar que todas las ventanas tendrán sus iconos propios de Minimizar, Redimensionar y Cerrar, aunque en los prototipos no aparezcan.

En caso de que queramos ampliar el mapa, se podrá hacer doble click en el mismo y se abrirá un thickbox con el mapa en grande.

Si se pulsa en el link '¿Ubicar en un mapa?' también se desplegará la misma ventana de thickbox que se muestra a continuación:



La ventana de búsqueda se mostrará como sigue:



12.3 PROTOTIPOS DEL PORTAL DE ADMINISTRACION

Portal de Administración de AnonyMSG

Formulario de inicio de sesión:

Login:

Contraseña:

Fig. Login del Portal de administración

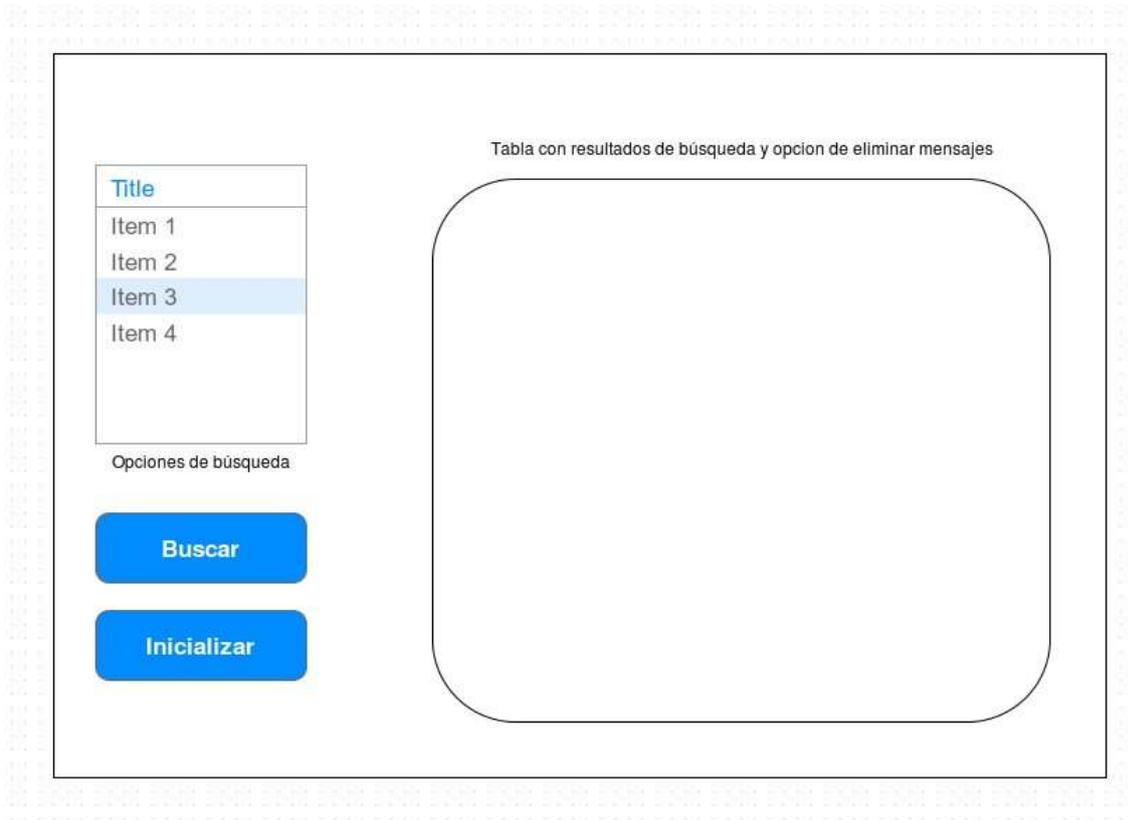


Fig. Representativa de como quedará la ventana de gestión de la herramienta de administración.

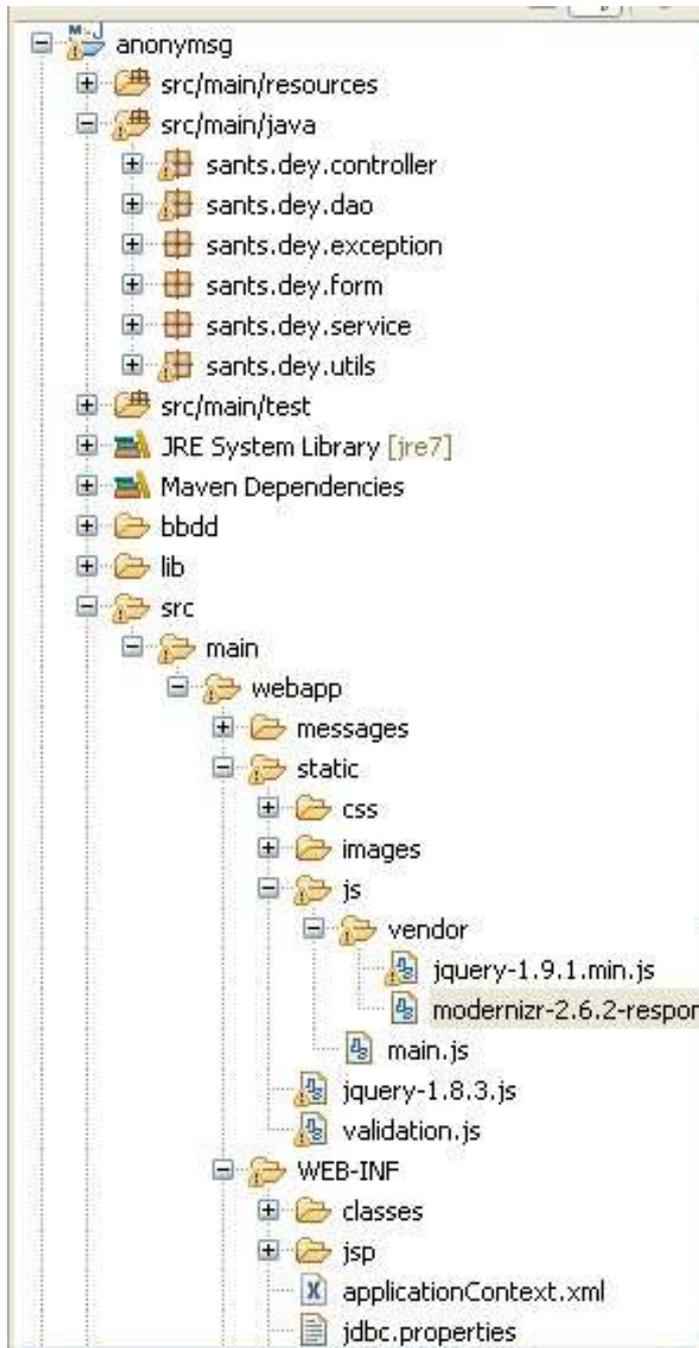
13. IMPLEMENTACION

La etapa de la implementación ha intentado cumplir con los siguientes requisitos:

- Intentar con el mínimo desarrollo posible, para lograr mantenibilidad, dar soporte a todos los tipos de dispositivo.
- Intentar frescura de datos para que los mensajes aparezcan visibles y localizados en el mínimo tiempo posible sin ir en detrimento del rendimiento de la aplicación. Los accesos a base de datos son solo para ir a buscar los datos y nunca para traer estructuras completas de html.
- Se intenta mantener ciertos criterios de usabilidad, ayuda contextual, titles en las imágenes, accesos directos como vías de salida, etc, para evitar que el usuario en algún momento se pueda encontrar perdido.
- Aplicación multi-idioma: Aunque la entrega se realiza solo para el castellano, existen los ficheros de properties y la infraestructura para cuando exista la posibilidad de realizar las traducciones.
- Código comentado para lograr mantenibilidad.
- Generación de los ficheros de base de datos de forma automática y versionada.

13.1 ORGANIZACIÓN DE LA ESTRUCTURA DEL PROYECTO

El proyecto, como se describió en secciones anteriores, se ha estructurado con de Maven a través de Eclipse. Las carpetas han quedado de la siguiente forma:



Hemos creado una carpeta para los ficheros de Base de Datos, los mismos han quedado ubicados siguiendo la estructura siguiente:



13.2 REPOSITORIO DE SOFTWARE

Como repositorio de software, se ha utilizado el p4. Se trabajó con el plugin para el STS y así poder obtener un versionado del software.

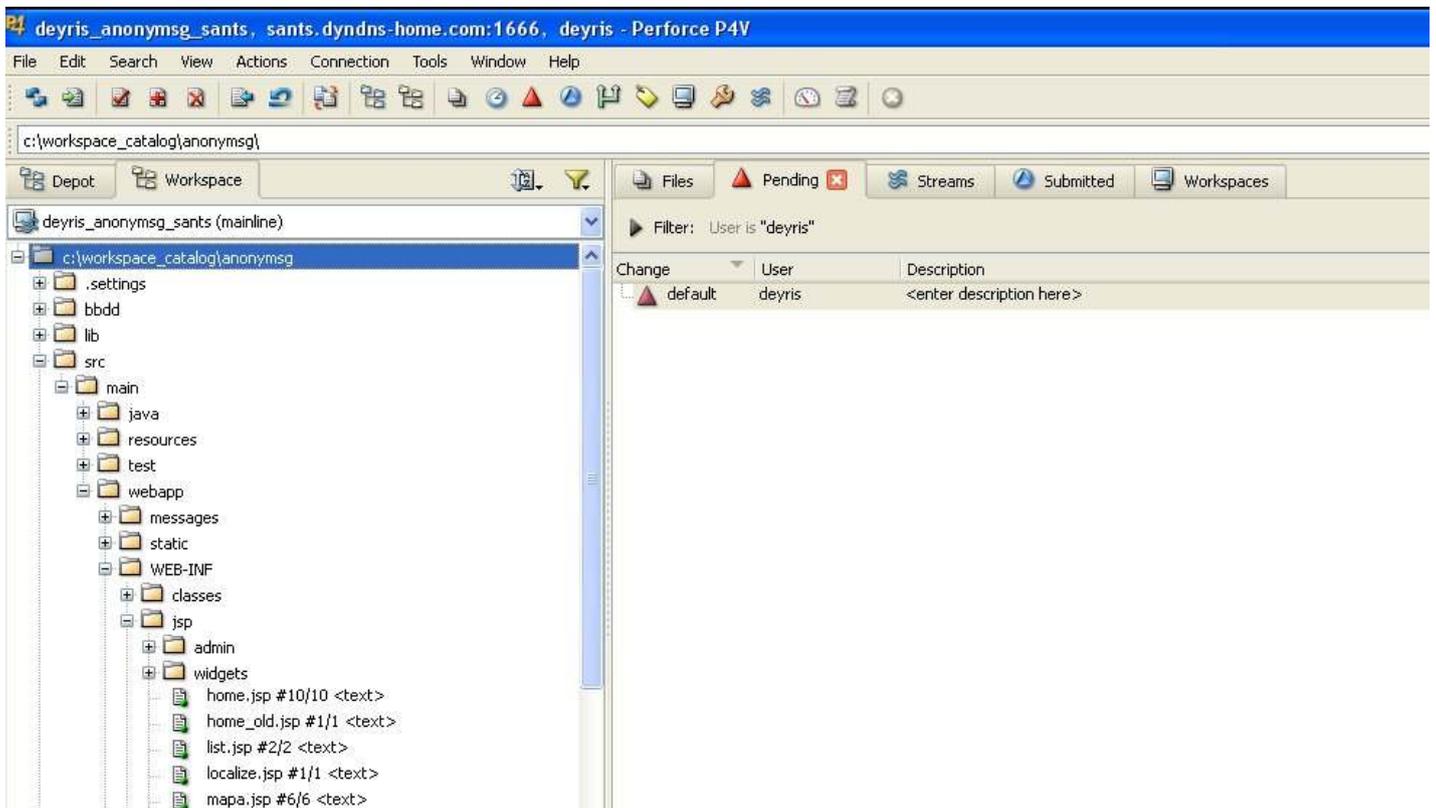


Fig. Proyecto en el repositorio de Perforce

13.3 *TEMPLATES HTML5 BOILERPLATE*

¿Por qué rechazamos jquerymobile?

El gran problema de este framework es que esta basado en librerías “desktop” (por lo que la adaptación no es óptima), además, no ofrecen todas las características HTML5, desaprovechando muchas de las capacidades del mismo. Ofrece un rendimiento más bajo y una etapa de desarrollo más complejo.

En cambio html5 Boilerplate, nos ofrece las siguientes ventajas:

Entre las características de HTML5 Boilerplate cabe destacar:

- Multinavegador (también IE6)
- Preparado para HTML5
- Optimizaciones para la visualización en navegadores para móviles
- Clases específicas para IE, que facilitan un mayor control multinavegador
- Perfiles Javascript en función del navegador
- Una hoja de estilos de impresión óptima
- IE6 pngfix

Las plantillas son fáciles de adaptar de forma robusta y no se necesita mucho tiempo para empezar a ver los resultados.

Par utilizarlo, hemos incluido en la cabecera de la aplicación las llamadas correspondientes a los ficheros .js y css.

13.4 *ORGANIZACIÓN DE CSSs*

Los ficheros de css, quedan organizados de la siguiente forma:

Fichero	Descripción
Global.css	Configuración de las características comunes a los elementos.
Main.css	Configuración determinada por el tamaño de la ventana que emite la llamada.
Normalize.css	Ficheros de HTML5 Boilerplate
Normalize_min.css	Ficheros de HTML5 Boilerplate
admin_global.css	Configuración de las características comunes a los elementos para el Portal de Administración.
Admin_main.css	Configuración determinada por el tamaño de la ventana que emite la llamada para el Portal de Administración.

En los ficheros main.css y admin_main.css, existe el código que maneja como se pintarán las css en las diferentes plataformas. Así podremos ver una misma ventana cargada de diferentes formas y funcionando de diferentes formas basado en el cambio de tamaño con el que se muestra.



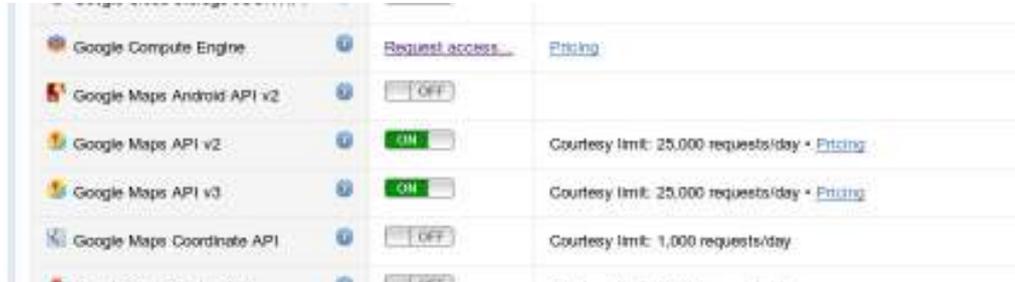
Fig. Ventana web del Portal de Administración.



Fig. Ventana versión mobile del Portal de Administración visualizada a tamaño de dispositivo mobile. Visualización del Firebug, herramienta fundamental utilizada para el desarrollo responsive.

13.5 GEOLOCALIZACIÓN

Para gestionar la geolocalización de la aplicación, hemos dado de alta servicios de la api de Google.



Cuando el usuario va a geolocalizar el mensaje, la aplicación pedirá autorización para geolocalizar su posición y que la ventana del mapa muestre su ubicación. La petición solo la realiza la primera vez que se accede. Existe una función timeout que calcula un tiempo de 4 segundos:

```
navigator.geolocation.getCurrentPosition(success,
error, {maximumAge:60000, timeout: 4000});
```

Si no se puede localizar su ubicación, la posición por defecto es Barcelona

```
var divMapa = document.getElementById("geoMap");
    var barcelona = new
google.maps.LatLng(41.39483307195539, 2.1501411410156512);
    var myOptions = {
        zoom: 12,
        mapTypeId: google.maps.MapTypeId.ROADMAP,
        center: barcelona
    };
```

13.6 JSON Y JQUERY

Las peticiones de datos se realizan a través de la librería de javascript jquery. A continuación vemos el ejemplo del código que se encarga de guardar los mensajes en la bbdd:

```
var guardarSms = function(st){
    $('#form#message')[0].submit();
    //var msg = new Object();
    var title=$('#title').val();
    var body=$('#body').val();
    var place=$('#place').val();
    var id=$('#id').val();
    var state=st;
```

```

var data = "title=" + title + "&body=" + body + "&place=" + place + "&state="
+ state;
if (id!=""){
    data += "&id=" + id;
}
$.ajax({
    type:"POST",
    url: './home/addWithReturn',
    data:data,
    success: function(data) {
        if(st==1){
            $('#addmsg').removeClass('hide');
            rellenarListado();
        }else{
            //Guardamos id del mensaje y el estado en los hidden
            $('#state').val(2);
            $('#id').val(data);
            abrirPopup('./mapa.msg?idMessage=' +
data,'Geolocalizar mensaje');
        }
    },
    error:function(data, textStatus, errorThrown){
        alert("error\n" + data + ", " + textStatus + ", " + errorThrown);
    }
});
};

```

Las llamadas se generan una dependiente de la otra para evitar que se solapen por la asincronía de ajax.

El objeto data que se obtiene en javascript, se utiliza luego en la página para pintar los listados que van cambiando dinámicamente sin la necesidad de que el usuario recargue la página.

Para probar las salidas json obtenidas, se ha utilizado un editor de json online:

<http://braincast.nl/samples/jsoneditor/>

13.7 SINCRONIZACIÓN CLIENTE-SERVIDOR

La aplicación está en constante actualización con el servidor. Hay un evento timeout que se encarga de lanzar llamadas para comprobar si existen mensajes nuevos. La llamada está activada solo si el usuario está en la ventana de publicación de mensajes y no en la ventana de búsqueda.

El disparador del evento timeout es el body en su evento onLoad:

```
<body onload="messageTimeout ();">
```

La declaración del evento es la que sigue, en la misma se lanza una llamada a la función que rellena el listado de mensajes a x milisegundos. La variable REFRESH_TIME deberá estar especificada.

```
var messageTimeout = function () {
    setInterval(function () {
        rellenarListado();
    }, REFRESH_TIME);
};
```

Com la llamada `rellenarListado ()` trae resultados por es ajax/jquery/json, no hay recarga de página.

Para la variable `REFRESH_TIME` se hacen pruebas de carga para el servidor en el que está montada actualmente la aplicación. Actualmente hay un refresco cada 10 segundos (`REFRESH_TIME=10000`).

13.7.1 DESARROLLO PARA LA VISUALIZACIÓN MOBILE

La web para mobile, está básicamente ligada al diseño responsive. Toda la carga de la aplicación básicamente se hace en una única página. Lo que hace el javascript de la página es conseguir que las secciones se cierren y se abran según la petición del usuario.

Para conseguirlo se han creado unos escenarios y según el momento, se carga un escenario u otro:

```
var defineState_0 = function () {
    //La home, todo lo demás está oculto
    $('#header').removeClass('hidden');
    if(idDispositivo==0){
        $('#footer').addClass('hidden');
        $('#mainContent').addClass('hidden');
    }else{
        $('#sectionPublic').addClass('hidden');
    }
    current_state=HOME_STATE;
};

var defineState_1 = function () {

    if(idDispositivo==0){
        $('#header').addClass('hidden');
        $('#mainContent').removeClass('hidden');
        $('#sectionPublic').removeClass('hidden');
        $('#sectionBuscar').addClass('hidden');
        $('#scroll').addClass('hidden');
        //$('#divSmsToday').addClass('notToday');
        $(document).scrollTop(
            $('#sectionPublic').offset().top );
    }else{
        $('#sectionPublic').removeClass('hidden');
        $('#sectionBuscar').addClass('hidden');
        $('#scroll').removeClass('hidden');
        $('#mensajesBuscados').addClass('hidden');
        $('#mensajesPublicados').removeClass('hidden');
    }

    current_state=PUBLIC_STATE;
};
```

```

};

var defineState_2 = function () {
    if(idDispositivo==0) {
        $('#header').addClass('hidden');
        $('#mainContent').removeClass('hidden');
        $('#sectionPublic').addClass('hidden');
        $('#sectionBuscar').removeClass('hidden');
        $('#scroll').addClass('hidden');
        $(document).scrollTop(
            $("#sectionBuscar").offset().top );
    } else {
        $('#sectionPublic').addClass('hidden');
        $('#sectionBuscar').removeClass('hidden');
    }
    current_state=SEARCH_STATE;
};

var defineState_3 = function () {
    if(idDispositivo==0) {
        $('#scroll').removeClass('hidden');
    } else {
        $('#scroll').removeClass('hidden');
        $('#mensajesPublicados').addClass('hidden');
        $('#mensajesBuscados').removeClass('hidden');
        $('#sectionBuscar').removeClass('hidden');
        $('#sectionPublic').addClass('hidden');
    }
    current_state=SEARCH_LIST;
};

var defineState_4 = function () {
    if(idDispositivo==0) {
        $('#header').addClass('hidden');
        $('#scroll').removeClass('hidden');
        $('#mainContent').removeClass('hidden');
        $('#mensajesPublicados').removeClass('hidden');
        $('#sectionBuscar').addClass('hidden');
        $('#sectionPublic').addClass('hidden');
    } else {
        $('#mensajesPublicados').removeClass('hidden');
        $('#sectionBuscar').addClass('hidden');
        $('#sectionPublic').addClass('hidden');
    }
    current_state=PUBLIC_LIST;
};

```

De esta forma se consigue que todo el código se cargue no más entrar a la aplicación. Después de esa primera carga, todo lo demás que se trae al cliente son datos a través de las llamadas con json.

En un principio se quería conseguir que no hubiera una recarga de las páginas en ningún momento, y se usan ventanas modales para la geolocalización. Pero al probar la aplicación en el simulador, las llamadas a las ventanas modales no funcionaban, con lo que esta parte de la aplicación para android, ha sido una de las últimas modificaciones hechas. En el caso que la ventana se recargue desde un dispositivo móvil, se hace una redirección.

Toda la salvaguarda de datos se hace en las ventanas independientes, sin tener que pasar los datos de una ventana a otra para evitar las ventanas modales debido a esta diferencia de comportamiento entre las plataformas.

13.8 DESARROLLO PARA ANDROID

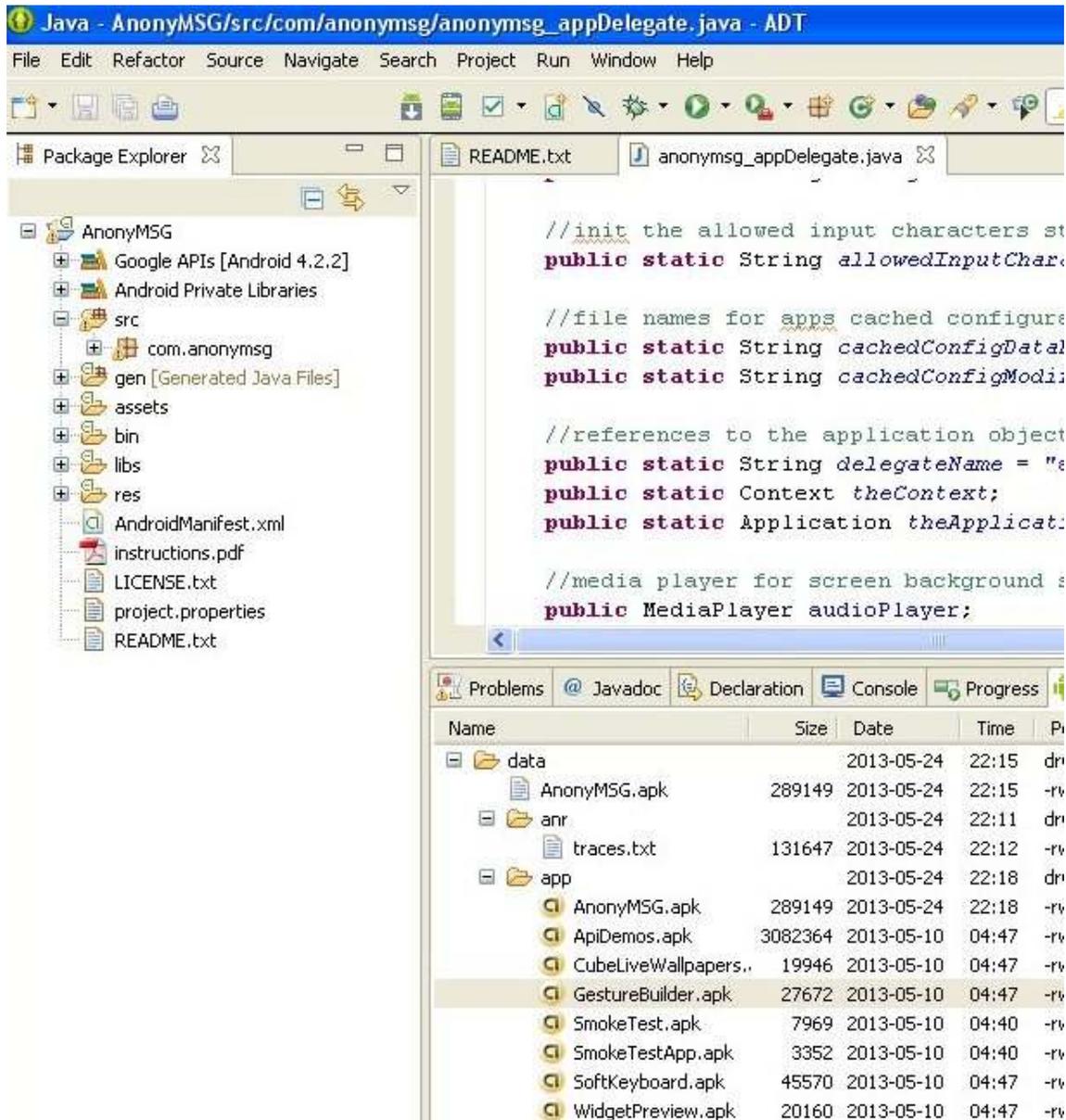
Se decide hacer una aplicación mobile para android que funcione como lanzador de la página. De esta forma podremos cumplir con el objetivo de dejar la aplicación funcionando como si se fuera a subir a Play Store.

Para ello se ha desarrollado un proyecto pequeño tomando una plantilla a través de una interfaz de servicios conocida como <http://buzztouch.com/>

Hemos seleccionado una simple aplicación que llama a nuestra url. Buzztouch genera un código básico de app mobile que se ha instalado en un ADT.

Para instalar el ADT, nos hemos actualizado el SDK correspondiente y actualizado los paquetes correspondientes para Android.

Se muestra a continuación una imagen de la organización del proyecto una vez instalado en el ADT:



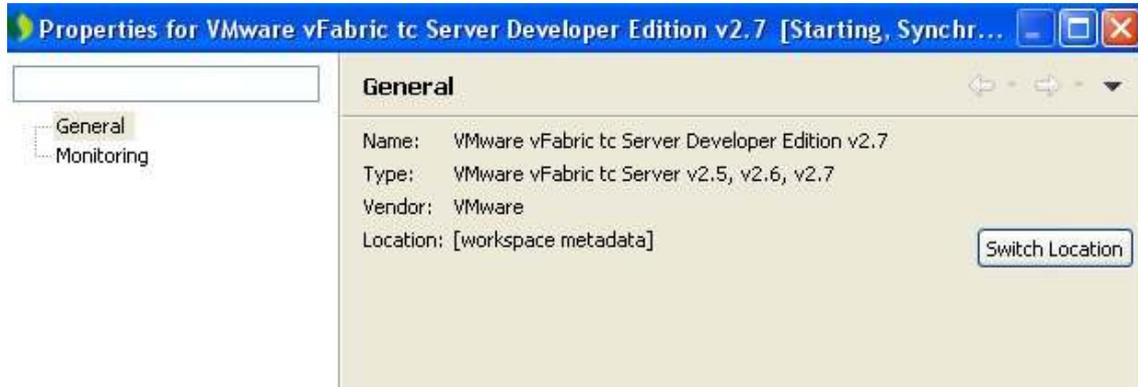
Se genera el proyecto AnonyMSG para android 4.0.3 y se compila. A continuación se genera un apk que nos instalamos en nuestros dispositivos Android como una aplicación más.

Se generan algunas pruebas con un simulador. A continuación se muestran las características técnicas del simulador elegido teniendo en cuenta las prestaciones del ordenador con el que se ha desarrollado:

```
[2013-05-24 15:40:02 - AnonyMSG] New keystore C:\Androide\adt-bundle-
windows-x86-20130514\eclipse\AnonyMSG.apk has been created.
[2013-05-24 15:40:02 - AnonyMSG] Certificate fingerprints:
[2013-05-24 15:40:02 - AnonyMSG] MD5 :
37:D0:7C:B2:63:D4:2F:63:6F:50:0D:7B:0E:C7:3B:65
[2013-05-24 15:40:02 - AnonyMSG] SHA1:
C2:41:49:D1:FD:5F:02:A7:99:23:00:EF:6A:86:08:28:42:D0:35:6F
```

13.9 EJECUCION DE LA APLICACIÓN DURANTE LA ETAPA DE IMPLEMENTACIÓN

Durante el desarrollo empleamos el software de servidor que viene por defecto con el Eclipse:



Para hacer las pruebas sobre un dispositivo móvil, empleamos un dispositivo móvil HTC con versión de Android 2.3.5. Se hacen pruebas además con un simulador del ADT.

13.9.1 Para instalar el proyecto

Se han de seguir las siguientes instrucciones de instalación:

Requerimientos básicos:

- Servidor web
- Servidor MySQL
- Conexión a internet
- Servicio DynDns instalado y puerto 8080 abierto.

Instrucciones:

- Ejecutar los scripts de creación e instalación de bbdd. Los mismos se encuentran dentro de la carpeta del proyecto en la ubicación de bbdd. Las llamadas a ejecutar podrán ser las siguientes:

Estando dentro del directorio: `\anonymsg\bbdd\instalation`, ejecutar:

```
mysql anonymsg --password=habanera < instalation_anonymsg_v1.sql
```

De esta forma quedará montada la base de datos en estado inicial.

Existe también el fichero script para deshacer la instalación que se ejecutaría con la siguiente llamada:

```
mysql anonymsg --password=habanera < instalation_anonymsg_v1.sql
```

- A continuación se debe copiar el fichero `anonymsg.war` dentro del directorio correspondiente del servidor. En nuestro caso lo colocamos en la carpeta `webapps` del `apache`.
- Arrancar la aplicación. En caso de cambio de url, habría que darle de alta a la nueva url en los servicios de Google.

13.9.2 Pruebas orgánicas de la aplicación

Para probar la aplicación, se instala en un servidor Apache Tomcat con MySQL 5.0 sobre Linux centos en una máquina virtual.

Se copia el fichero .war que se ha obtenido con maven y se coloca en la carpeta webapps del servidor.

La dirección con la que se genera el proyecto, se da de alta tanto en Google Api como en la aplicación de Android, con lo que es a esa dirección a la que se redirigirán nuestras pruebas.

El principal obstáculo para este tipo de proyecto se ha encontrado en la elaboración de las pruebas para los diferentes dispositivos y navegadores.

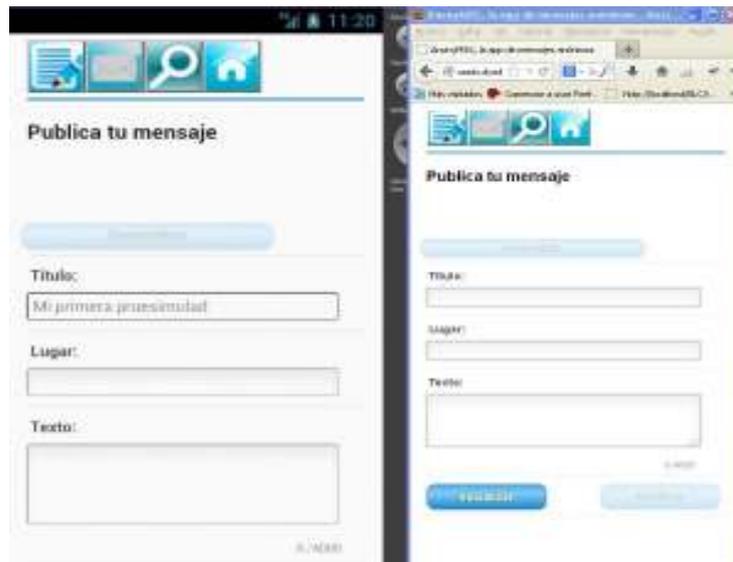


Fig. A la izquierda, desarrollo para Android desde el simulador. A la derecha, vista con el navegador Mozilla Firefox de la sección de publicación de mensajes.

14. DESCRIPCIÓN DE LA APLICACIÓN

Todo lo que se puede hacer con la aplicación está en la home. Como existen plataformas que no permiten ventanas modales, se ha tratado de evitar el paso de información entre ventanas.

Las funcionalidades que se describirán a continuación son las siguientes:

- Visualización de mensajes:
 - o Visualización de mensajes publicados hoy.
 - o Visualización de mensajes publicados en el sistema.
 - o Calificación de los mensajes (Me gusta, No me gusta y Denuncias)

- Publicación de un mensaje:
 - o Publicación
 - o Geolocalización de lugares

- Búsquedas:
 - o Búsquedas de mensajes por coincidencia de texto.
 - o Búsqueda de mensajes en un mapa.

14.1 MOVILIDAD A TRAVÉS DE LAS OPCIONES DE LA APLICACIÓN

Tanto en plataforma web como mobile, podemos decir que todas las opciones están accesibles fácilmente. En web, todas las opciones aparecerán siempre visibles en la cabecera. En mobile, las opciones también estarán disponibles desde la cabecera de los widgets y algunos en el footer.

Por ejemplo, publicar mensaje, podremos encontrarlo tanto en la home principal en mobile como en cada uno de las cabeceras de las secciones:



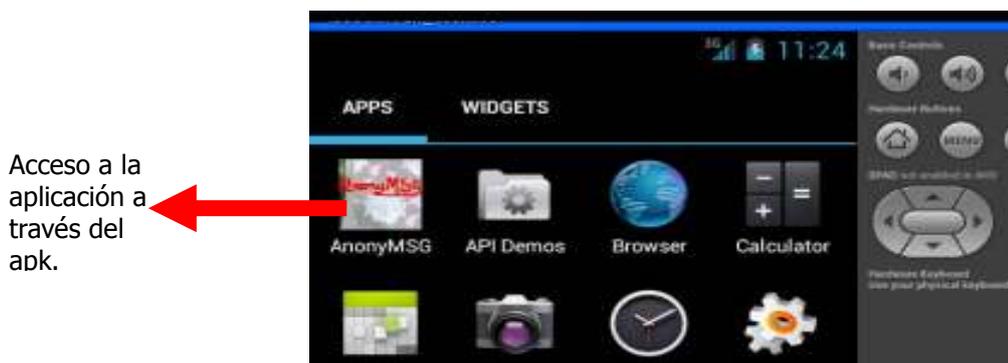
Fig. Menú de la aplicación mobile

14.2 FUNCIONAMIENTO MOBILE

Describiremos a continuación las diferentes funcionalidades en el modo mobile.

14.2.1 Listar mensajes

Accedemos a la aplicación a través del icono que aparecería en el escritorio:



Acceso a la aplicación a través del apk.

Fig. Escritorio de mobile con el icono de AnonyMSG.



Fig. Acceso al listado de los últimos mensajes.

La visualización de los mensajes del sistema en mobile, se muestran como sigue a continuación. A dichos mensajes se accede desde el botón 'Últimos mensajes anónimos' y también existe la sección 'Ver más mensajes'.



Fig. Mapa con puntos geolocalizados

Texto de los mensajes que se despliega al clicar sobre los marcadores.



Fig. Detalle del texto del mensaje

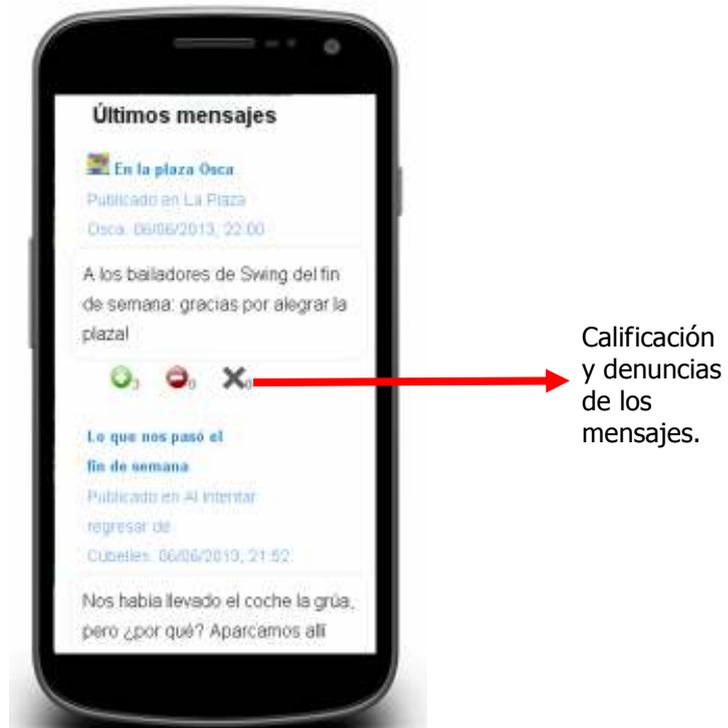


Fig. Listado con detalles de los mensajes

14.2.2 Publicación de los mensajes:

Se describe a continuación, el proceso para móviles que es bastante similar solo con la característica de que como no podemos mostrar el mapa de geolocalización en una ventana emergente, hacemos una redirección a una página diferente.



Fig. Acceso a la publicación de anónimos.

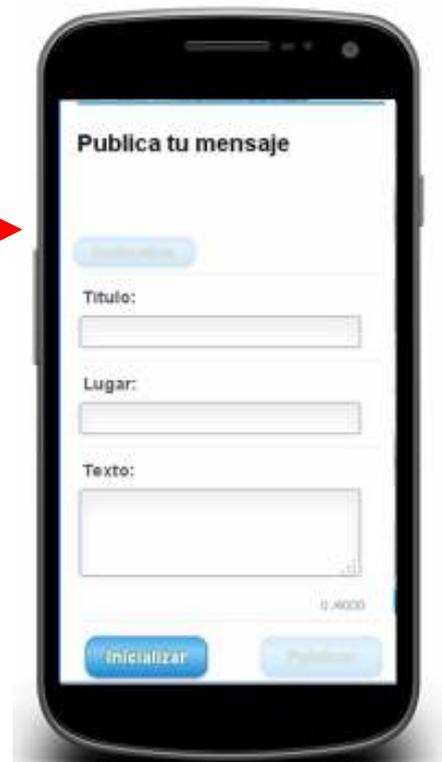


Fig. Formulario de entrada.

Cuando se accede al formulario, se podría publicar el mensaje sin geolocalización pulsando en publicar o geolocalizándolo haciendo click en el botón de geolocalización.

Tanto el botón de Geolocalización como el de Publicar, se activarán una vez que el usuario haya completado todos los campos obligatorios para disminuir la posibilidad de errores.



Fig. Mapa de geolocalización de mensajes



Fig. Mapa con mensaje ya publicado.

La barra de búsquedas, siempre que encuentre un lugar, es capaz de ubicarlo en el mapa. Una vez ubicado, el sistema nos deja poner un único marcador en el mismo. Si pulsamos aceptar, a continuación veremos nuestro mensaje publicado y geolocalizado en el sitio correspondiente.

14.2.3 Búsqueda de los mensajes:

A continuación mostramos las ventanas del proceso de búsqueda:



Acceso a la búsqueda de mensajes.

Fig. Acceso a la búsqueda de mensajes.

Fig. Formulario de búsqueda

La barra buscadora se completa con la frase que se quiere encontrar en los mensajes. Se buscará en el campo título del mensaje o nombre del lugar.

A continuación aparecerán los mensajes geolocalizados en caso de que existan que coincidan con la búsqueda generada y debajo el listado con los detalles de los anónimos que cumplen con la búsqueda ejecutada.



En estos mensajes también tenemos acceso a la calificación y denuncias.

Fig. Listado de búsqueda generada.

14.3 FUNCIONAMIENTO WEB

La aplicación web consta de una ventana principal desde la que se puede acceder a todas las funcionalidades:

- Listar mensajes
- Búsqueda de mensajes
- Publicación de mensajes (Geolocalizados o no)
- Calificación de mensajes

Las funcionalidades son las mismas que las descritas en mobile con muy pocas variantes, ya que el principio de AnonyMSG es ser un desarrollo 'One web'.

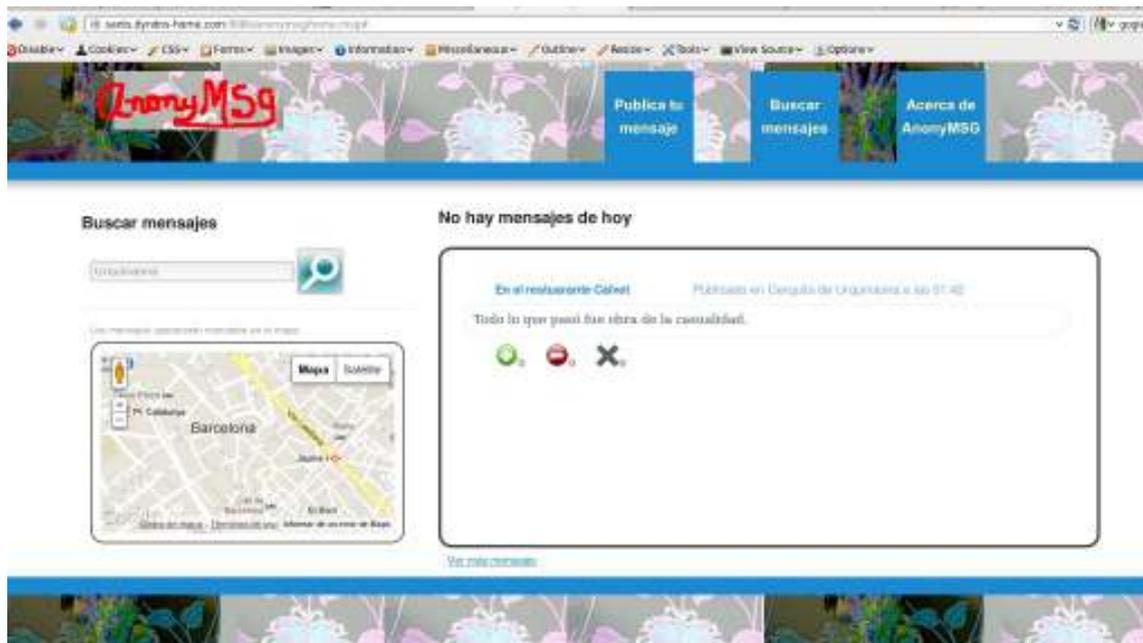


Fig. de la home de AnonyMSG

Las listas de los mensajes aparecerán siempre en la home según la petición del usuario. Lo que va variando es el menú de la izquierda según si lo que se quiere es publicar o buscar.

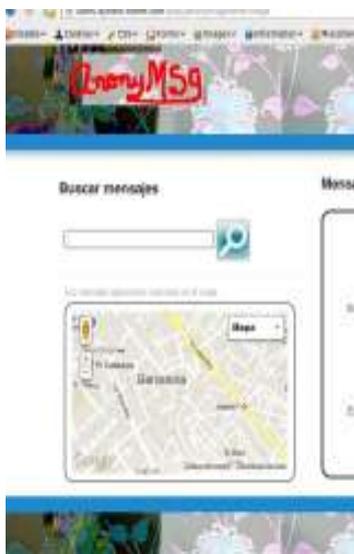


Fig. Formulario de búsqueda



Fig. Resultados de una búsqueda.



Fig. Formulario de publicación de mensajes. Una vez que el mensaje haya sido publicado, aparecerá el mensaje de confirmación y el mismo aparecerá en el listado de 'Últimos mensajes publicados'.

14.4 HERRAMIENTA DE ADMINISTRACION

Para controlar la publicación de los mensajes, poderlos eliminar y saber como van las calificaciones, se ha generado la herramienta administrativa de AnonyMSG.

La url de acceso es:

http://url del proyecto/portal_anonymsg

Se accede con usuario y contraseña.

Las ventanas que describen la herramienta se encuentra a continuación:

La herramienta de administración es bastante sencilla. Presenta, como toda herramienta de administración, una ventana de login, dónde el usuario pondrá su usuario y su contraseña y accede a la página desde donde la que se podrá administrar los mensajes.



Fig. Login de la herramienta de administración.

Actualmente solo se encuentra desarrollada la ventana de administración de funciones sobre los mensajes, no hay aún gestión de usuarios.

La ventana de administración de los mensajes se muestra a continuación y a través de ella se podrá:

- ▲ Visualizar todos los mensajes.
- ▲ Filtrar según criterios de mensajes.
- ▲ Eliminar mensajes.

Gestión de mensajes

Activo
 Inactivo
 Borrado
 Todos

No. de denuncias mayor que:

Titulo:

Lugar:

Texto:

40 mensajes encontrados

id	Fecha	Estado	Titulo	Lugar	Texto				Acciones
40	08/06/2013, 21:44	ACTIVO	Para La Habana	en el camino a casa	Habrán, si nos ojos me abandonarán, si la vida me desentona a un rincón de la Tierra, yo te juro que soy a hombre de amor y de ganas, de andar tus calles...	0	0	0	Eliminar
39	08/06/2013, 21:40	ACTIVO	Marchando a casa	el camino a La Habana	Si sortara todas las veces que ha hecho este camino, serían más o menos así un centenario, así embargo todas son diferentes, hay muchas cosas, muchas alegrías... muchas sorpresas.	0	0	0	Eliminar
38	08/06/2013, 21:12	ACTIVO	Caminando por Riego	la calle Riego en Sants	Es mi calle preferida de la zona. Me encantan los bares, la terraza, los negocios de toda la vida y sobre todo, la Catalana...	0	0	0	Eliminar
37	08/06/2013, 22:00	ACTIVO	En la plaza Oza	La Plaza Oza	A los baladines de Swing del fin de semana: gracias por alegrar la plaza! Nos había llevado el coche la grúa, pero ¿por qué?	3	0	0	Eliminar

Fig. Panel de administración de los mensajes con su formulario de búsqueda.

Desde este panel también podremos saber qué mensajes tienen un número elevado de denuncias para poderlos eliminar.

Gestión de mensajes

Activo

Inactivo

Borrado

Todos

No. de denuncias mayor que:

Título:

Lugar:

Texto:

Inicializar **Buscar**

Cuando eliminamos los mensajes, pasan a estado 'BORRADO' pero no se eliminan físicamente del sistema.

El portal de Administración no ha seguido una línea de desarrollo responsive. Es una mejora planteada para el futuro en el siguiente apartado.

id	Fecha	Estado	Título	Nombre del lugar	Cuerpo	I like	I dont like	Denuncias	Acciones
26	2013-05-24 09:15:06.0	ACTIVO	Estoy tan cansada	no arreglará más nada	Como esté va.	0	0	0	Eliminar
27	2013-05-24 18:01:18.0	ACTIVO	pruebavapp	movil	FFF	0	2	0	Eliminar
12	2013-05-23 00:50:06.0	INACTIVO	prueba cuatro	prueba cuatro	prueba cuatros	0	0	0	Eliminar
20	2013-05-23 01:08:45.0	INACTIVO	Prueba once	sdfsdf	asdfsdf			0	Eliminar
1	2013-05-22 23:11:32.0	BORRADO	prueba	prueba	ssf sdfsdf	0	0	0	Eliminado
2	2013-05-22 23:12:13.0	BORRADO	prueba	prueba	ssf sdfsdf	0	0	0	Eliminado
3	2013-05-22 23:18:14.0	BORRADO	prueba	prueba	ssf sdfsdfhh	0	0	0	Eliminado
4	2013-05-23 00:44:56.0	BORRADO	prueba dos	prueba dos	prueba dos	0	0	0	Eliminado
5	2013-05-23 00:46:37.0	BORRADO	prueba dos	prueba dos	prueba dos	0	0	0	Eliminado

Fig. Tabla de mensajes de la herramienta de administración.

15. CONCLUSIONES

14.5 CUMPLIMIENTO DE OBJETIVOS

Los objetivos personales han sido cumplidos en cuanto a lo que al aprendizaje de las tecnologías implicadas en estos desarrollos se refiere.

Se ha hecho un estudio sobre las siguientes materias:

- Responsive Web Design
- SDK Android
- JQuery, HTML5.
- Spring + Hibernate

La aplicación aún tiene puntos de mejora que se valorarán en el apartado siguiente, pero aún así, su objetivo de publicación de mensajes, listados de los mensajes publicados y geolocalización cumplen con las expectativas.

Las diferentes pruebas en cuanto a tipos de dispositivo y navegador son pruebas bastante costosas aún, y no se ha de menospreciar las prestaciones técnicas de los equipos que se emplean para desarrollar.

Uno de los principales errores cometidos fue comenzar a probar en un simulador cuando el producto estaba prácticamente acabado y en ese momento ver que había funcionalidades que no iban debido a incompatibilidades de las plataformas.

El diseño responsive es una materia complicada dónde la organización del código es uno de sus fuertes principales para llegar a buen puerto.

La gran ventaja del desarrollo multiplataforma, es que con un mismo desarrollo, se cubre gran cantidad de dispositivos. Para una actualización de AnonyMSG, solo se necesita actualizar el desarrollo web y la actualización, automáticamente ya la ven los usuarios en sus terminales, sin necesidad de actualizar la carcasa de la aplicación. Es incuestionable que es mucho más cómodo para todos, para los usuarios, para los diseñadores, para el desarrollo. Es un ahorro que no se ha de pasar por alto cuando se piensa en un proyecto que necesita satisfacer a un gran número de usuario de ambas plataformas.

14.6 DESDE LO QUE SE PLANTEÓ HASTA LO QUE SE DESARROLLÓ

De lo planteado en su día, cabe comentar que como punto principal está el desecho de la tecnología de Jquerymobile. En apartados anteriores ya hemos comentado el por qué se desechó la misma, no obstante, también se debe señalar que para haberla empleado, se debía haber comenzado mucho antes en su empleo que el punto en donde se miró para aplicarla.

Es decir, si vamos a emplear Jquerymobile, ha de ser desde el principio del desarrollo cuando se vaya a implantar, porque es una tecnología bastante intrusiva y requiere un esfuerzo extra para la mantenibilidad multiplataforma.

14.7 PLANTEAMIENTO DE MEJORAS

Las mejoras más importantes a tener en cuenta son las siguientes:

- Revisión de funcionalidades en diferentes modelos de dispositivo, no solo en cuanto a visualización sino en cuanto a funcionamiento. Hay algunos dispositivos en los que no se ven los mapas, por ejemplo y otros dispositivos en los que el uso del scroll empobrece la experiencia de usuario significativamente.
- Revisión de las búsquedas, en principio se planteó el uso de un indexador previendo un volumen de mensajes que impidiera la navegación con fluidez.
- Ampliación de las funcionalidades de la herramienta de administración
- Revisión de los mensajes de comunicación con el usuario. Se ha intentado seguir la política de Evitar los errores mejor que informarlos a posteriori, pero aún así, hay algunos funcionamientos que quizá no hayan quedado bien explicados para el usuario.
- Hacer pruebas de funcionalidad con usuarios finales para así determinar el nivel de usabilidad de la aplicación.
- Desarrollar la carcasa para Ios de igual forma que se ha desarrollado para Android.

14.8 DESARROLLOS FUTUROS

Sería interesante el desarrollo de las siguientes funcionalidades:

- Utilización de la api Google Maps para mobile que ofrece la posibilidad de independencia de red para el trabajo con mapas y localizaciones.
- Optimización en las búsquedas.
- Introducir en los listados la posibilidad de clasificar los mensajes según temáticas y además poder filtrar por ellas.
- Introducir la posibilidad de respuestas entre usuarios.

14.9 OPINIÓN PERSONAL

La experiencia de llevar un desarrollo completamente en solitario desde el principio ha sido muy enriquecedora. Me he adentrado en el mundo de la maquetación, del diseño.

Doy las gracias a todas las horas de los desarrolladores de todo el mundo que enriquecen la web con toda la información necesaria para que los que vamos detrás podamos tener un camino más llevadero.

16. FUENTES DE INFORMACIÓN

La información toda fue consultada entre enero y mayo del 2013.

<http://java4developers.com/2011/ejemplo-basico-de-spring-mvc-con-maven/>

http://es.wikipedia.org/wiki/Apache_Solr

<http://www.desarrolloweb.com/articulos/introduccion-jquery-mobile.html>

<http://viralpatel.net/blogs/spring3-mvc-hibernate-maven-tutorial-eclipse-example/>

<http://es.wikipedia.org/wiki/Hibernate>

<http://html5boilerplate.com/>

<http://webintenta.com/HTML5-Boilerplate.html>

<http://braincast.nl/samples/jsoneditor/>