

Plataforma web para la gestión de proyectos en Bilwi (Nicaragua)

Tesis de final de Grado presentada por la "Escola d'Enginyeria de
Telecomunicació de Barcelona" Universitat Politècnica de
Catalunya por **Laura Echevarría Garuz**

En cumplimiento parcial de los requisitos del grado en "Ciències
i Tecnologies de les Comunicacions"

Tutora: **Eva Maria Vidal Lopez**

Barcelona, mayo 2017

Abstract

This Project is the result of the collaboration between “Associació d’Universitaris per a la Cooperació de la UPC” (AUCOOP) and Bilwi Council. Bilwi is located in Puerto Cabezas, a District in the Caribbean Coast, in Nicaragua.

The reason of this project is to create a web to show all the cooperation projects, which are done by Bilwi, to the rest of the world for help. Moreover, this website also seeks to provide citizens with the information they need about the council services and with the news about the town.

The result has been a web which is very clear to use and that can be customized (without knowledge of code), because of this, it is possible to extend this web to other poor institutions. In addition, the Council can work off-line (without Internet) so the work can continue been done when there is a lack of Internet connection and it is designed in different languages to reach each part of the world. All this has been taken into account in avoiding any payment plugin as another websites.

Resumen

Este proyecto surge de la Asociación de Universitarios por la Cooperación de la UPC (AUCOOP) y la alcaldía de Bilwi, una ciudad situada en el municipio de Puerto Cabezas al norte de la costa Caribe nicaragüense.

El proyecto se basa en crear una web que permita mostrar los distintos proyectos de cooperación que se llevan a cabo en Bilwi al resto del mundo, para recibir ayuda, e informar a los habitantes de todas las noticias y servicios de su ciudad.

El resultado de ello ha sido una web fácil de manejar y personalizable para poderla extender a otras instituciones del tercer mundo. Además permite trabajar en modo local (sin internet) para no detener el trabajo y en distintos idiomas para llegar a más sitios. Todo esto, teniendo en cuenta que esta plataforma web no cuenta con ningún *plugin* de pago como otras.

Resum

Aquest projecte sorgeix de l'Associació d'Universitaris per a la Cooperació de la UPC (AUCOOP) i l'ajuntament de Bilwi, una ciutat situada en el municipi de Puerto Cabezas, al nord de la costa Carib nicaragüenca.

El projecte es basa en crear una web que permeti mostrar els diferents projectes de cooperació que es realitzen a Bilwi a la resta del món per rebre ajuda. Aquesta web també pretén informar als ciutadans de Bilwi informar-se de totes les notícies i serveis de la ciutat.

El resultat de tot això ha estat una web molt fàcil de manegar i personalitzable, per tal que es pugui portar a altres institucions del tercer món. A més a més, permet treballar en local (sense necessitat d'Internet) per tal de no interrompre el treball. També està dissenyada en diferents idiomes per a poder arribar a qualsevol racó del món. Tot això, tenint en compte que la plataforma web que s'ha dissenyat no compta amb cap mena de *plugin* de pagament com altres webs.

Agradecimientos

En primer lugar me gustaría agradecer a la alcaldía de Bilwi por depositar su confianza en mí. En especial, al alcalde Reynaldo Francis Watson, que nos propuso este reto y confió plenamente en nosotros para ejecutarlo. También agradecer al informático de la alcaldía Iván Osmar Boork y a los dos Directores de Cooperación Externa, Liza Lindo (estuvo en el puesto el año pasado, cuando se inició este proyecto) y Danny Wilson (que relevó el puesto a Liza a inicios del 2017 y ha sido el enlace entre la alcaldía y yo); ellos tres han estado desde el minuto cero hasta hoy en contacto conmigo, para poder realizar la web lo más intuitiva y práctica para los habitantes de Bilwi.

Otra persona que confió en mí fue mi tutora Eva Maria Vidal López, quizá suena tópico, porque si aceptó llevar a cabo este proyecto era porque creía que se podía obtener un buen resultado, pero por ello mismo, fue un gran apoyo para mí, por creer que era posible, por presionarme a sacar lo mejor de la web y por seguirme retando a mejorarla. A pesar que la programación web no sea su especialidad, ella ha sabido aportarme la visión de un posible usuario tanto de Bilwi como de los posibles interesados en ayudar a la alcaldía; por eso, ha sido mucho más fácil saber cómo organizar la información en la web y que elementos eran clave para que esta no fuese una web más.

Aunque este proyecto tampoco hubiese sido posible sin la ayuda de todas esas personas que me han facilitado el camino dándome consejos técnicos. En especial, Arturo Echevarría Armisen, que me aconsejó y guio desde un principio sobre cómo enfocar el proyecto y los distintos lenguajes con los que podía trabajar. Y se prestó en todo momento a resolver mis dudas. Todo ello me facilitó avanzar cuando me sentía totalmente estancada. Por eso, muchas gracias papá.

Por último, necesité más de dos ojos para mejorar el diseño gráfico de la web, ya que nunca había trabajado en ello y no era visualmente atractiva. Oriol Homedes me enseñó cómo aplicar código *responsive* con el módulo de estilos CSS (*Cascading Style Sheets*) *flexbox*. También a Lorena Salamero Moreno, que actualmente ha estado trabajando en el diseño web y me ayudó a situar los elementos de la portada para que fuesen más atractivos al usuario. Ya por último, agradecer también a Roger Garcia Serrano, que ha tenido la paciencia de enseñarme a usar *Photoshop* y todos sus trucos para conseguir los logos que yo deseaba.

A todos vosotros y a mi familia, por su apoyo moral (Tere, Arturo, Clara y Roger), gracias. Gracias por vuestra confianza y paciencia.

Historial de revisiones

Revisión	Fecha	Motivo
0	01/05/2017	Creación de documento
1	02/05/2017	Introducción y objetivos
2	03/05/2017	State-of-art
3	04/05/2017	Plan de trabajo
4	05/05/2017	Manual de usuario (anexo)
5	06/05/2017	Metodología
6	07/06/2017	Metodología y resultados
7	07/06/2017	Revisión Eva
8	08/06/2017	Costes y conclusiones
9	09/06/2017	Formato del documento
10	09/06/2017	Lista de figuras y tablas
11	09/06/2017	Adjuntar todos los archivos de la memoria en uno
12	10/06/2017	Revisión de Eva Vidal
13	11/06/2017	Revisión de Laura Echevarría
14	12/06/2017	Revisión de formato y tablas
15	13/05/2017	Revisión final de Laura Echevarría

Escrito por:	Revisado por:
Laura Echevarría Garuz 13/05/2017	Eva Maria Vidal Lopez 14/05/2017
Autora del proyecto	Supervisora del proyecto

Índice

Abstract.....	2
Resumen	3
Resum	4
Agradecimientos	5
Historial de revisions.....	6
Lista de imágenes y figuras	9
Lista de tablas.....	10
1. Introducción	11
1.1 Objetivos	12
1.2 Requerimientos y especificaciones.....	13
1.3 Métodos y procedimientos	14
1.3.1 Código usado	14
1.3.2 Entorno de trabajo	15
1.4 Plan de trabajo	16
1.5 Desviaciones del plan inicial e incidencias.....	17
1.5.1 Plan inicial.....	17
1.5.2 Incidencias	17
2. State of art	18
2.1 Wordpress.....	18
2.2 Joomla	20
2.3 Elección de CMS	21
3. Metodología.....	22
3.1 Esquema de la web.....	22
3.2 Mapa OSM.....	26
3.3 Multi-idioma.....	29
3.3.1 Contenido fijo	29
3.3.2 Contenido BD.....	30
3.4 Nuevas funciones Situm	32
4. Resultados	35
5. Costes.....	37
5.1 Costes del proyecto	37
5.2 Costes con CMS	39
5.3 Costes reales	40
6. Conclusiones personales y desarrollo futuro.....	41

7.	Bibliografía	42
7.1	Webgrafía.....	42
7.1.3	Mapa	42
7.1.2	Librerías JavaScript	42
8.	Anexos.....	43
8.1	Anexo 1: Manual de usuario.....	43
8.1.1	Idioma predeterminado de Situm	43
8.1.2	Acceso a Situm.....	43
8.1.3	Estructura de Situm	44
8.1.4	Módulos de edición de datos	45
8.1.4.1	Funciones	46
8.1.4.2	Todos los módulos.....	54
8.1.5	Vocabulario	66
8.2	Anexo 2: Diagramas.....	67
9.	Glosario	75
9.1	Conceptos usados durante la tesis	75
9.2	Lista de acrónimos.....	75

Lista de imágenes y figuras

Ilustración 3.1 Estructura básica de la web (dos elementos fijos y el contenido del medio dinámico)	22
Ilustración 3.2 Esquema de escalabilidad.....	23
Ilustración 3.3 Cabecera de la web (elemento fijo).....	24
Ilustración 3.4 Pie de página de la web (elemento fijo)	26
Ilustración 3.5 Librerías de OpenLayer para acceder al mapa en local (librerías JavaScript y gráficos).....	27
Ilustración 3.6 OSM con conexión a internet	28
Ilustración 3.7 OSM sin conexión a internet (off-line).....	28
Ilustración 3.8 Ejemplo de una tabla de idiomas.....	30
Ilustración 3.9 Selección de permisos en Situm	34
Ilustración 3.10 Tabla de permisos de la BD, se han enmarcado las que hemos creado anteriormente (idelem=3).....	34
Ilustración 5.5.1 Torre de ordenador para ser usada como servidor (almacén de datos)	37
Ilustración 8.1Inicio sesión Situm	43
Ilustración 8.2 Pantalla inicio Situm	44
Ilustración 8.3 Listado módulos Situm	45
Ilustración 8.4 Opciones Situm.....	45
Ilustración 8.5 Lista de entradas de Situm	46
Ilustración 8.6 Otra lista de entradas de Situm	47
Ilustración 8.7 Lista de logos de Situm	47
Ilustración 8.8 Formulario nueva entrada de Situm	48
Ilustración 8.9 Formulario multi-idioma de Situm.....	48
Ilustración 8.10 Opción edición de Situm.....	50
Ilustración 8.11 Formulario de edición de Situm.....	50
Ilustración 8.12 Opción eliminar de Situm	51
Ilustración 8.13 Opción publicar de Situm	51
Ilustración 8.14 Opción entradas destacadas de Situm.....	52
Ilustración 8.15 Opción galería de imágenes de Situm.....	52
Ilustración 8.16 Opciones de la galería de imágenes de Situm.....	52
Ilustración 8.17 Más opciones de la galería de imágenes de Situm	53
Ilustración 8.18 Módulo 'Inicio' de Situm.....	54
Ilustración 8.19 Entradas de Inicio en la web	54
Ilustración 8.20 Contenido de la entrada Inicio en la web	55
Ilustración 8.21 Módulo 'Noticias' de Situm	55
Ilustración 8.22 Sub-módulo 'Tipo de noticia' de Situm	56
Ilustración 8.23 Mostrar noticias en la web	56
Ilustración 8.24 Formulario 'Noticias' de Situm.....	57
Ilustración 8.25 Contenido completo de una Noticia en la web.....	57
Ilustración 8.26 Mostrar imagen de la Noticia en la web (con Fancybox)	58
Ilustración 8.27 Módulo 'Servicios' de Situm.....	58
Ilustración 8.28 Sub-módulo 'Tipo de servicios' de Situm	59
Ilustración 8.29 Sub-módulo 'Logos de tipo de servicio' de Situm	59
Ilustración 8.30 Formulario de 'Tipo de servicio' en Situm.....	60
Ilustración 8.31 Página principal de servicios en la web.....	60
Ilustración 8.32 Selección de un tipo de servicio en la web	61

Ilustración 8.33 Contenido de un servicio en la web	61
Ilustración 8.34 Módulo de 'Proyectos' de Situm	62
Ilustración 8.35 Sub-módulo 'Estado del proyecto' de Situm	62
Ilustración 8.36 Sub-módulo 'Temática del proyecto' de Situm	62
Ilustración 8.37 Sub-módulo 'Logos del proyecto' de Situm	63
Ilustración 8.38 Formulario de los logos del proyecto de Situm	64
Ilustración 8.39 Página principal de los proyectos en la web	64
Ilustración 8.40 Contenido de un proyecto en la web (con Fancybox)	65
Ilustración 8.41 Formulario de contacto de la web	65

Lista de tablas

Tabla 1.1 Tabla de requerimientos y especificaciones de la web	13
Tabla 5.1 Tabla comparativa de precios de dominio	38
Tabla 5.2 Tabla de horas trabajadas y coste.....	38
Tabla 5.3 Tabla de costes totales (con IVA y IRPF)	39
Tabla 5.4 Costes por generar una web equiparable a la generada en este proyecto en Wordpress.....	40
Tabla 5.5 Costes reales para la alcaldía de Bilwi.....	40
Tabla 8.1 Diagrama de Gantt real. Como se han distribuido las tareas desde setiembre hasta mayo	69
Tabla 8.2 Repartición de las horas de trabajo	71
Tabla 9.1 A la izquierda los acrónimos mencionados en esta tesis y a la derecha los significados	75

1. Introducción

Este es un proyecto de cooperación para la alcaldía de Bilwi, con la que estuvimos realizando un proyecto de introducción de herramientas TIC en su localidad para que los niños y mujeres que habían sido excluidas de la sociedad tuviesen acceso a Internet para poder realizar sus tareas educativas. Este proyecto se realizó el pasado verano (agosto de 2016) por tres estudiantes de la ETSETB (Escola Tècnica Superior d'Enginyeria de Telecomunicacions de Barcelona) gracias a la colaboración del Centre de Cooperació per al Desenvolupament de la UPC (CCD).

La ciudad de Bilwi se ubica en el municipio de Puerto Cabezas, en la costa norte del Caribe nicaragüense. La población vive principalmente de la pesca, ya que debido a su gran lejanía con la capital, Managua, los productos tardan mucho en llegar (como alimentos como hortalizas y/o frutas, entre otros). Es una zona que vive de su propia tierra, por lo que necesitan ayuda para poder obtener más recursos para progresar. Además, alrededor de Bilwi, dentro del municipio de Puerto Cabezas, se localizan distintas aldeas las cuales, además de tener pocos recursos, también carecen de una buena educación y sistemas sanitarios para sus habitantes.

Debido a esta falta de recursos, la alcaldía de Bilwi decidió reunirse con nosotros durante nuestra estancia allí, para informarnos de todo esto y buscar una solución. En esta reunión nos comentaron que habían pensado en crear una web que les permitiese contactar con organizaciones que pudiesen estar interesadas en realizar proyectos de cooperación con ellos, de este modo, podrían mejorar sus tierras de conreo para poder obtener más alimentos y trabajo para los ciudadanos, mejorar la educación de los niños tanto en Bilwi como en las aldeas de alrededor, crear plataformas para concienciar a la gente de la situación de la mujer... entre otras mejoras. Además, también les interesaba tener una web para poder informar a sus habitantes de todos los servicios que se ofrecen desde la alcaldía, ya que se encontraban con que muchos de los habitantes no sabían muchos de los servicios que ofrecían. Por otro lado, querían que esta plataforma admitiese publicar noticias, ya que actualmente usan Facebook para ello y querían algo más serio.

Su primera propuesta fue el uso de alguna plataforma web gratuita (wordpress, blogspot...) pero las versiones gratis no les permitían poner todas las funcionalidades que ellos necesitaban. Además, querían su propio dominio del modo más barato posible. Tras mi regreso del viaje mantuve una charla con mi tutora Eva Vidal y se decidió crear una web que incluyera todas las funcionalidades que ellos necesitaban.

Por otro lado, la idea de esta plataforma también fue poder extenderla a otras instituciones, además de la alcaldía de Bilwi, con bajos recursos y con iniciativa por mejorar su situación. Por ello, se pretende que este proyecto pueda llegar a otras poblaciones con la colaboración de AUCOOP (Associació d'universitaris per a la Cooperació de la UPC) de la que formo parte desde hace 3 años y que organiza proyectos de cooperación.

1.1 Objetivos

Los objetivos del proyecto se han basado todos en función de sus necesidades y en crear una web lo más personalizable posible para extenderla a todas aquellas instituciones con problemáticas parecidas.

1. Crear un administrador. Para la gestión de todos los datos de la web se decidió incluir un administrador que permita introducir toda aquella información que ha de incluir el cliente para que se visualice en el portal web. Para ello se planteó buscar un administrador ya existente de código libre y adaptarlo a la nueva web.
2. Permitir que la web esté en varios idiomas para poder acceder a más países y, por consiguiente, que más instituciones puedan informarse de los proyectos actuales que se realizan en Bilwi para poder colaborar.
3. Web escalable para permitir la creación, modificación o eliminación de módulos fácilmente sin tener que tocar prácticamente nada del código.
4. Diseñar los distintos módulos que el cliente pedía:
 - **Noticias:** para informar a todos los habitantes de las últimas noticias de la ciudad que a la vez pueda compartirse en distintas redes sociales, ya que actualmente todo se hacía a través de Facebook y querían un portal más serio que permita ambas posibilidades.
 - **Servicios:** donde listar todos los servicios que ofrece la alcaldía a los habitantes, ya que detectaban que la gente desconocía muchos de los servicios que ofrecen y su uso.
 - **Proyectos de cooperación:** para mostrar todos los proyectos que se realizan en la alcaldía y permitir que las distintas instituciones que estén interesadas en alguno de los proyectos puedan colaborar rellenando una inscripción.
 - **Intercambio cultural:** para enviar correos entre escuelas de Bilwi con otras de otros lugares del mundo, ya que des de Bilwi se hacen intercambios de cartas entre los niños de algunos centros con otros de Catalunya, en concreto con Vilafranca del Penedés. Este módulo permitiría el envío de estas cartas directamente por internet.
5. Permitir que la gestión de la web pueda usarse en modo local, esto quiere decir que todo el contenido que hay en ella debe ser off-line. De este modo, el cliente puede acceder al servidor directamente para poder realizar los cambios necesarios en la web incluso cuando no tengan acceso a Internet (hay que tener en cuenta que en países como Nicaragua el acceso a Internet es mucho más limitado debido a factores externos como la climatología o las malas instalaciones de la compañía, por lo que les suele fallar muy a menudo, causándoles durante un tiempo ilimitado la desconexión a Internet).

1.2 Requerimientos y especificaciones

Núm.	Requisito	Comentarios
100	Gestión de los datos con un administrador	
110	Gestionar cada módulo por independiente	
111	Añadir nueva entrada	Incluido en el administrador heredado
112	Modificar cualquier entrada	Incluido en el administrador heredado
113	Eliminar una entrada	Incluido en el administrador heredado
114	Activar/desactivar entrada	Incluido en el administrador heredado
120	Funciones extras del administrador	
121	Galerías para cada módulo y entrada	Personalizar las fotografías des del Administrador
122	Editar la web des del administrador	Personalizar los logos, títulos de la web
123	Administrar los idiomas de la entrada	
130	Administrador multi-idioma	
140	Acceso según usuario	Que los usuarios solo puedan acceder a determinadas entradas, para evitar incompatibilidades.
141	Crear distintos perfiles	
142	Crear permisos para los perfiles	
143	Crear usuarios dentro de un perfil	
200	Módulo servicios	
210	Mostrar listado de servicios	
220	Clasificar los servicios según temática	
300	Módulo noticias	
310	Mostrar todas las noticias	
311	Poder enlazar con distintas redes sociales	
320	Clasificar noticias según temática	
400	Modulo proyectos de cooperación	
410	Mostrar todos los proyectos	
411	Clasificarlos según temática	
412	Clasificarlos según el estado del proyecto	
413	Acceso a un formulario para participar en el proyecto	
414	Mostrar toda la información del proyecto aparte	
420	Mostrar los proyectos en un mapa	
421	Marcadores según temática y estado	
422	Enlazar marcador con descripción	
423	Marcadores personalizables	Función extra
500	Incluir información de la alcaldía	
510	Incluir un listado en el inicio para acceder a cualquier información de la alcaldía	
520	Mostrar contactos	
521	Poder contactar desde la web	
600	Otras funcionalidades de la web	
610	Web multi-idioma	
620	Poder acceder a la web en local	Acceso des de la alcaldía sin necesidad de Internet
621	Librerías externas en local	
622	Librerías del mapa en local	

Tabla 1.1 Tabla de requerimientos y especificaciones de la web

1.3 Métodos y procedimientos

1.3.1 Código usado

Debido a que se trata de una web de diseño nuevo la mayoría de este se ha creado desde cero. Aunque sí es cierto que se han usado librerías de JavaScript ya existentes a la web para implementar funcionalidades ya programadas previamente:

- JQuery: para interactuar con html dinámicamente. En concreto se han usado:
 - AJAX¹ que permite rellenar un contenedor html con distinto contenido (texto, imágenes...) en función de unos parámetros que se indican a través de la función que llama a éste.
 - Slideshow que permite incluir varias imágenes en un solo contenedor mostrándolas en distintas diapositivas. Es una forma muy útil para mostrar muchas imágenes en un contenedor de gran tamaño pero usando solo el espacio de una imagen. De lo contrario, se deberían mostrar las imágenes en tamaño reducido para ocupar el mismo espacio o mostrarlas con el tamaño original y ocupando un espacio innecesario de la web.
- Fancybox: permite mostrar una información en una ventana emergente, en nuestro caso, se ha usado para poder visualizar la galería de imágenes de una noticia y para mostrar la información de un proyecto. De este modo puedes visualizar un contenido concreto sin tener que abrir ningún módulo nuevo (borrando el contenido anterior).

Además de estas funciones JavaScript ya existentes, se han creado de nuevas más concretas, para funcionalidades propias de la web. Entre estas nuevas funciones, cabe destacar las que se han implementado para la generación del mapa y sus complementos en el módulo de los proyectos de cooperación. Para esto se ha usado OSM² (Open Street Map) debido a que:

- es de código libre
- permite editar el mapa, así, si la zona del mapa que te interesa no está correctamente cartografiada, tú mismo puedes modificarlo. Esto es importante ya que hay áreas de los países en desarrollo que no están incluidas en la mayoría de los mapas web
- permite descargarlo para usarlo en modo local

Para la creación del Administrador también se ha usado ya un código existente, para ello se ha usado uno de código libre. Se decidió usar este debido a que todo el código era completamente libre (debido a que ya no está sujeto a ninguna licencia) y permitía la mayoría de las funcionalidades que se quería:

- Introducir datos
- Modificar datos
- Eliminar datos
- Organizar los datos según módulos
- Acceder al administrador con un usuario
- Diseño CSS (Cascading Style Sheets) completo

Todas las nuevas funcionalidades³ que se han creado se comentan más adelante.

¹ AJAX(*A*synchronous *J*avascript *A*nd *X*ML) creada en 2005 por Jesse James Garrett

² OSM creada en 2004 por Steve Coast, en 2006 se crea la fundación OpenStreetMap (www.openstreetmap.org)

³ Nuevas funcionalidades del Administrador en el punto 3. Metodología

1.3.2 Entorno de trabajo

En cuanto al entorno de trabajo que se ha escogido no ha habido muchas limitaciones en cuanto al Hardware, ya que todo el proceso de desarrollo de la web ha sido a nivel Software. Pero si hace falta destacar los servidores compatibles con la web que se ha diseñado.

En primer lugar, durante el desarrollo de la web se ha usado el servidor Apache⁴, los motivos por el que se escogió este servidor fue:

- código abierto
- permite trabajar en local, es decir, solo se puede acceder desde los dispositivos conectados en una misma red IP (Internet Protocol), generalmente la del router.
- procesa aplicaciones hechas con lenguaje web *PHP*(Hypertext Preprocessor) que es el que se ha usado para la programación de la web, por los siguientes motivos:
 - Es un lenguaje del servidor que permite interactuar directamente con *HTML*⁵(HyperText Markup Language) y con las bases de datos.
 - El administrador escogido para la web venía programado en *PHP*, lo cual programarlo todo en el mismo código es mucho más práctico.
- incluye PhpMyAdmin para generar las BD (bases de datos) necesarias para almacenar toda la información de la web.

Teniendo en cuenta todo esto, el dominio necesario para subir todo el contenido de la web ha de admitir el lenguaje *PHP*, ya que no es el único lenguaje de programación del servidor.

⁴ Apache empezó a funcionar en 1995 basado en el servidor NCSA HTTPd

⁵ HTML es un lenguaje web

1.4 Plan de trabajo

El plan de trabajo se organizó distribuyendo todos los requisitos que debe incluir el proyecto en distintas tareas y agrupándolas en bloques para unir las que tienen conexión.

Como se puede ver en el diagrama de Gantt⁶ que se adjunta en los Anexos (Anexo 2. Diagramas), los bloques son los siguientes:

- Estructurar el contenido de la web
Básico para mantener la escalabilidad de esta y para que cualquier modificación futura no afecte a lo que ya existe, sino que solo sea un complemento que incluir fácilmente.
- Administrador (requisito 100)
Para que la web pueda ser extensible a más instituciones, con necesidades parecidas y con pocos recursos económicos, se requiere gestionar la mayoría de la información en un administrador externo.
- Bases de datos
Para almacenar todo el contenido que se incluye en el administrador y que la web pueda gestionarla fácilmente es necesario estructurarlas previamente
- Crear cada uno de los módulos que necesita el cliente:
 - Servicios (requisito 200)
 - Noticias (requisito 300)
 - Proyectos (requisito 400)
 - Módulo principal con acceso a la información del cliente (requisito 500)
- Traducir (requisito 610)
Para que los elementos fijos de la web estén traducidos automáticamente, existe un módulo interno de la web con estos textos y sus traducciones en los distintos idiomas
- Diseño gráfico
A pesar de que este proyecto sólo se centra en la parte técnica que hay detrás de la creación de una web, se ha intentado crear un diseño acorde a las funcionalidades tanto del cliente como del usuario. Para ello se ha dedicado una parte de las horas en el diseño gráfico para captar la atención del usuario.
- Detección de errores
Este es un punto importante del proyecto, ya que hay que asegurarse que la web funciona perfectamente en cualquier entorno (Safari, Chrome, Mozilla, Explorer y dispositivos móviles).

Como se han desarrollado cada uno de estos bloques y tareas se explica más adelante, en el apartado de Metodología.

⁶ Diagrama de Gantt, en el Anexo 2. Diagramas en la tabla 9.2

1.5 Desviaciones del plan inicial e incidencias

1.5.1 Plan inicial

Uno de los grandes cambios que han surgido al largo del proyecto ha sido la extensión de la fecha de entrega de Enero a Mayo. Se decidió hacerlo así debido a que durante el cuatrimestre de Otoño (de setiembre del 2016 a enero del 2017) estuve realizando prácticas en empresa y una asignatura obligatoria del grado, por ello, no pude dedicar todo el tiempo que pensaba antes de empezar el curso. Así pues, los tiempos dedicados a cada tarea se ampliaron.

1.5.2 Incidencias

El seguimiento de la planificación inicial ha sufrido cambios de tareas y temporización debido a las distintas incidencias que me he encontrado durante el desarrollo de la web.

En un principio se tuvo en cuenta que la programación era, en gran mayoría, nueva para mí, por ello se dedicó un tiempo en documentación y práctica de esta. A pesar de ello, acabó siendo un limitante más adelante ya que la web implicaba necesidades más complejas.

- En concreto, el gran limitante fue el Administrador heredado de otra aplicación, ya que tenía una estructura y unas funciones desconocidas para mí y fue todo un laberinto descubrir las conexiones que había entre todos los módulos y clases. Otro limitante de este fue el acceso a las bases de datos (BD), ya que accedía de modo diferente a como había programado yo las BD, esto implicó una tarea extra en descubrir todas las diferencias entre unos y otros. Por lo que limitó tanto a la estructuración de la web como a la ampliación del administrador.
- Otro gran limitante al largo del proyecto ha estado la implementación del mapa en el módulo de los proyectos de cooperación. Se eligió usar el mapa OSM ya que permitía la edición de este y porque al ser de código abierto es posible usar sus librerías para usarlo en local (off-line). Debido a su reciente crecimiento y empezar a darse a conocer en el entorno web, está apareciendo documentación de ayuda que facilita su manejo. A pesar de ello, no es suficiente, ya que el gran competidor es Google Maps, el cual sigue siendo el más usado y más documentado. Debido a esta falta de documentación y facilidad para acceder a los archivos, que permiten desarrollar el mapa OSM en local, el desarrollo del módulo de los proyectos de cooperación se estancó más de lo esperado.

Otras incidencias que he tenido al largo del proyecto han sido a nivel del entorno de trabajo.

- A pesar de tener todo el código en un disco duro para evitar cualquier tipo de incidencia (y generar copias diarias), a mitad del proyecto el ordenador que he usado de trabajo se bloqueó, lo que implicó formatearlo y volver a instalar todo el entorno de trabajo más volver a instanciar todas las bases de datos que se usaban en los distintos módulos de la web. Esto no afectó a las tareas ni la programación, pero sí a los tiempos de trabajo, ya que tuve que estar un par de semanas en volver a configurarlo todo tal y como estaba.
- Para acceder a la web en modo local hay que configurar unos archivos para que el programa sepa a que IPs acceder para conectarse a la web. Pero a finales del proyecto, el puerto IP que accede a Apache se desconfiguró (a pesar de estar bien puesto en el archivo de configuración). Para esto fue necesario volver a instalar el Apache y modificar los antivirus del ordenador para que no bloqueasen el puerto. Esto afectó que durante un par de días no pudiese dedicarme al contenido de la web para poder arreglar la configuración de los puertos.

2. State of art

En cuanto al mundo web está claro que ya existen muchas plataformas en la red con grandes ventajas, pero también desventajas que suelen estar ligadas al precio (recordemos que este proyecto se basa en un proyecto de cooperación, para instituciones que necesitan darse a conocer a través de una web del modo más económico posible).

Previamente, antes de empezar con este proyecto, se analizó todo lo que ya existía en Internet que pudiese aplicar las necesidades que pedía el cliente y satisfacerlas todas. Para ello se estudiaron los tres CMS (Content Management System, en español Gestores de Contenido) “gratuitos”⁷ que más se adaptaban a esto: Blogspot, Wordpress y Joomla. Antes de analizarlos independientemente, destacar que actualmente están saliendo nuevos portales como Wix, Weebly, etc. que están poniéndose a la altura de las anteriormente enunciadas, pero no vamos a hablar de ellos ya que no están todavía muy documentadas y las extensiones que incluyen actualmente no distan de las que vamos a comentar a continuación.

Entre los tres escogidos, vamos a analizar al detalle Wordpress y Joomla. En cuanto a Blogspot no vamos a hacer un análisis exhaustivo, debido a que es un portal web con una estructura de diseño bastante limitada y centrada en la publicación de varios posts (una entrada de texto ordenada según fecha de publicación). Es una muy buena opción para publicar distintos cuadros informativos de manera gratuita, pero los requisitos del cliente son más extensos, por lo cual no nos encaja en este portal web. Todo esto en cuanto a contenido, no hace falta comentar que en cuanto a los requisitos de multi-idioma, galería, etc. también nos limita como veremos a continuación con los otros dos portales.

2.1 Wordpress

Hay tres opciones de trabajar con este servidor web:

- Wordpress.org: es la versión de código abierto, el cual puedes descargar y modificar el código a tu gusto en función de los requisitos que necesite la web. Tiene un límite de almacenamiento a 3 GB.
- Wordpress.com
 - Gratuito: permite crear una especie de portal blog en el cual puedes elegir el diseño básico de tu web y organizar la información que necesitas en distintos módulos. Esta opción incluye un almacenamiento 3 GB. Esta opción es muy parecida al uso de Wordpress.org solo que este te genera un dominio del tipo miweb.wordpress.com, mientras que el otro te exige que contrates tu dominio donde colgar la web.
 - Premium: su uso es parecido al anterior, pero incluye todos los temas de diseño, personalizar el diseño a tu gusto, servicio de ayuda, enlaces en redes sociales y almacenamiento de 13 GB. Hay otras opciones de pago, entre ellas la opción Business con alguna funcionalidad más, pero que no vamos a tener en cuenta ya que no nos interesan.

En cuanto a la primera opción, wordpress.org, descargas el esqueleto de la web con un tema básico de diseño, pero a partir de aquí has de programar todo el contenido web, por lo que no nos aporta ninguna ventaja.

⁷ Todos incluyen una versión gratuita básica, pero para tener un dominio personalizado e incluir extensiones es necesario pagarlo. Todo esto se analiza en este apartado.

Así pues, vamos a analizar la segunda opción, wordpress.com, la cual ofrece toda una serie de extensiones que también se aplican en wordpress.org. Este análisis lo vamos a hacer desde el punto de vista económico, ya que éste es el gran limitante del proyecto, se trata de hacer una web al mínimo coste posible e incluyendo todos los requisitos.

De este modo, veamos qué diferencias hay entre usar wordpress.com y programar tu propia web:

1. Seguridad: cuando trabajas con Wordpress toda la seguridad está a manos de la empresa, que significa esto, si trabajas sobre la versión gratuita y tienes algún ataque hacker la empresa no se hace cargo, por lo que tú debes asumir las consecuencias y buscar una solución. Si trabajas sobre Wordpress Premium sí que puedes exigir una solución por parte de la empresa.
En cuanto a la creación de una nueva web, tú pones tus barreras de seguridad en la web, si es cierto que en caso de recibir un ataque la única solución eres tú mismo, pero si eres ingenioso es muy probable que el hacker no sepa detectar tu punto débil. No voy a desvelar mis barreras, como es obvio, pero una opción muy fácil y segura es poner una clave de control de acceso en las BD.
2. Diseño web: en Wordpress existen temas de diseño muy llamativos, pero te limitan a la distribución de los elementos que hayas seleccionado previamente con el tema. También hay que destacar que los temas que incluye la versión gratuita suelen ser demasiado básicos, poco llamativos para el usuario, por lo que muchas veces implica echar mano de la programación, que es justo lo que estamos evitando con Wordpress.
En cuanto a programar tú mismo la web, el diseño es 100% programable en función a tus gustos, modas y/o necesidades. Sí es cierto que para hacer un diseño bonito hacen falta conocimientos altos de CSS y en edición de imágenes (crear logos atractivos y que gusten a los usuarios)
3. Funcionalidad de la web: para adaptar la web en función de los requisitos del cliente la versión gratuita de Wordpress queda muy limitada, por lo que a veces es necesario ir al código y hacer alguna modificación. Por lo que estamos en el mismo punto que antes, tenemos que pagar para evitar tener que programar.
4. Acceso a redes sociales: en la versión Premium viene incluido poder crear enlaces a estas, mientras que en la versión gratuita hay que instalar un plugin (de pago).
En programación, se pueden crear scripts que enlacen en distintas redes sociales para compartir un enlace de tu web directamente.
5. Multi-idioma: hay extensiones para Wordpress que permiten incluir la opción de seleccionar el idioma en la web de modo gratuito, el único inconveniente es que la traducción del contenido se hace a través de un traductor on-line, lo que puede generar errores en la escritura de este. Sí es cierto que alguna de estas extensiones incluyen un portal para detectar los posibles errores de traducción para mejorar estos Plugins. Además, el Plugin te resulta gratis siempre y cuando no superes un límite de palabras, que no suele ser demasiado generoso (un máximo alrededor de 2000 palabras).
La ventaja de programar el multi-idioma en tu propia web es que todo lo traduces tu manualmente, lo que implica menos errores (se entiende que la persona que traduce los textos domina la lengua en cuestión) y en caso de cometer algún error la modificación es directa, no dependes de ninguna aplicación externa.
6. Velocidad de acceso: a pesar que este es uno de los puntos fuertes que te venden en muchas plataformas web como Wordpress, la velocidad de acceso en una web depende, en gran parte, del host que contratas para la web, de modo que no depende del portal que selecciones (Wordpress, Blogspot o incluso tu propia web), sino de contratar un host compatible con tu código. Sí es cierto que hay hosts que casan muy bien con Wordpress, lo que permiten que la velocidad de acceso sea rápida en la mayoría de los casos.

En definitiva, este punto no nos afecta en seleccionar una opción u otra, ya que para ambos casos no depende de la selección de la aplicación web sino del host que contrates para tu dominio.

7. Actualizaciones automáticas: todo el código Wordpress usa las versiones más nuevas de las librerías que incluyen y miran que no haya ningún problema de compatibilidad, lo que permite que la web esté actualizada en todo momento y no te tengas que preocupar.

En una web programada manualmente, es necesario estar al tanto de estas actualizaciones y de su correcto funcionamiento.

Por ello, podríamos decir que este es un punto a favor de Wordpress, ya que la plataforma que hay detrás se encarga de estas actualizaciones, mientras que en una web programable has de tener en cuenta que haya alguien que se encargue de este mantenimiento, sino es posible que caiga en desuso.

8. Plataforma grande, visibilidad: al ser una plataforma con tanto éxito, es mucho más fácil que un usuario acceda a una web Wordpress que no otra sin ninguna referencia. Sin duda, esto es una gran ventaja si creas una página web a través de Wordpress, aunque hay que remarcar que tienen mucha más visibilidad las páginas Premium e incluso hay Plugins de pago para mejorar el contenido de la web y obtener mejores resultados de búsqueda (Herramientas SEO⁸).

En resumen para Wordpress, tanto si quieres incluir algún Plugin (código para ampliar el contenido de la web) o Widget (plugins del tipo visual, para mostrar iconos o fotografías con diferentes estilos) que complemente la web has de pagar por ellos. En la mayoría de estos, el pago te da el derecho de usarlos para siempre, pero solo te permiten acceder a las actualizaciones y al soporte de estas durante un año desde que las descargas (pagando), en caso de querer acceder a las actualizaciones y al soporte, se debe volver a realizar un pago. Por lo que incluir todas estas ventajas que te ofrece Wordpress pueden resultar muy caras para instituciones con muy pocos recursos.

2.2 Joomla

Joomla es otro Gestor de contenido muy parecido a Wordpress. Las diferencias entre ambos es que Wordpress es una plataforma mucho más documentada y más fácil de retocar el código en función de las necesidades de la web. Por otro lado, incluye una cantidad de extensiones más numerosa en las que puedes elegir entre una versión gratuita o de pago. Para ello se han investigado los distintos Plugins que existen para los requisitos de nuestra web:

1. Multi-idioma: Joomla es una plataforma que incluye directamente esta funcionalidad, por lo que solo es necesario acceder a la web de Joomla para seleccionar el idioma que deseas incluir en la web. Del mismo modo que hace Wordpress, este traduce el contenido a través de un traductor on-line y solo traduce el contenido que se introduce de nuevo una vez instalado el complemento.
2. Mapa: permite incluir un Google Maps, pero para poder interactuar con este con marcadores y otros complementos es necesaria la versión de pago. Además de que no hay ningún plugin que permita insertar OSM ni ningún mapa que permita trabajar con él off-line.
3. Redes sociales: también existe un plugin gratuito para poder crear accesos a redes sociales, pero el acceso a estas es limitado (no se puede poner en cualquier sitio web...). Por lo que la versión gratuita ya es bastante buena para los nuestros requisitos pero no nos permite actuar con las redes sociales libremente.

⁸ SEO (Search Engine Optimization), es el proceso técnico para optimizar la información de la web para obtener un buen posicionamiento en los buscadores de Internet y así hacer que sea más visible.

4. Galerías: existen muchos complementos que permiten mejorar la visualización de las imágenes, pero estas limitan su uso (cantidad de imágenes, una estructura y un diseño fijo...)

En general, la incorporación de estas extensiones es compleja de adaptar a la plataforma Joomla ya creada. Además, todas las extensiones gratuitas, solo son aplicables durante un tiempo limitado, lo que no nos interesa, ya que queremos que el contenido de la web sea lo más estable posible. Por otro lado, investigando los diferentes complementos, he observado que algunos ya están obsoletos, por lo que se debería estar al tanto de las actualizaciones de las extensiones, lo cual dificulta la implementación de estas a largo tiempo.

Como vemos, el uso de Joomla parece que se adapte mejor a nuestros requisitos, pero aun así son demasiado limitados y deberíamos acabar optando por las versiones de pago. Sin contar que seguramente implicaría mucho más tiempo para combinar la plataforma web Joomla con las extensiones elegidas para complementarla.

2.3 Elección de CMS

En conclusión, tanto en Wordpress como en Joomla, si queremos tener las siguientes funcionalidades, hay que pagar:

- Multi-idioma, a pesar de que existen varios Plugins para ambas que funcionan bastante bien y cumplen con su función (poder seleccionar los idiomas y traducir el contenido)
- Compartir un enlace en redes sociales
- Un diseño web con un cierto atractivo y poderlo modificar según la funcionalidad que interesa para cada módulo
- Seguridad
- Incluir mapa dinámico que permita cambios en función de las distintas propiedades de los proyectos creados

Destacar también que para acceder a los Administradores de ambos servidores, es necesario disponer de acceso a Internet, recordemos que, en las áreas geográficas en las que se va a instalar esta aplicación, es muy común que el acceso a Internet falle debido a una instalación del sistema, velocidad de Internet reducida, según la climatología y otros factores. Por lo que diseñar un Administrador que se pueda acceder en cualquier momento (tanto en modo off-line como on-line) es una gran ventaja, ya que permite trabajar en la web en cualquier momento, teniendo en cuenta que los resultados solo se verán cuando vuelvan a tener acceso a internet y se actualice la BD del host.

Teniendo en cuenta que queremos cumplir con la mayoría de los requisitos y con el menor coste posible, la mejor opción es crear una web programada totalmente de nuevo.

3. Metodología

Todo el desarrollo del proyecto se ha desarrollado teniendo en mente el que he considerado el punto central estratégico, la **escalabilidad**. Este punto es básico a la hora de crear una web para garantizar que cualquier modificación que se quiera hacer en la web no afecte al código que ya existe. Además, en nuestro caso, también ha sido importante este punto para poder manejar la máxima cantidad de información desde el Administrador y que la web sea un esquema fácil de moldear. Esto ha sido así, ya que se desea poder facilitar esta web a todas esas áreas geográficas con pocos recursos económicos pero muchas ganas de mejorar y que ellas puedan personalizar el contenido de la web en función de sus necesidades.

Otro punto muy importante ha sido generar una web que sea **sencilla** de usar. En primer lugar, fácil de gestionar los datos desde la alcaldía de Bilwi, ya que, a pesar de contar con un informático en el equipo, no todos tienen los conocimientos de informática que hoy en día tenemos, por eso es necesario que la introducción de datos sea lo más fácil y evidente posible. Además, también se ha querido diseñar una web que sea atractiva y dinámica para el usuario, ya que por un lado queremos que sea fácil para el habitante de Bilwi poder encontrar la información de su ciudad (servicios y noticias) y también queremos atraer la atención de las posibles instituciones que puedan colaborar con los proyectos. Por eso se ha querido hacer un diseño muy moldeable, como vemos, esto nos enlaza de nuevo con la escalabilidad que comentábamos.

Finalmente, el último punto que se ha querido garantizar desde un inicio ha sido la **seguridad** de los datos, ya que al tratarse de una aplicación web puede ser que otras personas accedan a la base de datos para manipular los datos que hay en la web. De nuevo, para conseguir que los datos de nuestra web sean seguros es tener la información lo más independiente entre ella posible, si el acceso a ella está distribuido en varias capas (clases y funciones), reconstruir el camino para acceder a ella es mucho más complejo. Para garantizar la seguridad también se ha echado mano de claves encriptadas.

Por todo ello, antes de empezar a programar y generar el contenido de la web, se dedicó una gran parte de tiempo en crear un prototipo firme de la web y generar un diseño sobre papel consistente y que permitiese estos puntos: sencillez y seguridad, pero sobretodo, escalabilidad. Así pues, veamos cómo se diseñó toda la estructura de la web desde un inicio y como se ha ido implementando.

Hay que remarcar que cuando hablamos de diseño nos referimos a la estructura de la web, es decir, como gestionar todo el código en diversas clases y como éstas interactúan entre ellas. Todo el diseño gráfico, es decir, el estilo con una coherencia lógica de colores, letras y tipografías no se estudió hasta finales del proyecto, ya que consideramos que es una parte importante en vista del usuario pero no en cuanto al contenido de la web, que es en lo que realmente nos vamos a centrar.

3.1 Esquema de la web

La estructura básica, que se propuso en un principio y se ha mantenido hasta hoy y sobre la que se ejecuta toda la web, es la siguiente:



Ilustración 3.1 Estructura básica de la web (dos elementos fijos y el contenido del medio dinámico)

A simple vista puede parecer demasiado sencillo, pero este es el truco de realizar una web lo más escalable y modulable posible, crear una estructura básica sencilla sobre la que se van montando el resto de componentes del mismo modo, una especie de fractal, simples figuras que se van haciendo más complejas a modo que nos vamos acercando.

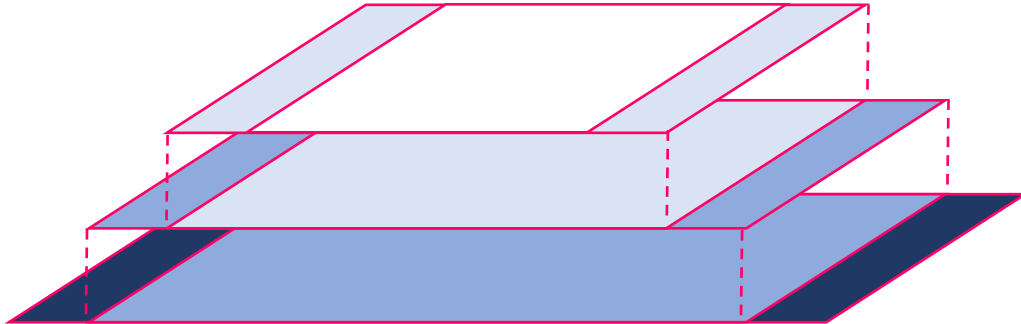


Ilustración 3.2 Esquema de escalabilidad

Así pues hemos dejado fijos la cabecera y el pie de página y los módulos se van rellenando en función de los módulos que se seleccionan.

- Cabecera: todos los elementos que queremos que sean accesibles en todo momento para poder acceder a cualquier portal fácilmente.
- Pie de página: el resto de información extra que se quiere mantener en toda la web pero que no es primordial.
- Módulos: todo el contenido que irá cambiando en función del portal al que queramos acceder.

Veamos cómo se refleja esto en el código:

```
<?php include('config.php');  
Frontend::get_header();  
echo $mensaje == '' ? '' : $mensaje.'  
echo Frontend::accio();  
Frontend::get_footer();  
?>
```

Este es el código que se ejecuta cuando entramos en la página web (en el índice que ejecuta la página y sus enlaces) www.midominio.com/index.php

Como vemos solo crea 4 líneas de código las cuales cada una se encargan de gestionar el contenido de estas, justamente crea lo que hemos comentado anteriormente: cabecera, la acción que gestiona los distintos módulos y el pie de página. La primera línea de código incluye una clase externa (`config.php`) para que el `index.php` sepa donde tiene los archivos a los que acceder.

Como también se puede ver, la clase que realmente gestiona estos tres elementos es la clase `Frontened.php`. Veamos cómo funcionan estas tres funciones con **pseudocódigo**.

1. `Frontened::get_header();`

```
public static function get_header(){
//Incluimos CSS y JS externos (generales y por módulo)
    $_CSS = FP::includeCSS();
    $_JS = FP::includeJS();

//Creamos todo el <head> de la web, donde se incluye: dominio web, título
ventana, keywords de búsqueda, descripción de búsqueda y se incluyen los CSS y
JS que hemos enunciado antes.
    $_TITLE = FP::titol();
    $html = FP::head();

//Por último, se crea el navegador superior que se verá en la página web. En
este hemos incluido el listado con todos los módulos que podemos visitar y la
selección de idiomas:
    $html .= '<body>';
    $html .= '<html lang="es">
        <header>';
    $html .= '<nav rol="nav" id="menu">';
        $html .= FP::listadoModulos();
    $html .= '</nav>';
    $html .= '<nav class="idiomes">';
        $html .= FP::print_idiomes();
    $html .= '</nav>';
    $html .= '</header>';

    echo $html;
}
```

Como podemos ver, esta parte usa funciones que se crean en una clase externa FP (`funcions_publicques.php`) para que el `frontened` simplemente se encargue de gestionar las tres secciones que se crean en `index.php`, mientras que todas las funciones necesarias (todo el código) están en otra clase. De este modo, por ejemplo, si queremos añadir un idioma nuevo solo hay que ir a la función `print_idiomes()` de la clase `funcions_publicques.php` e indicarlo en el listado que genera esta, así de vista al `frontened` esto no tiene ninguna relevancia.

El resultado de la cabecera es el siguiente, el diseño se ha hecho con CSS, pero no lo vamos a comentar aquí ya que no es la base de este proyecto:



Ilustración 3.3 Cabecera de la web (elemento fijo)

2. `Frontened::accio();`

Esta función es muy sencilla, simplemente se encarga de llamar el módulo que el usuario ha seleccionado y ejecuta la función que crea el contenido de la web. Por norma general se llama a la función `l·listar($parametro='')`, ya que esta contiene todo el main de la web. Aunque es posible que se ejecute alguna otra función, como por ejemplo, si se quiere mostrar una noticia concreta, en este caso se ejecuta `mostrarNoticia($id)`. En ambos casos, siempre se hace a través de `accio()`.

```
private static function crearAccio(){
//Si no se ha seleccionado ningún módulo es que se ha entrado por primera vez
en la web, por lo que se muestra la portada.
```



```
//En caso de que se haya seleccionado un módulo, el valor se recoge a través
de $_SESSION
    if ( !$_SESSION['modul'] ) $retorna = self::portada();
    else{
        $modulo = $_SESSION['modul'];
        $funcion = $_SESSION['funcio'];
        $parametro = $_SESSION['identitat'];
//Se comprueba que el módulo está instanciado y que la función existe. En caso
que el módulo o la función sean erróneos se notifica en la web
//También se comprueba si se ha pasado un parámetro o no.
        if (empty($funcion)){
            if (!$parametro) $retorna = $modulo::$funcion();
            else $retorna = $modulo::$funcion($parametro);
        }else{
            $retorna = FP::notifica('Sección no encontrada.');
```

En esta función vemos el uso de `$_SESSION`, esta es una variable global propia de *php*, los valores que se introducen en este array se guardan en el archivo de *Session* del ordenador automáticamente, para que estas variables sean fijas (mientras no reciban un nuevo valor), independientemente de lo que se esté haciendo en la web. Para usarla solo hay que indicarlo con `session_start()` e indicar que parámetros tendrá este vector, esto se hace en `config.php` que se incluye en `index.php`. Como podemos ver en la función `accio()`, se usan tres parámetros: 'modul', 'funcio' e 'identitat'. El primero recoge el nombre del módulo seleccionado, el segundo que función de este módulo se quiere y el tercero el parámetro que se le pasa a la función (si es que necesita un parámetro).

Los resultados finales de todos los módulos que se han diseñado se muestran en el 'Anexo 1. Manual de Usuario'.

3. Frontened::get_footer();

```
public static function get_footer(){
    $html = '<footer>';
//Primero se incluyen las redes sociales de la alcaldía para que la gente
pueda acceder a sus canales e interactuar a través de estas.
    $html .= '<div class="redesSociales">';
    $html .= FP::listadoRedes();
    $html .= '</div>';
//Después se incluye un enlace para rellenar un Formulario para que la gente
pueda contactar por correo con Bilwi. Este crea un enlace al módulo de
contacto.
    $html .= '<div class="contacto">';
    $html .= FP::formulario();
    $html .= '</div>';
//Finalmente se crea un pie de página con información de los impulsores de
esta web:
    $html .= FP::pieInformacion();

    $html .= '</footer>';
    $html .= '</body>';
    $html .= '</html>';

    echo $html;
}
```

El resultado de este en la web es el siguiente:

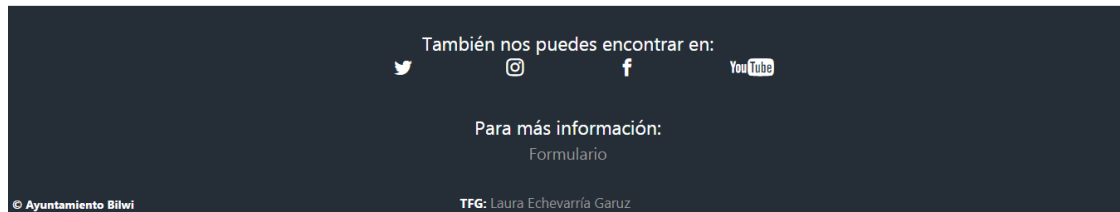


Ilustración 3.4 Pie de página de la web (elemento fijo)

3.2 Mapa OSM

El módulo de los proyectos es el que hemos tratado de hacer más práctico, visual y dinámico para, por un lado atraer a los posibles instituciones que quieran colaborar con Bilwi y que vean de manera muy visual la oferta de proyectos que hay y, por otro lado, para que la gestión de estos proyectos por parte de Bilwi sea lo más sencillo posible.

Por eso se decidió mostrar todos los proyectos sobre un mapa, cada proyecto sería un marcador y al seleccionarlo se mostraría todo el contenido de éste. Pero, ¿cómo ponemos este mapa?

El mapa más conocido en todas las aplicaciones webs es **Google Maps**, ya que insertarlo en una web es muy fácil ya que Google te da permiso a las librerías que gestionan todo el mapa. Además, como es un mapa muy conocido está muy bien documentado y es muy fácil interactuar con él. Hasta aquí todo era muy práctico, pero había dos problemas:

- **Edición del mapa:** al analizar bien el mapa de Bilwi, pudimos ver que había zonas de Bilwi que no se reflejaban en el mapa, en especial, las aldeas que están en torno de la ciudad y las cuales son las que necesitan más ayuda. Por eso es necesario poder editar estas zonas fácilmente para que aparezcan en el mapa de la web, pero esto en Google Maps es complejo, ya que lo gestiona la misma compañía de Google.
Además, esta web está pensada para implementarla en otros poblados que quieran informar de sus problemáticas para que puedan colaborar con ellos. Así que tener actualizado el mapa debería ser lo más fácil posible.
- **Mapa local:** poder acceder al mapa off-line es una ventaja para la alcaldía (o las futuras organizaciones que puedan usar esta web) para poder gestionar todos los proyectos en cualquier momento. Debido a que los cortes en la red de Internet en estas zonas son mucho más comunes que aquí es una gran ventaja poder acceder a todo el contenido de la web en local. De este modo, pueden modificar los datos y gestionarlos en cualquier momento.

Por estos dos motivos, decidimos buscar un mapa que incluyese estas dos propiedades. Buscando alternativas encontramos OSM, el cual la principal ventaja es que el mapa se genera a partir de la información que introducen los usuarios, por ello, cuando se detecta que el mapa no corresponde con la realidad el usuario puede editarlo como si fuese una imagen (añadiendo las calles, accidentes geográficos...). Hoy en día, este proyecto ha crecido mucho, por lo que empieza a implementarse en otros sitios web y por lo que el mapa tiene la mayoría de las áreas del mundo completadas.

Además, OSM permite descargar las librerías de la web para poder usarlo en local. Así que por ese motivo elegimos usar este mapa. Algunas de las librerías que insertan OSM son: *OpenLayers* y *Mapbox*. Se estudiaron estas dos ya que ambas son librerías de JavaScript.

Optamos primeramente por **Mapbox** ya que es una plataforma que usan otras aplicaciones como Uber y estaba bien documentado, se empezó instanciando el mapa e incluyendo los marcadores, estos se programaron en función de los proyectos seleccionados. Así pues, se insertó Mapbox (con OSM) y a partir de este se miró como descargar todos los archivos y vectores necesarios para que fuese funcional en local, pero todas las librerías hacían referencia a algún archivo en internet, lo que complicaba el código y la búsqueda de estos.

Así pues, se decidió por investigar el caso de **OpenLayers**. Este es un código más fácil de manejar, ya que directamente es una librería de JavaScript de código abierto. Su inserción fue muy parecida a las dos anteriores:

- Para crear el mapa solo hace falta indicar las propiedades de este (coordenadas, zoom...) e instanciar la librería que crea todo el mapa
`var mapnik = new OpenLayers.Layer.OSM();`
- Para crear los marcadores debemos indicar las propiedades de este (coordenadas, icono del marcador, tamaño...) y añadir este marcador al mapa
`marker = new OpenLayers.Marker(position, icon);`
`markers.addMarker(marker);`

Luego quedaba descargar todas las librerías que hacían posible el uso del mapa en local. Para ello se buscó en los repositorios de GitHub hasta que finalmente se encontró un proyecto con todos los archivos en local de OpenLayers:
















 apidoc_config	04/04/2017 19:55	Carpeta de archivos	
 art	04/04/2017 19:55	Carpeta de archivos	
 build	04/04/2017 19:55	Carpeta de archivos	
 doc_config	04/04/2017 19:55	Carpeta de archivos	
 img	04/04/2017 19:55	Carpeta de archivos	
 lib	04/04/2017 19:55	Carpeta de archivos	
 notes	04/04/2017 19:55	Carpeta de archivos	
 theme	04/04/2017 19:55	Carpeta de archivos	
 .gitignore	31/03/2017 17:14	Archivo GITIGNORE	1 KB
 license	31/03/2017 17:14	Documento de tex...	2 KB
 OpenLayers.debug	31/03/2017 17:14	Archivo JS	2.874 KB
 OpenLayers	31/03/2017 17:14	Archivo JS	753 KB
 OpenLayers.light.debug	31/03/2017 17:14	Archivo JS	1.224 KB
 OpenLayers.light	31/03/2017 17:14	Archivo JS	286 KB
 OpenLayers.mobile.debug	31/03/2017 17:14	Archivo JS	1.397 KB

Ilustración 3.5 Librerías de OpenLayer para acceder al mapa en local (librerías JavaScript y gráficos)

Gracias a la obtención de este repositorio por fin pudimos instaurar el mapa OSM en local. A pesar de que el gráfico del mapa también se puede poner en local, se ha decidido no optar por ello ya que sería necesario actualizarlo a menudo (para que puedan visualizarse todas las actualizaciones), además de que puede llegar a ocupar hasta GB de memoria, lo que implica que poder actualizar estos datos en un lugar como Bilwi (donde la calidad de Internet por ahora no es muy buena) es inviable.

El resultado del mapa (con conexión o en local) es el siguiente:



Ilustración 3.6 OSM con conexión a internet



Ilustración 3.7 OSM sin conexión a internet (off-line)

Como podemos ver en las imágenes anteriores, la gran diferencia es que no se ve el mapa. A pesar de ello, los marcadores se sitúan en las mismas coordenadas y al clicar se abre el enlace con toda la información de estos.

Ya por último, para la mejora de la visualización de los marcadores en el mapa, se ha creado una función que une los logos del estado del proyecto (color del marcador) con el icono de la temática.

```
static private function crearMarcador($iconoTema, $iconoEstado, $rutaDestino){  
    //Creo imagen destino png  
    $dest = imagecreatetruecolor(500,500);  
    imagesavealpha($dest, true);  
    $trans_background = imagecolorallocatealpha($dest, 0, 0, 0, 127);  
    imagefill($dest, 0, 0, $trans_background);  
  
    //Imagen del tipo de proyecto y color de marcador según el estado  
    $estado = imagecreatefrompng($iconoEstado);  
    $tema = imagecreatefrompng($iconoTema);  
  
    //Copio las imagenes en la destino  
    imagecopy($marcador, $estado, 0, 0, 0, 0, 500, 500);  
    imagecopy($marcador, $tema, 0, 0, 0, 0, 500, 500);  
}
```

```
//Guardo la imagen resultante
imagepng($dest, $rutaDestino);

//Destruyo las imágenes
imagedestroy($tema);
imagedestroy($estado);
imagedestroy($marcador);
}
```

3.3 Multi-idioma

Para generar el multi-idioma tanto para la web como para el administrador Situm se han seguido dinámicas muy parecidas. Hay que destacar que el multi-idioma afecta a dos casos distintos:

- El contenido fijo de la web: títulos y descripciones que siempre se muestran en la web, estos textos se fijan en los distintos idiomas, en función del idioma que se seleccione se muestra uno u otro.
- El contenido dinámico de la web: este es el que se almacena en la BD, por ello, este contenido ha de estar en la BD en los distintos idiomas y correctamente identificados.

3.3.1 Contenido fijo

Para todos aquellos contenidos de la web que ya vienen fijos des de código (títulos, pequeños fragmentos de texto, errores, ventanas informativas...) lo que se ha hecho es crear una función que los traduzca, el procedimiento es el que sigue: todos estos textos fijos se han introducido en el idioma predeterminado, que en este caso es el Español y, posteriormente, se indica que deben ser traducidos en caso que se seleccione otro idioma a través de la función `t()` de la clase `funcions_publicues.php`:

```
$retorna = FP::notifica(FP::t('Sección no encontrada.'));
```

Para ello, por cada idioma que incluye la web hay un vector de traducción `$_IDIOMA_ACTIU` este vector tiene como primer parámetro el texto en español (texto predeterminado que incluye la web) y como segundo parámetro la traducción en el idioma correspondiente, veamos un fragmento de este para el idioma de catalán (`ca.php`):

```
$_IDIOMA_ACTIU = array(
//Menú superior de la cabecera
'Servicios' => 'Serveis',
'Portada' => 'Portada',
'Noticias' => 'Notícies',
'Proyectos' => 'Projectes',
//...
)
```

Pero, ¿cómo sabe el programa qué vector debe ir a buscar? Esto se hace desde la clase `config.php`, donde la variable `$_SESSION` incluye en el parámetro `idm` (id del idioma) el idioma seleccionado por el usuario, en caso que no se haya seleccionado ningún idioma se escoge el idioma predeterminado, en este caso el Español (es). Si se ha seleccionado algún otro idioma se recoge a partir de la variable `$_GET`, esta variable es una variable global de *php*, esta recibe el valor de que se envía a partir de `$_POST`, en nuestro caso esta se envía desde el navegador de idiomas que se crea en la cabecera que hemos visto antes (se envía en `FP::print_idiomes()`).

```
if (!isset($_SESSION["idm"])) {
    $_SESSION["idm"] = 'es';
}
elseif (isset($_GET["idm"])) {
```

```
$_SESSION["idm"] = $_GET['idm'];  
}  
if($_SESSION['idm'] != 'es'){  
//Si el idioma no es el predeterminado, se llama al vector con las  
traducciones ($_IDIOMA_ACTIU)  
include 'idiomes/' . $_SESSION["idm"] . '.php';  
}
```

Finalmente, veamos cómo se traducen los idiomas a partir de la función `t()`:

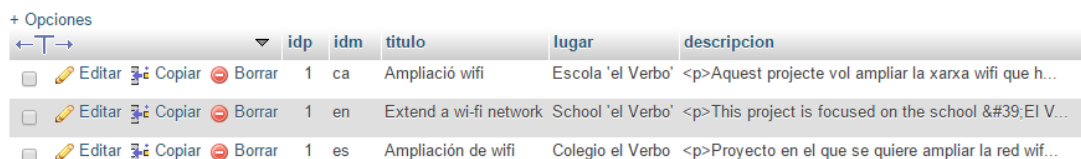
```
public static function t($cadena){  
// Se llama a la traducción del objeto que recibimos (texto a traducir)  
global $_IDIOMA_ACTIU;  
//Si existe el idioma (idioma.php) y la cadena que se ha recibido está  
incluida en este vector entonces se recoge el parámetro con la traducción  
if(is_array($_IDIOMA_ACTIU) && array_key_exists($cadena, $_IDIOMA_ACTIU)){  
return (!empty($_IDIOMA_ACTIU[$cadena])) ? $_IDIOMA_ACTIU[$cadena]:$cadena;  
}else{  
return $cadena;  
}  
}
```

Estas funciones son las que se usan en la aplicación web, pero para el administrador *Situm* se ha hecho exactamente lo mismo, ya que los idiomas son los mismos. Lo único que varía es el vector con las traducciones de los contenidos fijos del administrador, en este caso, el vector es mucho más extenso ya que todos los listados se incluyen manualmente desde las clases y funciones del administrador, mientras que en la web es menor ya que la mayoría del contenido de ésta se recoge de la BD.

3.3.2 Contenido BD

Este es el que introduce el usuario del administrador *Situm* para que se pueda añadir en la web, por eso, en este caso tenemos que traducir los elementos de la BD, como es evidente, el método que hemos usado anteriormente aquí no tiene ningún sentido, ya que antes teníamos un vector con elementos fijos mientras que ahora son dinámicos. Por eso mismo, la traducción se ha de incluir en la misma BD, si un elemento en español se modifica, también debe hacerlo el de inglés y catalán.

Por ello lo que se decidió fue crear para cada entrada en la BD una réplica para cada idioma. De este modo cuando se crea una entrada en el administrador *Situm* se crea para cada idioma:



+ Opciones		idp	idm	titulo	lugar	descripcion
<input type="checkbox"/>	Editar Copiar Borrar	1	ca	Ampliació wifi	Escola 'el Verbo'	<p>Aquest projecte vol ampliar la xarxa wifi que h...
<input type="checkbox"/>	Editar Copiar Borrar	1	en	Extend a wi-fi network	School 'el Verbo'	<p>This project is focused on the school 'El V...
<input type="checkbox"/>	Editar Copiar Borrar	1	es	Ampliación de wifi	Colegio el Verbo	<p>Proyecto en el que se quiere ampliar la red wif...

Ilustración 3.8 Ejemplo de una tabla de idiomas

Como se trata de la misma entrada se identifican las tres con el mismo `idp` (id de la entrada), pero se diferencian por el `idm` (id del idioma) que identifica el idioma de cada una (`ca`=catalán, `es`=español, `en`=inglés). Por eso, la clave principal de la BD es el conjunto de `idp` e `idm`.

En un principio se diseñaron las BD así para todos los módulos, una tabla para cada una con cada campo para cada idioma, pero esto implicaba un problema en los campos que no tenían ningún idioma, porque podían crearse incompatibilidades (campos con distintos valores para cada idioma), así pues, se diseñaron dos tablas para cada módulo:

- Una incluía todos los valores que no dependen del idioma (tabla `modulo`)
- La otra contenía todos los valores que deben ser traducidos (tabla `modulo_idm`)

Esto es importante ya que para acceder a la BD hay que tener en cuenta que los datos están en dos tablas distintas en función de si son valores de texto o no (traducidos o no).

Para acceder a los datos desde la web, lo que hacemos es un vector de datos que agrupen tanto los de la tabla con texto como la que no. Esto es ídem gracias al comando SQL⁹ (Structured Query Language) `JOIN IN ... ON...`, veamos un ejemplo:

```
$proyecto = Conexion::query('SELECT * FROM proyectos INNER JOIN proyectos_idm  
ON proyectos.id = proyectos_idm.idp WHERE proyectos_idm.idm =  
"'.$_SESSION['idm'].'" AND proyectos.estat = 1');
```

En la anterior consulta lo que estamos haciendo es seleccionar todos los campos de la tabla `proyectos` y de la tabla `proyectos_idm` siempre y cuando el `id` de un proyecto sea el mismo que el `idp` del `proyecto_idm`, otras condiciones que imponemos son que seleccione los campos del idioma activo (`proyectos_idm.idm = "'.$_SESSION['idm'].'"`) y que se haya activado la publicación desde Situm (`proyectos.estat = 1`).

Como hemos visto, la selección de los datos de la BD es muy fácil, para la inserción se complica un poco más ya que hay que tener en cuenta que los datos que no tienen idioma van a una tabla y los que necesitan traducción van en otra tabla de la BD. Así pues, se tuvieron que analizar todos los campos que usaba ya el administrador e irlos clasificando. Esto comportaba realizar llamadas `SQL` por duplicado, por ello, se decidió crear una clase extra (`Conexion.php`) que realizaba todos los comandos `SQL`, de este modo, los accesos a la BD se hacían siempre del mismo modo indicando simplemente la tabla a la que se accede (de idioma o no), los comandos y los valores.

Por otro lado, hemos diseñado las BD para que los campos comunes entre ellas tengan el mismo nombre (`id`, `idp`, `idm`, `id_autor`, fecha de creación, fecha de modificación, `estado`), así podemos generalizar muchos de los campos.

Veamos el **pseudocódigo** de la función que inserta los campos a la BD:

```
//Enviamos la $tabla en que queremos insertar los campos de $arr, en caso de querer  
incluir alguna condición la enviamos a través de $extra (no es obligatorio este campo)  
public static function insertar($tabla, $arr, $extra = ''){  
    self::abrir();  
    $sentencia = "INSERT INTO ".$tabla;  
    //Separamos de la variable $arr los $campos de la tabla y los $valores a insertar  
    $campos = implode(", ", array_keys($arr));  
    $valores = implode("'", "'", $arr);  
    $sentencia .= " (".$campos.") VALUES ('".$valores."') ".$extra;  
    //Devolvemos el id de la conexión, si no ha sido correcta devolvemos 0  
    if (mysqli_query(self::$connexio, $sentencia)){  
        return = mysqli_insert_id(self::$connexio);  
    }  
    else return 0;  
    self::cerrar();  
}
```

⁹ Los comandos SQL son los que nos permiten interactuar con la BD (base de datos)

Como podemos ver, no tiene una gran complejidad de código, pero nos permite adaptarla de manera muy fácil donde anteriormente existía una llamada a la BD, en este caso del tipo `INSERT INTO...`, ya que simplemente hay que separar los campos que se incluyen en la tabla `modulo_idm` y los que van a `modulo` (sin idioma). De este modo podemos tratar los dos tipos de campos paralelamente sin tener que modificar gran parte del código que existía anteriormente.

Nota: hay que tener en cuenta, que cuando se añade un campo de idioma, debe generar tantas filas en la tabla (con el mismo `id`) como idiomas que acepta la web. Por lo que en el código, se tuvo que tener en cuenta esto e ir rellenando los campos según el idioma que se seleccionaba y dejar vacíos los campos que no se introdujesen.

3.4 Nuevas funciones Situm

Entre las modificaciones que hemos hecho en *Situm*, la mayor ha sido la adaptación a multi-idioma, pero también hemos hecho otras modificaciones para mejorar el administrador para poder generalizar la mayoría del contenido de la web y así poder personalizarla desde aquí.

El administrador, en su origen, tenía era otro nombre, a pesar de ello, se ha decidido rebautizarlo ya que hemos generado una nueva versión de este.

Entre alguna de las mejoras que hemos incluido, está la creación de una **galería de imágenes** para las distintas entradas. Para ello lo que hemos hecho es crear una nueva función dentro de las distintas acciones que ya incorporaba el administrador (listar entradas, crear una nueva, editarla y/o eliminarla). Para entender cómo funciona esta función, veamos el **pseudocódigo** de la función que incluye las imágenes (también se ha creado la opción de borrar las imágenes y añadir un texto, pero el código es muy similar, por lo que no lo vamos a comentar aquí; tampoco vamos a comentar el estilo que se ha diseñado para mostrar las imágenes insertadas en *Situm*):

```
private static function core_galeria($id, $modul){
    $carpeta = '../arxiu/galeria/';
    if (isset($_FILES)){
        foreach($_FILES as $key => $value){
            //Compruebo que la imagen es jpg, png, gif
            if ($tipo == '.jpg' || $tipo == '.gif' || $tipo == '.png'){
                new Arxiu('../arxiu/');
                // Si no existen las carpetas se crean
                if(!file_exists('../arxiu')|| !file_exists('../arxiu/galeria')){
                    Arxiu::crear_carpetas();
                }
                //Recojo el nombre del archivo
                $nomfinal = $_FILES[$key]['name'];
                //Recojo los datos de la imagen
                list($ancho, $altura, $tipo, $atr) =
getimagesize($carpeta.$nomfinal);
                //Creo dos tamaños de imagen (medi=mediana, mini=miniatura), estas siempre son
                //del mismo tamaño para no tener problemas de dimensiones cuando las presentemos
                //en la web.
                //Mini 250x160 o al revés (según si son verticales o horizontales)
                //Medi 1250x800 o al revés (según si son verticales o horizontales)
                if($ancho > $altura){
                    Arxiu::thumbnail('galeria/'.$nomfinal,250,'/mini_'.$nomfinal);
                    Arxiu::thumbnail('galeria/'.$nomfinal,1250,'/medi_'.$nomfinal);
                }else{
                    Arxiu::thumbnail('galeria/'.$nomfinal,160,'/mini_'.$nomfinal);
                    Arxiu::thumbnail('galeria/'.$nomfinal,800,'/medi_'.$nomfinal);
                }
            }
        }
    }
}
```



```
//Insertamos en la BD de la galería el id de la galería (a que entrada pertenece), el módulo al que se ha añadido, el nombre de la imagen y la fecha de creación.
    $cams['id_galeria'] = $id;
    $cams['modul'] = $modul;
    $cams['imatge'] = $nomfinal;
    $cams['alta'] = time();
    Conexion::insertar('l2_galeria', $cams);
} else {FX::infopop('Formato incorrecto')}
}
}
return $html;
}
```

Como podemos ver, en esta función se llama a la clase externa Arxiu.php, en la cual se crean las funciones que interactúan directamente con la carpeta y sus archivos. Entre estas funciones está thumbnail(), esta crea la imagen que se ha subido según su formato de origen y la agrega en la carpeta /arxiu/galeria (la carpeta que está en el servidor con todos los archivos que se han subido).

Otra de las modificaciones que se ha incluido es la **gestión de permisos**, este administrador permitía entrar los datos según usuario, pero sin ningún permiso concreto a los usuarios. Para ello se crearon tres tablas nuevas:

- Tabla de permisos: donde se añaden todos los permisos que puede tener un usuario. Cada función de los módulos es un permiso, de este modo, cuando se selecciona un permiso se incluye en esta tabla con el id del grupo que estamos creando o editando.
- Tabla de perfiles: desde el administrador se crean unos perfiles con unos permisos determinados, de este modo, cada usuario se asocia a un perfil (por ejemplo, se puede crear un perfil de 'edición de noticias' para crear usuarios que solo puedan acceder a las distintas funciones del módulo noticias.
- Tabla de usuario-perfil: este relaciona el usuario con el perfil al que se le asocia. Además en esta tabla también se incluye un campo para seleccionar si el perfil es total (puede editar todas las entradas que se han incluido) o individual (sólo puede editar las entradas que ha creado ese usuario).

Para gestionar todo esto, se ha partido de la clase que ya incluía el administrador para crear los distintos usuarios. Para ello lo que se ha hecho es crear un sub-módulo dentro del módulo de usuarios que permita editar los grupos, ha este se le da un nombre y un alcance (individual o total) y posteriormente se seleccionan todos los permisos:

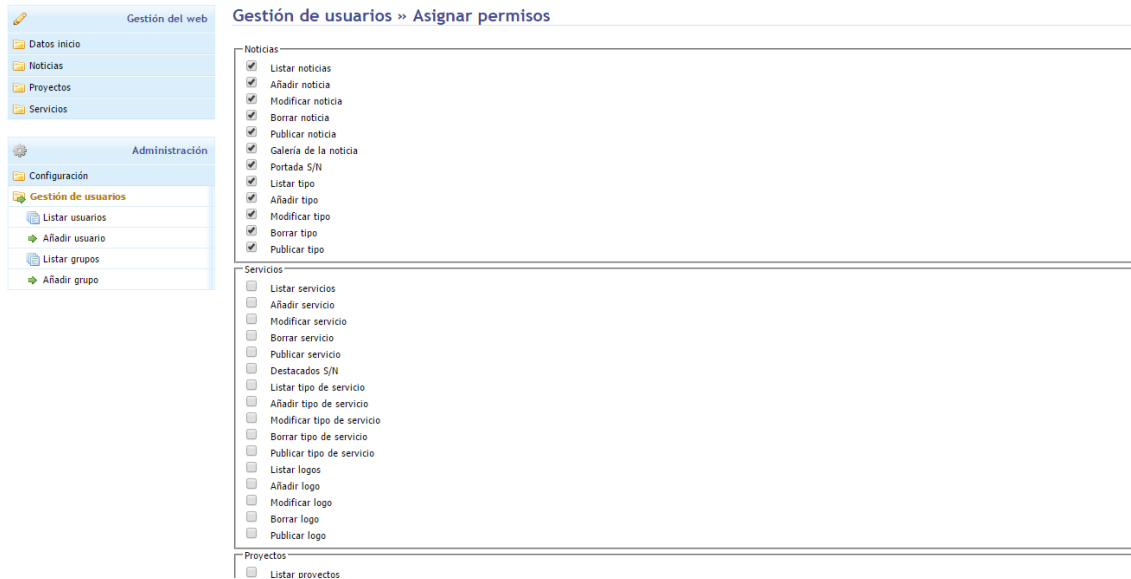


Ilustración 3.9 Selección de permisos en Situm

y estos se añaden en la tabla:

	tipus	idelem	idmodul	1	accio
<input type="checkbox"/>	Editar Copiar Borrar	0	1	3	listar_noticias
<input type="checkbox"/>	Editar Copiar Borrar	0	3	3	modificar_noticias
<input type="checkbox"/>	Editar Copiar Borrar	0	3	3	afegir_noticias
<input type="checkbox"/>	Editar Copiar Borrar	0	3	3	esborrar_noticias_tipo
<input type="checkbox"/>	Editar Copiar Borrar	0	3	3	modificar_noticias_tipo
<input type="checkbox"/>	Editar Copiar Borrar	0	3	3	afegir_noticias_tipo
<input type="checkbox"/>	Editar Copiar Borrar	0	3	3	listar_noticias_tipo
<input type="checkbox"/>	Editar Copiar Borrar	0	3	3	activar_noticias
<input type="checkbox"/>	Editar Copiar Borrar	0	3	3	galeria_noticias
<input type="checkbox"/>	Editar Copiar Borrar	0	3	3	publicar_noticias_tipo
<input type="checkbox"/>	Editar Copiar Borrar	0	3	3	esborrar_noticias
<input type="checkbox"/>	Editar Copiar Borrar	0	3	3	listar_noticias
<input type="checkbox"/>	Editar Copiar Borrar	0	3	3	publicar_noticias
<input type="checkbox"/>	Editar Copiar Borrar	1	2	3	galeria_noticias

Ilustración 3.10 Tabla de permisos de la BD, se han enmarcado las que hemos creado anteriormente (idelem=3)

Por último, otra función que se ha añadido es el poder insertar y seleccionar los **logos** que se quieren poner en la web. Para ello se ha usado de ejemplo el código que ya existía. La idea es crear un sub-módulo dentro del módulo donde se quiere tener la opción de editar los logos, en esta se incluyen imágenes del mismo modo que se hace en la galería de imágenes. Por último, para seleccionar el logo que se desea, se hace desde la entrada, de este modo las entradas de la web son 100% personalizables, no solo del contenido, sino también como se muestran (el diseño se crea por parte del propio usuario del Administrador).

4. Resultados

Empiezo este apartado anunciando que ha sido posible cumplir con los objetivos que nos marcamos al empezar este proyecto, ya que hemos podido cumplir con todas las necesidades de la alcaldía de Bilwi. Por otro lado, han quedado algunos puntos en el aire, en concreto, crear una web 100% personalizable (desde el contenido hasta la selección de módulos a insertar), para poderla llevar a diferentes lugares y adaptarla a sus necesidades sin tener que acceder al código. A pesar de que se todas las mejoras del administrador han sido en función de esto, por tiempo, no ha sido posible acabar de mejorarla del todo, ya que nos hemos centrado en que los módulos de la web cumplieren con esta agilidad (personalizar el contenido desde el administrador) y con las necesidades de la alcaldía de Bilwi (y del resto de instituciones a las que queremos llevar la web). A pesar de ello, se va a seguir trabajando en ello, en primer lugar se creará un archivo de configuración, que pueda ser editado fácilmente desde el Administrador, que incluya todos los elementos que actualmente son fijos en la web (introducidos por código).

Vamos a analizar por cada objetivo cómo se han desarrollado y qué se ha resuelto y qué ha fallado en cada uno:

1. Administrador

El administrador debe ser una herramienta para la alcaldía de Bilwi, y la resta de instituciones que estén interesadas en esta web, muy fácil de usar y que permita editar cómodamente la web sin tener que acceder al código. Se ha conseguido que el administrador sea muy fácil de usar y que permita poder editar ciertos elementos de la web (logos, imágenes...). A pesar de ello, se quiere poder generalizar aún más la web, para que sea el propio cliente el que pueda editar la web a su gusto sin tener que acceder al código base (título, colores...)

2. Web y administrador multi-idioma

Para que la web pueda tener el mayor alcance posible se deseó poder verla en varios idiomas y que puedan acceder a los proyectos desde cualquier rincón del mundo y entenderlos. Esto se ha podido conseguir totalmente, ya que tanto para el administrador y para la web se pueden mostrar los distintos idiomas.

3. Web escalable

A efectos del usuario, que la web sea escalable no tiene visibilidad, pero es importante para poder añadir nuevos módulos fácilmente. En efecto esto se ha hecho así y desde un inicio se han creado todos los módulos paralelamente, de este modo para añadirlo sólo hay que añadirlo en el navegador de la cabecera y crear el acceso a este.

Finalmente, si se quieren añadir en el administrador también es muy fácil: simplemente hay que crear el archivo de configuración (las funciones que se quieren incluir y que se debe solicitar en el formulario) y añadir el módulo en la BD.

De este modo, tenemos una web totalmente escalable para poderla adaptar fácilmente a otras instituciones.

4. Distintos módulos

- Noticias

Se quería un módulo que permitiese listar todas las noticias de la alcaldía y que se pudiesen compartir en las redes sociales. Se ha conseguido hacer así, aunque se desea poder crear un diseño mucho más sencillo y con más opciones, ya que actualmente se muestra un listado para acceder a la noticia que se desea, pero sería interesante que el listado dependa de una selección de temas del usuario.

- Servicios
La idea de este módulo es poder listar e informar a los habitantes de los distintos servicios que ofrece el ayuntamiento. Se ha conseguido cumplir con esto y que el diseño sea fácilmente editable gracias a los logos.
- Proyectos
Se quería un módulo para mostrar la oferta de proyectos que se quieren realizar en Bilwi y así poder recibir ayuda. Este es el módulo más completo, ya que hemos considerado que era el módulo más importante para la alcaldía y el cual debía ser más completo para que empezase a funcionar cuanto antes. Así pues, se ha cumplido con lo que nos marcamos al principio y se ha logrado crear un módulo altamente dinámico, fácil de usar y muy visual.
- Contacto entre escuelas
Este módulo surgió a partir de un proyecto que tienen en Bilwi con la hermandad de Vilafranca del Penedés, el cual consiste en que distintas escuelas de ambos sitios se escriben cartas con las distintas inquietudes de cada uno. La idea era poderlo agilizar por medio de correo. Se ha conseguido poder enviar correos entre distintos correos, des de distintos correos. A pesar de ello, no se ha decidido poner en la web todavía ya que se quiere poder gestionar los datos des del administrador para poder controlar los correos que se envían.

5. Gestión en local

Este era otro punto importante a cumplir, ya que es muy común que el acceso a internet sea débil en estas áreas geográficas y no se pueda acceder durante largos tiempos. Por eso era muy importante que todo lo que se incluyese fuese local, sobretodo, las librerías externas. En concreto, el mapa de los proyectos para poder gestionar los proyectos en cualquier momento. Aunque no se han podido implementar los gráficos del mapa en la web, sí se han podido incluir todas las librerías que gestionan todo el mapa (marcadores, coordenadas...) de este modo se siguen visualizando los marcadores y se pueden introducir datos incluso sin tener acceso a internet.

Estos han sido los resultados que se reflejan en la web de la alcaldía de Bilwi, pero hay que tener en cuenta que éste es un proyecto real, una web que se va a utilizar en la alcaldía de Bilwi y en otras organizaciones con las que tenemos contacto hace tiempo de parte de la asociación de estudiantes de AUCOOP (Asociación de Universitarios para la Cooperación). Por ello se va seguir trabajando en estos puntos que acabamos de comentar y en crear más módulos en función de lo que cada institución necesite y así acabar obteniendo una web totalmente modular, que se pueda diseñar por el mismo cliente.

5. Costes

En este apartado se va a hacer un cálculo de costes en el hipotético caso que se desarrollase en una empresa, donde es necesario tener en cuenta los costes de licencias, maquinaria y las horas de trabajo.

También se hará un estudio del coste que hubiese implicado desarrollar esta misma plataforma web con la ayuda de los Gestores¹⁰ ya existentes en el mercado.

Finalmente, debido a que se trata de un proyecto de cooperación real, se presentará el coste total final que le supondrá al cliente, en este caso, la alcaldía de Bilwi.

5.1 Costes del proyecto

Licencias

Todo el software que se ha usado para programar ha sido de código libre, por lo que no ha sido necesario pagar ninguna licencia. Estos han sido:

- Notepad: software para editar código fácilmente, permite la integración de varios lenguajes, en nuestro caso hemos podido trabajar sin ningún problema con *php*, *html*, *css*, *javascript* y *sql*.
- MySQL: software para la gestión de bases de datos con el que hemos creado todas nuestras tablas de la web y hemos implementado las del administrador.
- Apache: servidor web que nos ha permitido trabajar cómodamente la web para poder manejar los datos de la BD desde la web.

TOTAL: 0€

Maquinaria

Nuestra herramienta de trabajo únicamente ha sido mi ordenador personal. Además, para la implementación de la web, se ha decidido tenerla en un servidor local (en la misma alcaldía), por lo que será necesario el uso de una torre de ordenador¹¹ donde tener todo el contenido que irá al dominio web. Por lo que solo vamos a incluir el precio de este:



Ilustración 5.5.1 Torre de ordenador para ser usada como servidor (almacén de datos)

TOTAL: 499€

¹⁰ Comentadas al detalle en el apartado State-of-art

¹¹ Torre HP IntelCore i5 con 8GB

Dominio

Como se va a usar el mismo servidor de la web como *host*, solo es necesario la contratación de un dominio. Según el dominio que se desee varía el precio:

Nombre de dominio	Precio por 1 año	Renovación de dominio
.es	14,95 €	14,95 €
.fr	20,95 €	20,95 €
.eu	14,95 €	14,95 €
.com	13,95 €	13,95 €
.net	13,95 €	13,95 €
.org	27,95 €	27,95 €

Tabla 5.1 Tabla comparativa de precios de dominio

En este caso, hemos decidido usar el dominio .com en lugar de .ni (Nicaragua) para que la web sea fácilmente accesible en cualquier rincón del mundo.

TOTAL: 13,95€/año

Horas de trabajo

Se ha hecho un cálculo de horas a partir del diagrama de Gantt¹² y se ha generado una tabla¹³ que muestra las horas que se han dedicado por cada tarea y como se han distribuido estas por semana. En definitiva, sale un total de 665 horas empleadas en la realización de este proyecto.

Hay que tener en cuenta que en este proyecto, muchas de las tareas previamente han necesitado una formación intensa, en primer lugar porque no había trabajado nunca en web. En segundo lugar, algunas de las librerías que se han insertado carecían de información, por lo que se ha necesitado toda una documentación para implementarlas correctamente.

En la realidad, cuando se contrata a un ingeniero para que programe una web, se considera que este ya tiene conocimientos en el tema. Por ello, de las horas reales empleadas en el proyecto, se van a restar las horas de formación¹⁴, quedando en 413 horas. El precio que se ha fijado es de 8 €/hora, ya que es el precio medio de un recién graduado o de un estudiante en prácticas. En total:

Horas trabajadas	413 horas
Precio/hora	8,00 €
TOTAL (sin IVA)	3.304,00 €

Tabla 5.2 Tabla de horas trabajadas y coste

¹² Anexo 2. Diagramas, tabla 8.1

¹³ Anexo 2. Diagramas, tablas 8.2 y 8.3

¹⁴ Anexo 2. Diagramas, tablas 8.4 y 8.5

Costes totales

	Precio base	IVA 21%	IRPF 11%	TOTAL
Costes personales	3.304,00 €	-	363,44 €	3.667,44 €
Servidor/host	412,40 €	86,60 €	-	499,00 €
Dominio web (año)	13,95 €	-	-	13,95 €
TOTAL:				4.180,39 €

Tabla 5.3 Tabla de costes totales (con IVA y IRPF)

5.2 Costes con CMS

Para hacer este estudio de costes, vamos a suponer que queremos implementar una web con las mismas necesidades a través de *Wordpress* y los *plugins* que ofrece. Primero, hay que tener en cuenta que esta web se va a instalar en zonas geográficas pobres, donde lo más seguro es que no se tenga conocimientos de programación, por lo que cada extensión que se quiera poner en la web deberá hacerse vía *plugins* de la plataforma. En este caso, hemos escogido *CMS* ya que es la que incluye todo lo que necesitamos.

Por esto mismo, es necesario optar por *Wordpress Premium*, ya que la versión gratis de *Wordpress* solo permite crear entradas del tipo blog y necesitamos una web más moldeable, para adaptar los distintos módulos según lo definido y para que le sea fácil de utilizar al usuario. Por otro lado, se han estudiado los distintos *plugins* que permiten hacer muchas de las tareas que hemos incluido en nuestra web, para ello se han estudiado los gratuitos y los de pago, y se ha escogido el que se adapta a los requisitos iniciales (siempre buscando la versión más económica). Hay que tener en cuenta que con *Wordpress Premium* ya tenemos el dominio web, por lo que no deberemos adquirirlo.

En primer lugar, como ya hemos comentado anteriormente, *Wordpress* no es originalmente multi-idioma, para ello existen dos opciones:

1. *Plugins* gratuitos que traducen el contenido directamente (estos suelen tener ciertas limitaciones como el número de palabras) a través de un traductor *online*.
2. *Wordpress Multilingual Plugin*¹⁵ (WMPL), este es el único *plugin* que ofrece *Wordpress* para poder introducir las entradas de la web en varios idiomas. Este ofrece tres tipos de pagos:
 - 29\$ la licencia y un pago anual de renovación de 15\$.
 - 79\$ la licencia y un pago anual de renovación de 39\$. Este permite extras como traducir automáticamente los widgets y otros elementos fijos de la web y soporte técnico, entre otros.
 - 195\$ de licencia sin renovaciones. La única diferencia al anterior es no tener que renovar.

En cualquier caso, la licencia de este *plugin* sirve para un número ilimitado de webs. Por lo que se optaría por la última opción, ya que se quiere distribuir esta web en distintas instituciones, por lo que saldría a mejor precio (no deberían pagar renovación), por ello, vamos a contar que cada institución solo paga la licencia más económica (sin renovación).

Por otro lado, añadir OSM a Wordpress es posible con un *Plugin* gratuito ya que es una librería de código abierto. Pero, no incluye poder trabajar en modo local, ya que todo el contenido del mapa se encuentra en el servidor de Wordpress, al cual se accede vía Internet.

¹⁵ <https://www.wpml.org>

Finalmente, para acceder a las redes sociales también deberíamos instalar un plugin, para ello hemos buscado el más económico (Ultimate Social Deluxe), ya que las versiones gratis nos limitan demasiado. Esta incluye código *responsive* (se adapta a la pantalla del dispositivo) y admite incluir más de 25 botones.

En resumen, con Wordpress, el coste más económico que hemos encontrado para crear una web parecida a la nuestra es:

	Precio/mes	Precio/año
Wordpress Premium	8,25 €	99,00 €
WMPL		26,53 €
OSM		- €
Redes sociales		13,72 €
TOTAL Wordpress:		139,25 €

Tabla 5.4 Costes por generar una web equiparable a la generada en este proyecto en Wordpress

A esto se debería sumar el coste de los ordenadores con los que se va a trabajar para editar y completar el contenido de la web.

5.3 Costes reales

En la realidad, al ser un proyecto de cooperación, muchos de los costes se han abaratado. En primer lugar, el servidor será administrado por el programa “Reutilitza¹⁶” del CCD y la asociación “Tecnología per a Tothom de la UPC” (TxT). Por otro lado, los proyectos que se hacen por parte de AUCCOOP son sin ningún tipo de ánimo de lucro, es decir, los costes del personal tampoco van a incluirse en el total de costes reales ya que no es la intención de este proyecto.

Así pues, el coste real que tendrá que aportar la alcaldía de Bilwi será el del dominio:

Dominio web (año)	13,95 €
-------------------	---------

Tabla 5.5 Costes reales para la alcaldía de Bilwi

¹⁶ Reutilitza es un programa que organiza TxT para la obtención de ordenadores en desuso para formatearlos y reutilizarlos para proyectos de cooperación (<http://reutilitza.upc.edu>).

6. Conclusiones personales y desarrollo futuro

Desde que conocí a AUCOOP he podido aplicar los conocimientos aprendidos durante la carrera para mejorar las condiciones de distintas zonas del mundo con pocos recursos. Esto no solo me ha ayudado a consolidar algunos de los conceptos estudiados al largo de la carrera y ver su usabilidad, sino que también me ha aportado realidad. Durante mis estudios no paraba de darle vueltas como enfocar mi carrera, me quería especializar en procesamiento de señal (audiovisuales) pero no quería dejar el resto de asignaturas, en especial, la parte más telemática. Entonces descubrí que podía aplicar mis pocos conocimientos en otros lugares donde los necesitaban. Para ellos tecnología no es *High Definition* o *Dolby Surround*, para ellos tecnología es contacto con el exterior, es una fuente de recursos, es poder encontrar trabajo, es aprender, etc.

Este proyecto me ha permitido aportar mi pequeño grano de arena a una ciudad nicaragüense, Bilwi, para que puedan gestionar los datos de la alcaldía cómodamente para que puedan crecer como ciudad y como municipio.

Por otro lado, al ser un proyecto que tendrá un impacto real me ha dado más motivación para hacerlo lo mejor posible para que cumpla con sus necesidades. Por ello, este proyecto no acaba aquí, a partir de este punto se va a seguir trabajando en esta web, para seguir mejorándola para cumplir con todas sus necesidades. Además, este proyecto formará parte del repositorio de AUCOOP para ofrecerlo a otras instituciones con necesidades parecidas o que puedan adaptarse en esta plataforma web, por ello, también se quiere seguir trabajando en esta web para que se puedan recoger las distintas necesidades de estas organizaciones.

En concreto, se pretende llevar esta web a la alcaldía de Bilwi este verano y también se va a adaptar para llevarlo a Camoapa (una ciudad situada a 120km de Managua, la capital de Nicaragua) para la asociación 'Hogar Luceros del Amanecer', con los que también se trabajó des de AUCOOP con un proyecto de limpieza de aguas. Esta asociación se dedica a la educación de los niños con menos recursos de esta zona rural, se aseguran que estos niños tienen los alimentos y el material educativo suficiente. Así pues, si la convocatoria del CCD en la que hemos participado sale exitosa, iremos a estas dos ciudades nicaragüenses para implementar esta web y explicarles bien cómo funciona.

7. Bibliografía

J.L. Ojeda Martín, “Aplicació web per a la gestió d’un ajuntament” thesis, Facultat de Informàtica de Barcelona, Universitat Politècnica de Barcelona, 2009

César Perez. *MySQL para Windows y Linux*, 1ª ed. RA-MA, 2004

7.1 Webgrafía

Php API: php.net

7.1.3 Mapa

GoogleMaps: <https://developers.google.com/maps/web/?hl=es-419>

Mapa OSM (Bilwi): <https://www.openstreetmap.org/#map=13/14.0328/-83.3904>

Librería Open Layers: <http://openlayers.org>

Librería Mapbox: <https://www.mapbox.com>

Vectores del mapa OSM (gráficos): <https://openmaptiles.org>

7.1.2 Librerías JavaScript

Jquery: <https://jquery.com>

Ajax: api.jquery.com/jquery.ajax

Fancybox: fancybox.net

Boxslider: bxslider.co

8. Anexos

8.1 Anexo 1: Manual de usuario

En este Anexo se va a detallar el funcionamiento de la web tanto a nivel de usuario como de cliente. Debido a que la mayor complejidad de uso se encuentra en el Administrador (portal en el que el cliente puede introducir todos los datos de la web), al final de cada apartado se mostrará el resultado que se muestra en la web en función de lo que se ha introducido en el Administrador y se aprovechará para comentar como lo puede consultar el usuario de la web.

Por ello se va a hablar de usuario tanto para los que acceden al Administrador como a la web, aunque cabe aclarar que los usuarios del Administrador son los que gestionan todo el contenido web, mientras que los usuarios de la web son todas aquellas personas que acceden a la web para consultar los datos que contiene la web.

8.1.1 Idioma predeterminado de Situm

Se ha predeterminado como idioma principal el español ya que los principales gestores de este Administrador (la alcaldía de Bilwi) son de habla española.

Cabe indicar que esto solo afecta a la visualización de todos los portales del Administrador, ya que, en cuanto a los datos que se quieren añadir a las distintas entradas de la web, se deberán añadir en los distintos idiomas que permite la página web.

Se puede cambiar el idioma en cualquier momento que el usuario necesite.

8.1.2 Acceso a Situm



Ilustración 8.1Inicio sesión Situm

Este es el portal que aparece cuando accedes al Administrador. Como se puede ver es necesario introducir el “Nombre de usuario” y la “Clave de acceso”, estos deben corresponder a los que se hayan creado previamente, no se puede acceder con un nombre de usuario inexistente.

También es posible seleccionar el idioma en el que se quiere mostrar este portal, recuperar la contraseña en caso de olvidarla y acceder a la web.

8.1.3 Estructura de Situm

Este tutorial vamos a explicarlo con una sesión con el perfil de “administrador”, es decir, con todos los permisos.

Esta es la apariencia que tiene *Situm* tal cual inicias sesión:

1. Gestión del web

- Datos inicio
- Noticias
- Proyectos
- Servicios

2. Administración

- Configuración
- Gestión de usuarios

3. Bienvenid@ de nuevo, Administrador

Fecha	Módulo	Entrada	Autor
03/05/17 16:41	inicio	Monuments	arturito
03/05/17 16:40	inicio	Cultura i clima	arturito
03/05/17 16:40	inicio	Cómo llegar	arturito
03/05/17 16:39	inicio	Ubicació geogràfica	arturito
03/05/17 16:39	inicio	Visió, missió i valors	arturito
03/05/17 16:38	inicio	Documentos de interés	arturito
03/05/17 16:38	inicio	Agermanaments	arturito
03/05/17 16:37	inicio	Direccions administratives	arturito
03/05/17 16:37	inicio	Qui som	arturito
03/05/17 16:36	inicio	Quienes somos	arturito

Estadísticas

- 3 Regs. Noticias
- 4 Regs. Servicios
- 3 Regs. Proyectos
- 9 Regs. Datos inicio

Situm Core (Versió 1.0 FR OhNO!)
A la vera de un charco cacó un zapo

Ir arriba

Ilustración 8.2 Pantalla inicio Situm

En esta se muestra en el lateral izquierdo las distintas acciones que se pueden hacer:

1. Módulos de edición de datos: son los que permiten introducir datos en la base de datos para que se muestren en la web. En este ejemplo se muestran todos los módulos¹⁷ que se incluyen en la web, pero si el gestor no tiene permisos para editar algún módulo, estos no se visualizarán en este sector.
2. Módulos del administrador: este cuadro muestra las acciones que únicamente puede realizar el usuario con rol de “administrador”, si el gestor no tiene estos permisos, este cuadro no se mostrará. Los módulos¹⁸ que se incluyen aquí sirven para la administración del propio *Situm*.

En el centro de la página se muestran:

3. Las estadísticas de las entradas que se han introducido desde *Situm*. A la izquierda se muestran por fecha las entradas que se han realizado (módulo al que pertenecen y nombre de la entrada) y a la derecha los números de entradas realizadas para cada módulo.

¹⁷ Módulos de edición de datos: se explican más detalladamente en los próximos apartados.

¹⁸ Módulos del administrador: se explican más detalladamente en los próximos apartados.

8.1.4 Módulos de edición de datos

Ahora vamos a analizar cada uno de los módulos que puede editar el usuario, como ya hemos comentado anteriormente, estos módulos son los que se muestran en la web que se ha diseñado dentro del marco de requisitos de este proyecto.



Ilustración 8.3 Listado módulos Situm

Se pueden añadir otros módulos que se incluyan en la web creando un vector¹⁹ con todo el contenido que se quiere incluir en este módulo (indicando los campos que se piden en el formulario, como se listan las entradas del módulo y las funciones que se incluyen).

Cuando se accede a uno de los módulos, se muestran todas las entradas que se han incluido en ese módulo. Cada módulo incluye distintas opciones según la funcionalidad de cada uno de los módulos, pero veamos cual es la estructura general de estos listados.



Ilustración 8.4 Opciones Situm

Las entradas se ordenan alfabéticamente, para poder buscar una entrada en concreto se incluyen varias opciones:

1. Seleccionar la cantidad de entradas que se quieren mostrar por páginas
2. Pasar a la página anterior o siguiente para buscar la página con la entrada necesaria
3. Seleccionar la letra por la que empieza el título de la entrada
4. Buscar el nombre de la entrada (o parte del título), pero hay que asegurarse que no se escriba ninguna falta ortográfica ya que el sistema busca la coincidencia

Además, las distintas funciones que se incluyen en cada módulo se muestran de dos maneras:

5. En el mismo listado de la izquierda donde se listan los distintos módulos se abre un desplegable con todos los listados relacionados con ese módulo. En este caso, por ejemplo, no solo se listan los proyectos²⁰, sino también se muestra un listado de la temática que engloba el proyecto o los logos que forman parte del marcador que se mostrará en la web. Además de estos listados, también se pueden crear nuevas entradas para cada uno de ellos.

¹⁹ Este manual está enfocado al usuario tanto del Administrador como de la web, para modificar el código base es necesario entender todo el proyecto que se explica en esta Tesis.

²⁰ Todos estos listados que se muestran en el módulo proyectos se explican con más detalle en el apartado 3.2.XXX, donde se puede ver cuál es el resultado de cada uno en la web.

6. Otras funciones que se pueden hacer en un módulo son las que se muestran en el mismo cuadro donde se listan las entradas. Estas funciones aparecen aquí ya que son propias de cada una de las entradas, pero, para un mismo módulo, todas las entradas tendrán las mismas funciones.

Es decir, Puede ser que algún módulo no permita eliminar las entradas, de ese modo, no se podrá eliminar ninguna entrada de ninguna manera, pero en el caso que el módulo sí que incluya esta función, solo se podrá eliminar una entrada en concreto cuando se seleccione esta acción.

Para entender mejor cómo funcionan estas funciones, veámoslas cada una por separado.

8.1.4.1 Funciones

8.1.4.1.1 Listar las entradas

Como ya hemos visto, cuando se selecciona uno de los módulos, automáticamente se activa esta función para mostrar todas las entradas que hay en ese módulo. Por eso es una acción que incluyen todos los módulos, ya que es necesario para mostrar las entradas que incluye un módulo concreto.

Esta acción forma parte del listado que aparece en el lateral izquierdo ya que es una función genérica del módulo.

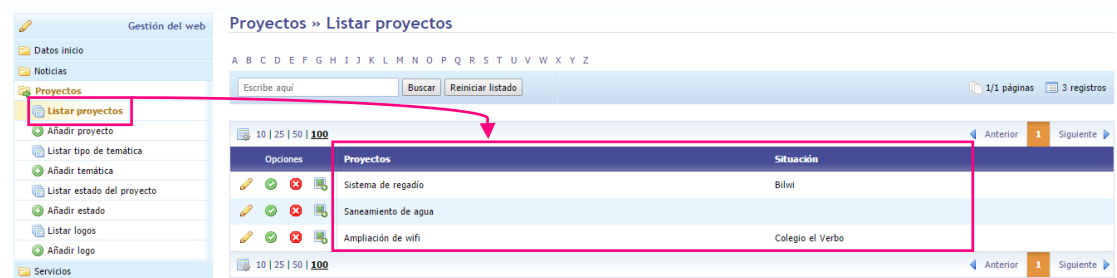


Ilustración 8.5 Lista de entradas de Situm

Además, hay módulos que pueden incluir más listados, generalmente estos listados están relacionados con algún tipo de clasificación de las entradas, por ejemplo, para los proyectos se han de seleccionar la temática a la que pertenece (agricultura...) y el estado en el que está (finalizado, por empezar...).

Como podemos ver en la imagen anterior el esquema del listado es el mismo que el anterior, a pesar de que las acciones que se pueden realizar con estos son diferentes al listado de proyecto, por ello, tiene sus propias funciones.



Ilustración 8.6 Otra lista de entradas de Situm

8.1.4.1.2 Añadir logos de la web

Esta función es muy parecida a la de listar las entradas. Forma parte del listado que aparece en el lateral izquierdo ya que es una función genérica del módulo.

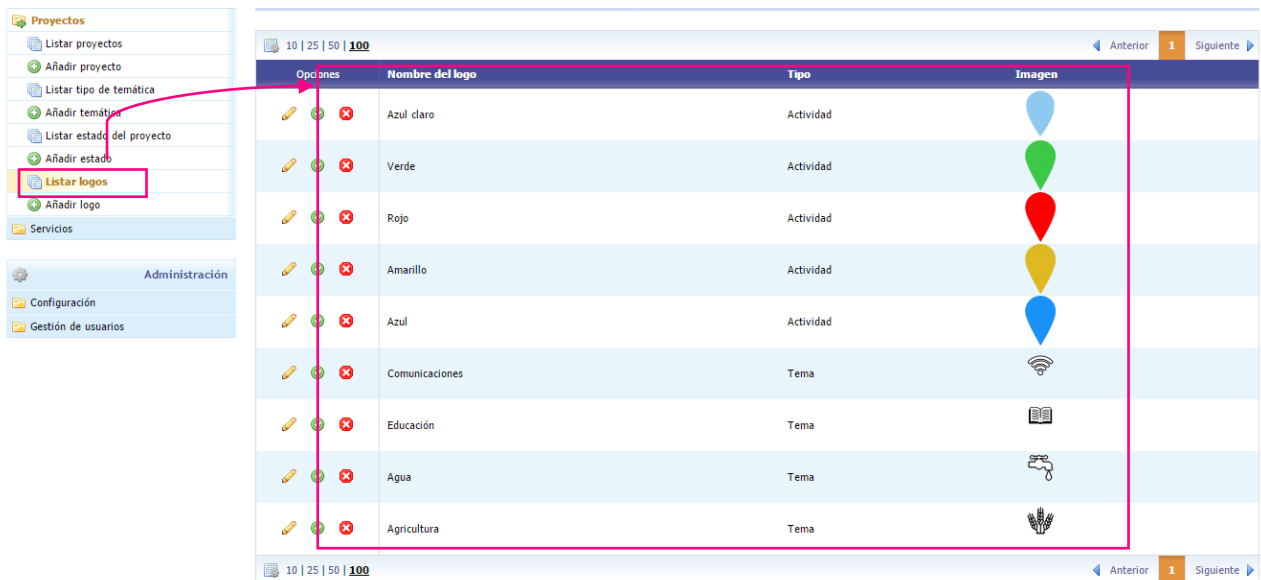


Ilustración 8.7 Lista de logos de Situm

Prácticamente es lo mismo que listar las entradas de un módulo, pero con la diferencia que estos tratan con imágenes. A vista del gestor es lo mismo, pero hay que tener en cuenta que las imágenes se guardan en una carpeta situada en el servidor, por lo que en cuanto a la codificación hay que interactuar con las BD y con la carpeta donde se guardan estos archivos (se guardan en un tamaño predeterminado para que al usarlas en la web tengan el mismo estilo).

Para añadir un logo es muy parecido a como se añade una entrada de un módulo. Lo explicamos en el próximo apartado 'Crear nueva entrada' y también se comentará en los módulos que incluyen esta opción.

8.1.4.1.3 Crear nueva entrada

Como podemos ver, para crear una nueva entrada en cualquier módulo (tanto si es un módulo de los genéricos o uno de los sub-módulos que incluyen estos) hay que hacerlo desde el listado que aparece en el lateral izquierdo, debido a que esta acción es una función genérica del módulo, no afecta a una entrada en concreto. Esta acción también se incluye en todos los módulos, ya que es necesario poder añadir nuevas entradas.



Ilustración 8.8 Formulario nueva entrada de Situm

En esta función nos aparecen nuevos elementos:

1. Se nos abre un formulario en el que rellenar todos los campos que incluye la entrada que queremos añadir, los campos de este formulario dependen del módulo en el que estamos trabajando.
2. Para poder editar los campos de los otros idiomas que incluye la web aparece un listado en la parte superior-derecha para seleccionar el idioma que queremos editar. Es aconsejable rellenar estos campos antes de publicar la noticia, ya que, de lo contrario, pueden aparecer elementos vacíos en la web.

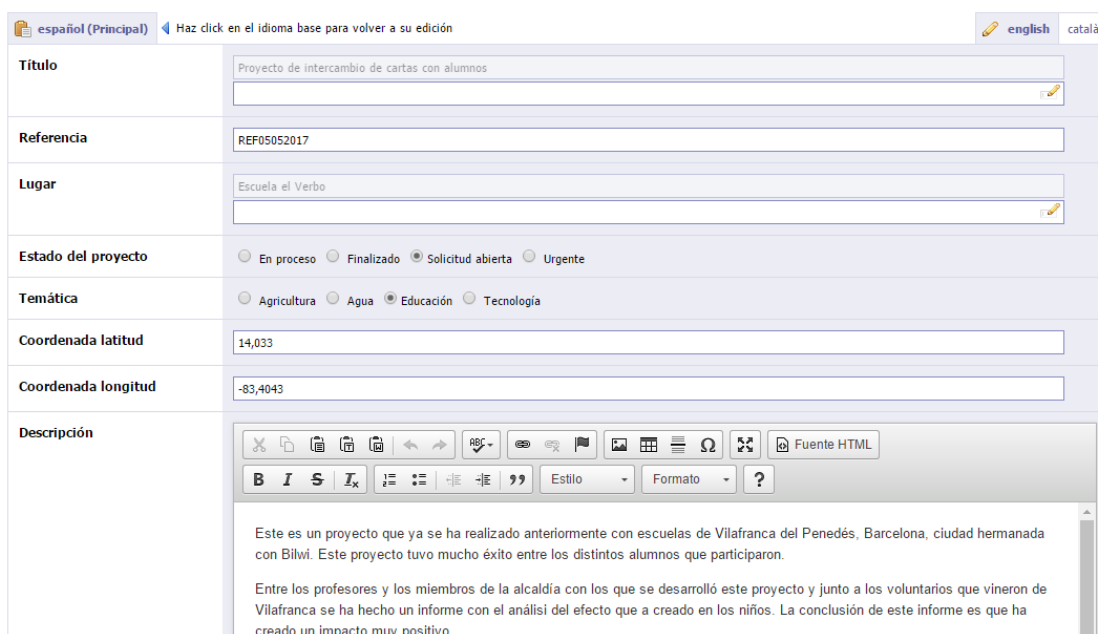


Ilustración 8.9 Formulario multi-idioma de Situm

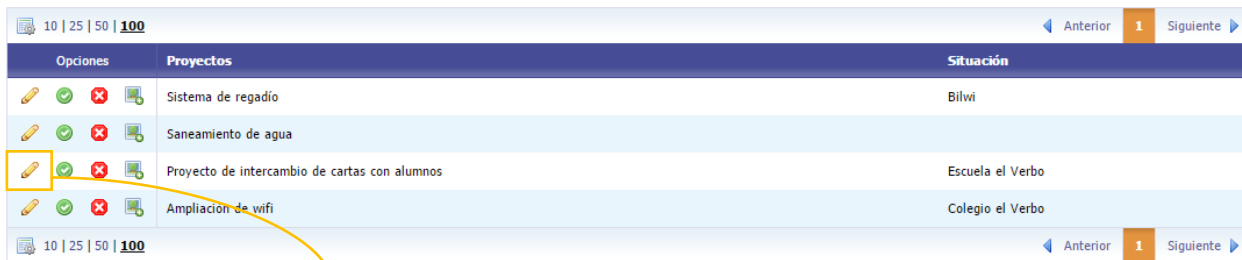
Como podemos ver en la anterior imagen, al seleccionar un nuevo idioma se nos muestra el mismo formulario pero con pequeños cambios:

- Los campos que hay que traducir aparecen con el idioma principal (en este caso tenemos seleccionado como principal el español), este campo no se puede editar, para hacerlo hay que volver al idioma principal, y un segundo campo para rellenar con la traducción de este.
 - Por otro lado, los campos que no implican traducción aparecen con los mismos datos que se han introducido anteriormente, no es necesario volver a rellenarlos.
3. Cuando acabamos de incluir los campos que nos pide la entrada, tenemos tres opciones para guardarlo:
- Enviar y publicar: de este modo se guarda el contenido y se publica automáticamente en la web
 - Enviar y continuar: de este modo podemos guardar el contenido que hemos introducido y seguir trabajando en este, es una manera de ir guardando el contenido para evitar que se nos borre inesperadamente.
 - Enviar: de este modo guardamos todo el contenido pero sin publicarlo en la web, esta opción es buena cuando solo hemos rellenado un idioma y falta hacer alguna modificación previa a la publicación. Se crea una especie de borrador previo.

8.1.4.1.4 Editar una entrada

Esta acción hace referencia a una entrada determinada, por ello, aparece en el propio listado de las entradas, para poder seleccionar cual de todas queremos editar.

Esta función nos permite acceder al formulario de la entrada para modificar el campo o los campos que necesitemos.



















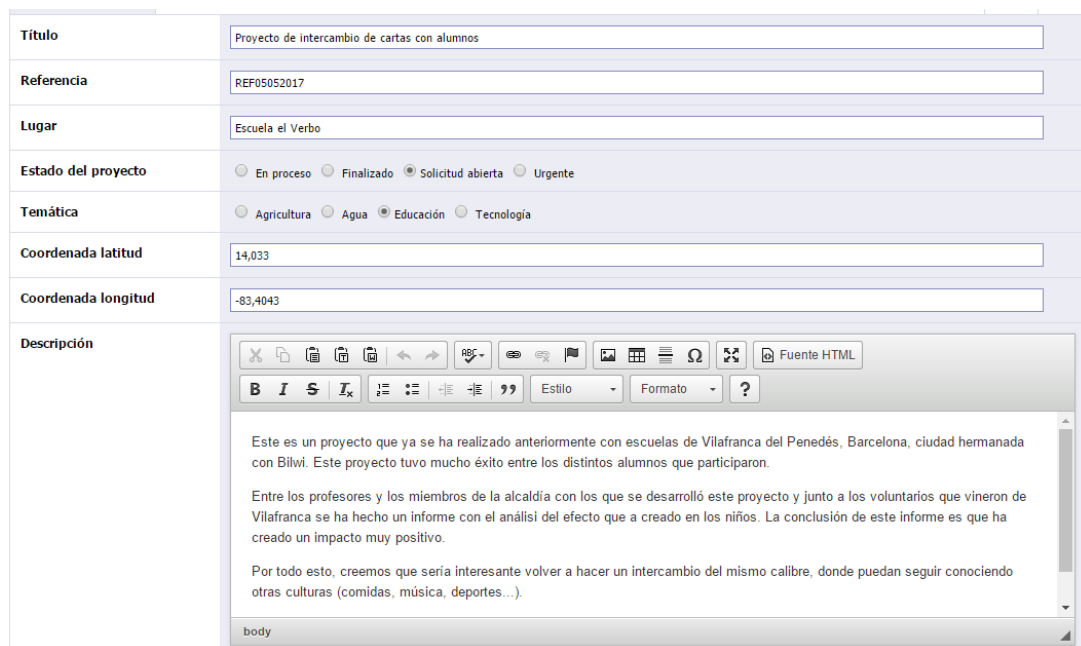
Opciones	Proyectos	Situación
   	Sistema de riego	Bilwi
   	Saneamiento de agua	
   	Proyecto de intercambio de cartas con alumnos	Escuela el Verbo
   	Ampliación de wifi	Colegio el Verbo

Ilustración 8.10 Opción edición de Situm



Título	<input type="text" value="Proyecto de intercambio de cartas con alumnos"/>
Referencia	<input type="text" value="REF05052017"/>
Lugar	<input type="text" value="Escuela el Verbo"/>
Estado del proyecto	<input type="radio"/> En proceso <input type="radio"/> Finalizado <input checked="" type="radio"/> Solicitud abierta <input type="radio"/> Urgente
Temática	<input type="radio"/> Agricultura <input type="radio"/> Agua <input checked="" type="radio"/> Educación <input type="radio"/> Tecnología
Coordenada latitud	<input type="text" value="14,033"/>
Coordenada longitud	<input type="text" value="-83,4043"/>
Descripción	<div><p>Este es un proyecto que ya se ha realizado anteriormente con escuelas de Vilafranca del Penedés, Barcelona, ciudad hermanada con Bilwi. Este proyecto tuvo mucho éxito entre los distintos alumnos que participaron.</p><p>Entre los profesores y los miembros de la alcaldía con los que se desarrolló este proyecto y junto a los voluntarios que vinieron de Vilafranca se ha hecho un informe con el análisis del efecto que a creado en los niños. La conclusión de este informe es que ha creado un impacto muy positivo.</p><p>Por todo esto, creemos que sería interesante volver a hacer un intercambio del mismo calibre, donde puedan seguir conociendo otras culturas (comidas, música, deportes...).</p></div>

Ilustración 8.11 Formulario de edición de Situm

Como podemos ver, no incluye ninguna novedad respecto el formulario que hemos visto en el apartado anterior (crear nueva entrada), la única diferencia es que al crearla los campos aparecen vacíos para rellenarlos y en el modo edición aparecen los campos que hemos añadido y guardado previamente para poderlos modificar.

8.1.4.1.5 Borrar una entrada

Esta acción hace referencia a una entrada determinada, por ello, aparece en el propio listado de las entradas, para poder seleccionar cual de todas queremos eliminar.

Opciones	Proyectos	Situación
	Sistema de riego	Bilwi
	Saneamiento de agua	
	Proyecto de intercambio de cartas con alumnos	Escuela el Verbo
	Ampliación de wifi	Colegio el Verbo

Ilustración 8.12 Opción eliminar de Situm

Al seleccionar la cruz de eliminar borramos todo el contenido de la entrada seleccionada, de este modo no se puede recuperar el contenido de ningún modo, ya que se elimina también de la base de datos. Por eso aconsejo que solo se borre cuando se desee eliminar el contenido de forma permanente, sino se tendrá que volver a introducir de nuevo.

No vamos a mostrar un ejemplo de cómo se elimina una entrada por motivos evidentes.

8.1.4.1.6 Publicar y des-publicar entrada

Para evitar tener que eliminar una entrada que no queremos que se muestre en la web, existe esta función. Esta acción hace referencia a una entrada determinada, por ello, aparece en el propio listado de las entradas, para poder seleccionar cual queremos dejar de publicar o volver a publicar.

Opciones	Proyectos	Situación
	Sistema de riego	Bilwi
	Saneamiento de agua	
	Proyecto de intercambio de cartas con alumnos	Escuela el Verbo
	Ampliación de wifi	Colegio el Verbo

Ilustración 8.13 Opción publicar de Situm

Para ello es tan fácil como apretar el icono del *tick* de verificación de color verde:

- Si está activado y se aprieta el icono, automáticamente se desactiva de la web la entrada seleccionada (aparece el círculo vacío)
- Si está desactivada, entonces la entrada se activa automáticamente en la web (aparece el *tick* verde).

8.1.4.1.7 Publicar en entradas destacadas (portada)

Esta acción hace referencia a una entrada determinada, por ello, aparece en el propio listado de las entradas, para poder seleccionar cual de todas queremos destacar.

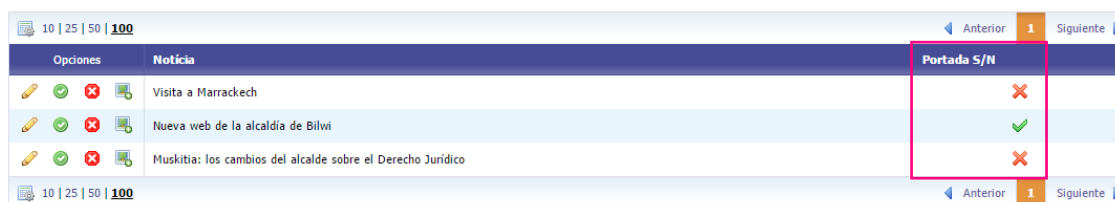


Ilustración 8.14 Opción entradas destacadas de Situm

Funciona de modo parecido a la acción de 'publicar y des-publicar noticia', si tiene una cruz roja es que no está seleccionada para mostrarse como destacada, mientras que si tiene un tick verde es que sí. Para pasar de un estado a otro solo que hay que apretar sobre el icono de la entrada.

8.1.4.1.8 Galería de imágenes

Esta acción hace referencia a una entrada determinada, por ello, aparece en el propio listado de las entradas, para poder incluir las fotografías que nos interesen en una entrada concreta.

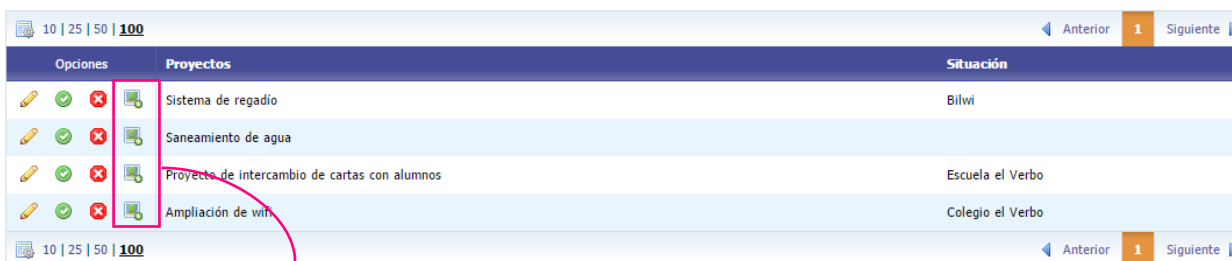


Ilustración 8.15 Opción galería de imágenes de Situm

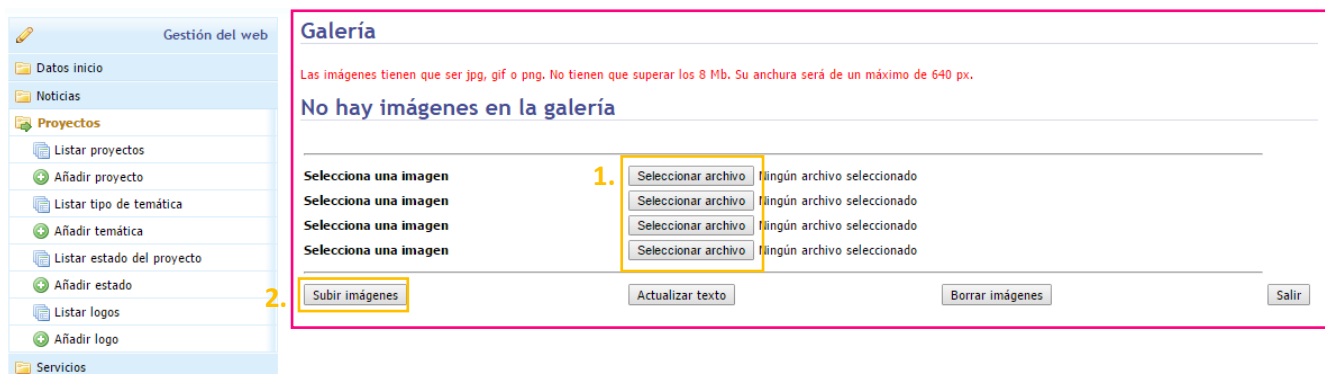


Ilustración 8.16 Opciones de la galería de imágenes de Situm

Al seleccionar la opción de galería de imágenes se nos abre el anterior formulario. Vemos que no se ha introducido ninguna imagen, para subir una fotografía:

1. Seleccionar las imágenes que se desean añadir a partir de los botones 'Seleccionar archivo'.
2. Cuando se han seleccionado las imágenes hay que apretar en el botón 'Subir imágenes' para que se añadan en la BD y las carpetas de almacenamiento interno. Como máximo se pueden subir 4 a la vez (tantas veces que se quiera).

Hay que tener en cuenta que las imágenes solo pueden ser del tipo jpg, gif o png y han de tener un tamaño máximo de 8 Mb. El sistema redimensiona las imágenes a un máximo de 1250px de ancho (si son horizontales) o a 800px de ancho (si son verticales). También se crea una imagen más pequeña, de 250px, para mostrarla en listados u otros lugares predeterminados de. Si no cumplen estas condiciones no se añadirán las imágenes. Esto se hace así para poder manipular las imágenes sin perder calidad.

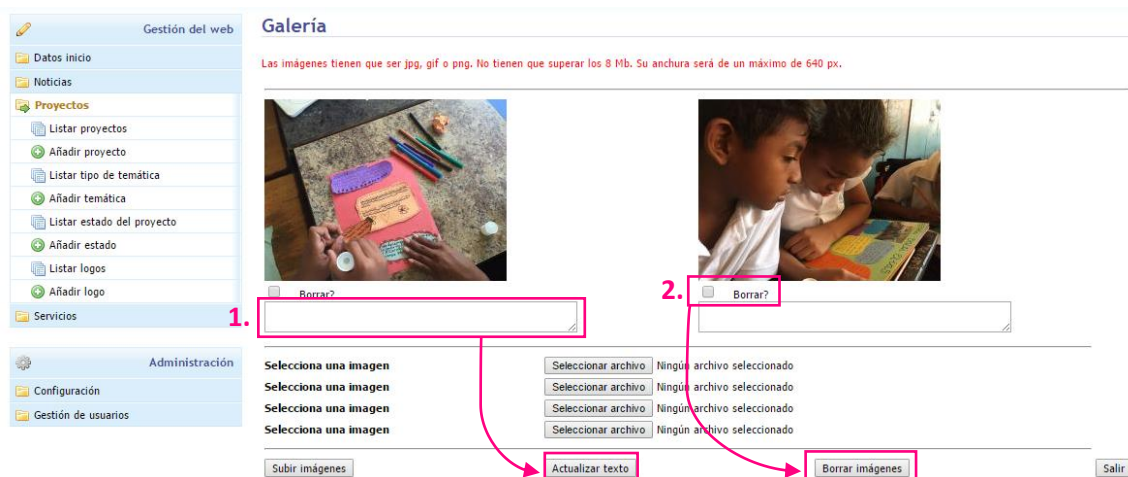


Ilustración 8.17 Más opciones de la galería de imágenes de Situm

Una vez se han subido las imágenes vemos que hay otras opciones que hacer con ellas:

1. Poner un pie de foto añadiendo un texto debajo la imagen que se desee.
2. Borrar las imágenes que ya no se quieren mantener en esta entrada.

8.1.4.2 Todos los módulos

Aquí se muestra que incluye cada uno de los módulos y como los datos que se incluyen se reflejan en la web.

8.1.4.2.1 Datos inicio



Ilustración 8.18 Módulo 'Inicio' de Situm

El módulo de inicio no incluye ningún sub-módulo. Como se puede ver en la imagen anterior, las entradas tienen las siguientes funciones:

- Editar
- Publicar
- Eliminar
- Galería de imágenes

Estas entradas se muestran en la portada de la web:

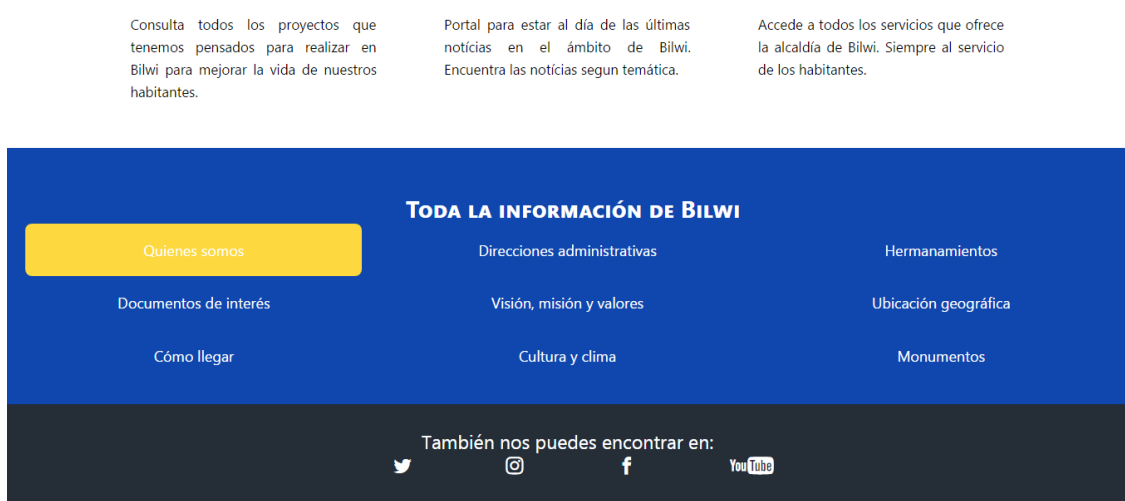


Ilustración 8.19 Entradas de Inicio en la web

Al seleccionar una de estas entradas se muestra el contenido a continuación:

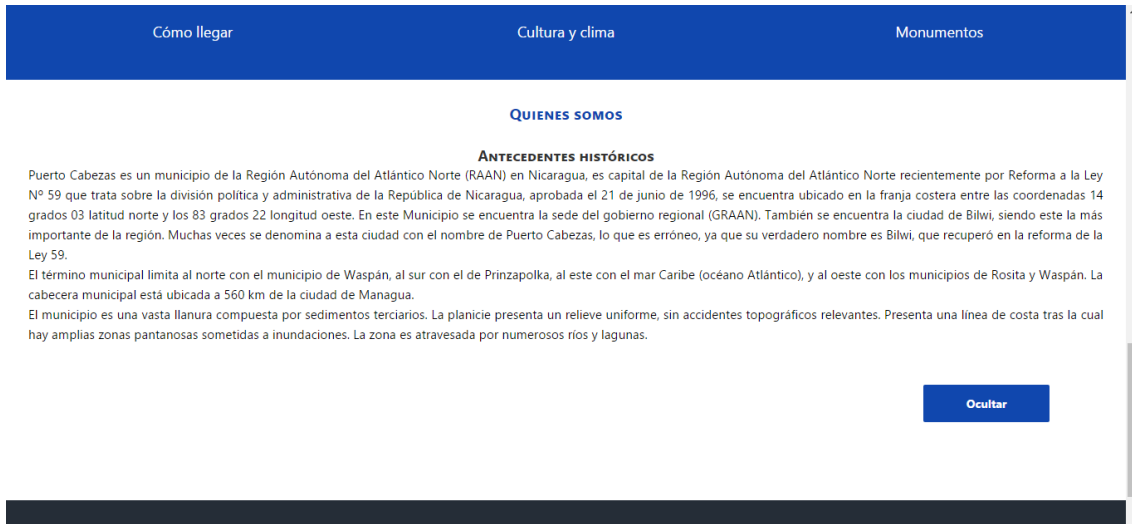


Ilustración 8.20 Contenido de la entrada Inicio en la web

Para mostrar otra entrada solo es necesario apretar en otro de los textos que aparecen en el cuadro azul con 'Toda la información de Bilwi'. En caso de querer ocultar el contenido de la entrada seleccionada, solo es necesario apretar en el botón 'Ocultar'.

8.1.4.2.2 Noticias



Ilustración 8.21 Módulo 'Noticias' de Situm

Este módulo incluye las distintas entradas que se quieren hacer en las noticias de la web. Con estas se pueden hacer las siguientes funciones:

- Editar
- Publicar
- Eliminar
- Galería de imágenes
- Destacar en portada Sí/No

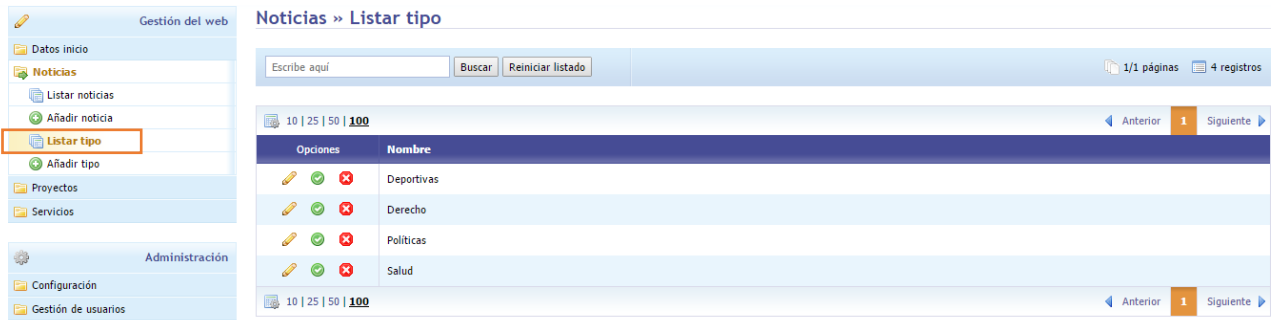


Ilustración 8.22 Sub-módulo 'Tipo de noticia' de Situm

También incluye el listado con el tipo de noticias que se pueden publicar, es decir, como se van a categorizar las noticias que se añadan. Como solo pretende crear un listado de categoría contiene menos funciones que las anteriores, solo incluye:

- Editar
- Publicar
- Eliminar

El resultado en la web es el siguiente:



Ilustración 8.23 Mostrar noticias en la web

1. A la derecha se muestra el listado con todas las noticias según el tipo en el que se han clasificado
2. A la izquierda se muestra un trozo de las noticias seleccionadas como destacadas, en caso que no haya noticias destacadas se muestran las más recientes.

Todo el contenido que incluye una noticia es el siguiente:

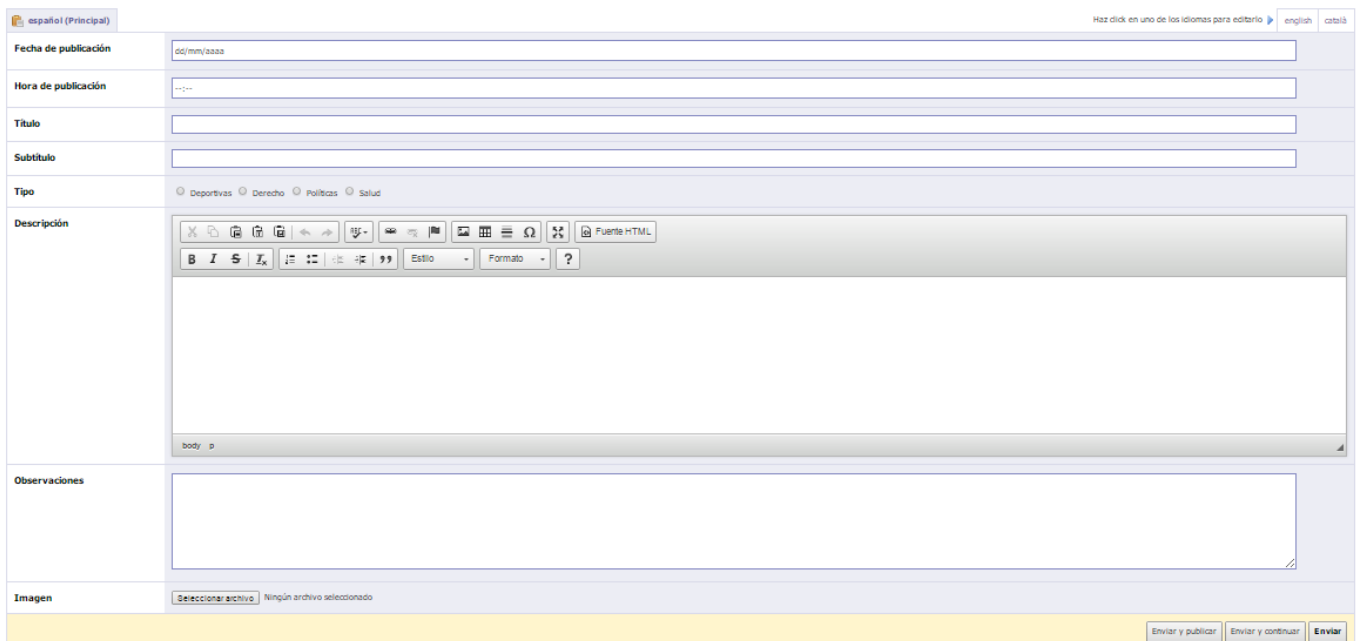


Ilustración 8.24 Formulario 'Noticias' de Situm

Como se puede ver, se debe seleccionar a qué tipo de noticia pertenece (para poderla clasificar en el listado de la web) y también se puede añadir una imagen, esta es la que se usa en el resumen de noticia y también como imagen principal de la noticia completa.

Veamos cómo queda todo este contenido en una noticia completa. Para acceder a esta, se puede hacer desde el listado de la derecha, apretando en el mismo título de la noticia o desde el botón 'Ver la noticia completa', el resultado es el siguiente:



Ilustración 8.25 Contenido completo de una Noticia en la web

Para poder visualizar las imágenes a más tamaño es tan sencillo como seleccionar una y hacer clic, se nos abrirá una ventana emergente como la siguiente:



Ilustración 8.26 Mostrar imagen de la Noticia en la web (con Fancybox)

También está la opción de compartir la noticia en las redes sociales a través de los logos de estas. Al seleccionar una red para compartir la noticia, se abre la red social y te comparte el enlace en tu perfil para que lo puedas compartir con los demás.

Para volver a la página principal de las noticias se puede hacer con el botón de 'Atrás'.

8.1.4.2.3 Servicios

Opciones	Servicio	Destacados S/N
	Servicio 2	
	Permiso de instalación de rótulos	
	Permiso de construcción	
	Constancia de uso de suelo	

Ilustración 8.27 Módulo 'Servicios' de Situm

Este módulo incluye las distintas entradas que se quieren hacer en los servicios de la web. Con estas se pueden hacer las siguientes funciones:

- Editar
- Publicar
- Eliminar
- Destacar en portada Sí/No
- Añadir logos (del tipo de servicio)



Ilustración 8.28 Sub-módulo 'Tipo de servicios' de Situm

También incluye el listado con el tipo de servicios que se pueden publicar, es decir, como se van a categorizar los distintos servicios que ofrece la web. Las funciones de este sub-módulo son las siguientes:

- Editar
- Publicar
- Eliminar

Como ya hemos comentado anteriormente, en algunos módulos se pueden seleccionar los logos que se quieren en la web. En este caso se puede hacer para seleccionar los logos que identifican cada uno de los tipos o categorías de servicios:

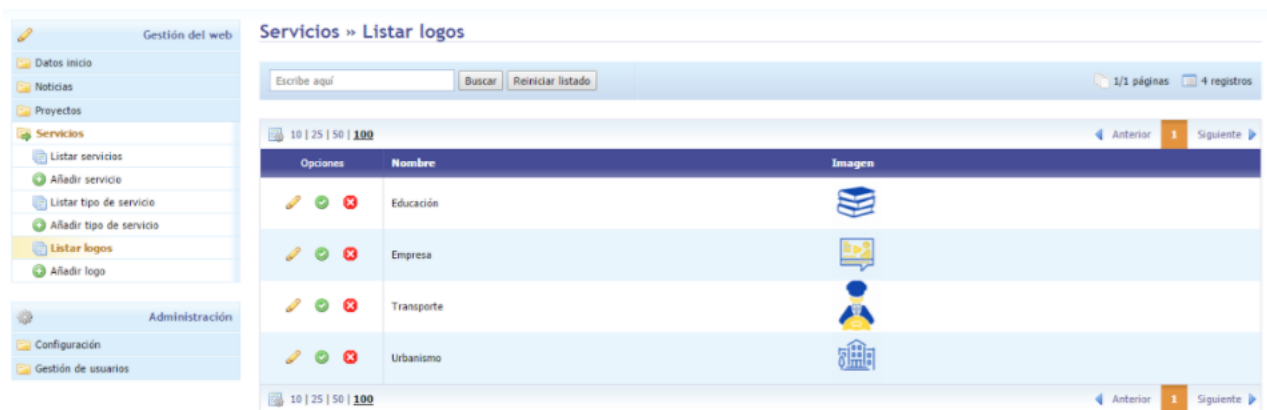


Ilustración 8.29 Sub-módulo 'Logos de tipo de servicio' de Situm

Para crear añadir un nuevo logo hay que acceder a 'añadir logo' y, de todos los logos que se añadido, cuando se crea un tipo de servicio hay que seleccionar el logo que se quiere asociar:

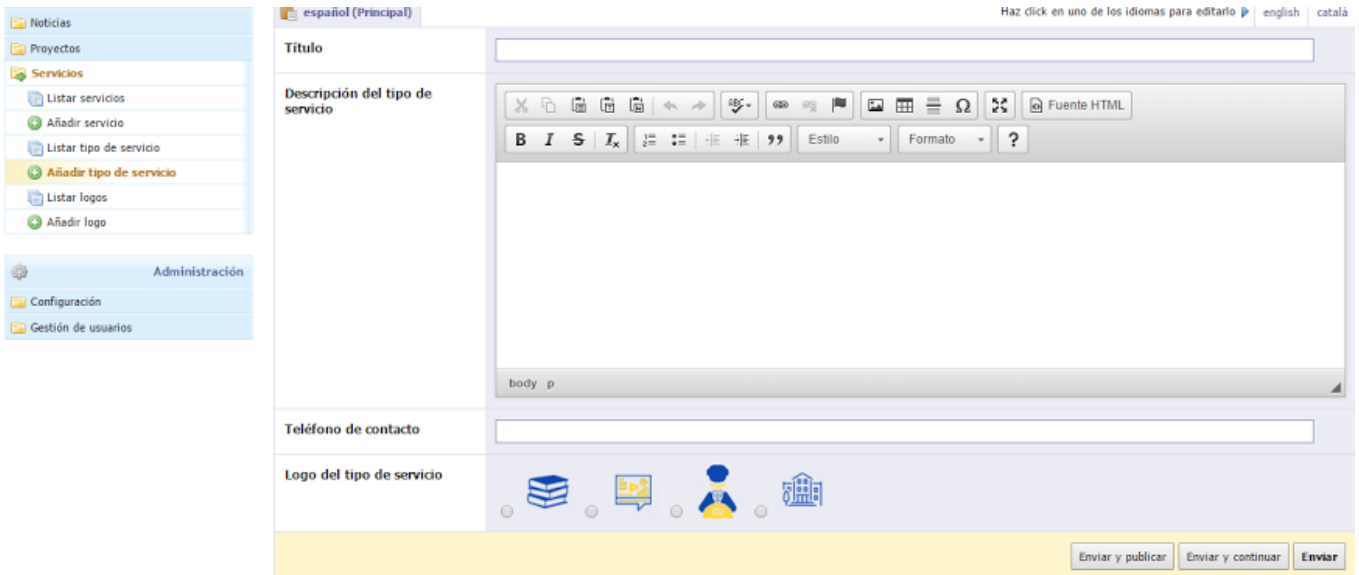


Ilustración 8.30 Formulario de 'Tipo de servicio' en Situm

El resultado de todo esto en la web es el siguiente:



Ilustración 8.31 Página principal de servicios en la web

Como se puede ver, las distintas categorías de los servicios se muestran en la barra superior con los logos que se han seleccionado desde *Situm*, al pasar el ratón por encima se muestran los nombres de cada uno de ellos por si no se sabe con claridad a que hace referencia el logo.

Cuando se selecciona uno de estos se muestra detalladamente la información de la categoría seleccionada y los servicios que incluye este:

SERVICIOS OFRECIDOS

1. **DEPARTAMENTO URBANISMO**

- Permiso de construcción
- Servicio 2
- Constancia de uso de suelo
- Permiso de instalación de rótulos

2. **¿De qué trata: Departamento Urbanismo?**

El **urbanismo** es el conjunto de disciplinas que se encarga del estudio de los asentamientos humanos para su diagnóstico, comprensión e intervención. El urbanismo utiliza a la geografía urbana como herramienta fundamental, e intenta comprender los procesos urbanos a fin de planificar las intervenciones para la cualificación del espacio.

¿Dónde nos puedes encontrar? En la Oficina de Urbanismo, planta baja de la Alcaldía Municipal de Puerto Cabezas.

Para contactar, llame al teléfono 2792-2482.

[← Atrás](#)

Ilustración 8.32 Selección de un tipo de servicio en la web

1. El listado de los servicios que contiene la categoría seleccionada se muestra permanentemente
2. Cuando se selecciona un servicio se muestra la información en el mismo cuadro donde está la descripción de la categoría.

DEPARTAMENTO URBANISMO

- Permiso de construcción
- Servicio 2
- Constancia de uso de suelo
- Permiso de instalación de rótulos

PERMISO DE CONSTRUCCIÓN

1. **¿En qué consiste y cuándo realizarlo?**
Es la autorización que otorga el área de Urbanismo para dar inicio a la realización de las obras de construcción.

2. **¿Dónde se puede realizar este trámite?**
La solicitud de este servicio se puede realizar en la oficina de Urbanismo ubicado en la planta baja de la Alcaldía Municipal de Puerto Cabezas. Los horarios de atención son de Lunes a Viernes de 8:00 am a 5:00 pm.

3. **Requisitos para este trámite**
Copia de Recibo Oficial de caja por pago de supervisión y de impuesto por construcción.
Solvencia Municipal del dueño del proyecto y empresa constructora.
Matrícula y cédula RUC de la empresa constructora.

4. **¿Cuánto hay que pagar?**
Tasa de Supervisión
Urbanizaciones (Pago Mensual) CS0,000
Edificaciones de 1 m² a 100 m² CS1/m²
Edificaciones de 101 m² a 200 m² CS2/m²
Edificaciones de 201 m² a 1,000 m² CS3/m²
Edificaciones de 1,001 m² a más: CS4/m²

5. **Aspectos a tomar en cuenta**
Este trámite dura aproximadamente 3 días.
Supervisión de Proyectos
Falta de pago por 2 meses causa multa del 20%.
Si el área declarada es menor que la existente, la diferencia causa recargo del 50%.
Cuando se suspenden las obras por intervención del área de Urbanismo, no se cobra la tasa de inspección. En caso que se reanuden sin autorización, es sancionada con una multa de 5 veces el valor de la tasa que devengaría desde el reinicio no autorizado de las obras.

6. **¿Dudas?**
Para mayor Información, favor contactarse con la oficina de Urbanismo.
Oficinas:
Oficina de Urbanismo, planta baja de la Alcaldía Municipal de Puerto Cabezas.
Teléfonos:
2792-2482

[← Atrás](#)

Ilustración 8.33 Contenido de un servicio en la web

En cualquier momento se puede acceder a la ventana anterior apretando en el botón de 'Atrás'.

8.1.4.2.4 Proyectos

Proyectos » Listar proyectos

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Escribe aquí

1/1 páginas 4 registros

Opciones	Proyectos	Situación
	Sistema de regadío	Bilwi
	Saneamiento de agua	
	Proyecto de intercambio de cartas con alumnos	Escuela el Verbo
	Ampliación de wifi	Colegio el Verbo

10 | 25 | 50 | 100 Anterior 1 Siguiente

Ilustración 8.34 Módulo de 'Proyectos' de Situm

Este módulo incluye las distintas entradas que se quieren hacer para cada proyecto que se quiere mostrar en la web. Para ello se pueden hacer las siguientes funciones:

- Editar
- Publicar
- Eliminar
- Galería de imágenes

Además, este módulo incluye dos sub-módulos más: temática del proyecto y estado del proyecto.

Proyectos » Listar tipo de temática

Escribe aquí

1/1 páginas 4 registros

Opciones	Temática del proyecto
	Agricultura
	Agua
	Educación
	Tecnología

10 | 25 | 50 | 100 Anterior 1 Siguiente

Ilustración 8.36 Sub-módulo 'Temática del proyecto' de Situm

Proyectos » Listar estado del proyecto

Escribe aquí

1/1 páginas 4 registros

Opciones	Estado del proyecto	Formulario S/N
	En proceso	
	Finalizado	
	Solicitud abierta	
	Urgente	

10 | 25 | 50 | 100 Anterior 1 Siguiente

Ilustración 8.35 Sub-módulo 'Estado del proyecto' de Situm

Ambos listados permiten las funcionalidades básicas:

- Editar
- Publicar
- Eliminar

Pero para el caso del estado del proyecto también se ha adaptado la función de 'Publicar en entradas destacadas' para seleccionar si ese estado permite rellenar el formulario de inscripción o no, esto se hace así para evitar que se rellenen formularios de proyectos que ya están finalizados o de los que ya han recibido ayuda.

Opciones	Nombre del logo	Tipo	Imagen
	Azul claro	Actividad	
	Verde	Actividad	
	Rojo	Actividad	
	Amarillo	Actividad	
	Azul	Actividad	
	Comunicaciones	Tema	
	Educación	Tema	
	Agua	Tema	
	Agricultura	Tema	

Ilustración 8.37 Sub-módulo 'Logos del proyecto' de Situm

Además, para estos, también se pueden seleccionar los logos que se muestran en los marcadores del mapa (para mostrar la situación de los proyectos).

Esto se ha diseñado de la siguiente manera:

- Estado del proyecto: incluye el color del marcador, para poder visualizar según el color de este en qué estado está.
- Temática del proyecto: incluye el logo del proyecto, para poder visualizar según el dibujo del marcador a qué temática pertenece.

De este modo se puede visualizar muy fácilmente que tipo de proyecto hay detrás de un marcador (por ejemplo, en este caso, un marcador rojo con un libro nos indicará que se trata de un proyecto que necesita colaboración urgentemente y está en el ámbito de la educación).

Para ello, cuando se añade un logo hay que seleccionar si pertenece a la temática o al estado:

Ilustración 8.38 Formulario de los logos del proyecto de Situm

Todo esto se visualiza en la web de la siguiente manera:

COOPERA CON BILWI

SELECCIONAR TIPOS DE MARCADORES

Temática:
 Selecciona tema:

Estado:
 Selecciona actividad:

[Ver todos](#)

Referencia	Estado	Temática	Nombre del proyecto	Lugar
REF3	Solicitud abierta	Agricultura	Sistema de riego	Bilwi
REF2	Urgente	Agua	Saneamiento de agua	
REF05052017	Solicitud abierta	Agricultura	Proyecto de intercambio de cartas con alumnos	Escuela el Verbo
REF1	En proceso	Tecnología	Ampliación de wifi	Colegio el Verbo

Ilustración 8.39 Página principal de los proyectos en la web

En el mapa se pueden seleccionar los marcadores que se quieren visualizar, para poder filtrarlos según el interés del usuario. Para poder volver a verlos todos solo hace falta apretar en 'Ver todos'.

Además, se incluye un listado con los proyectos, para poder ver esquematizada la información de estos.

Haciendo clic en el marcador o en la lista, se puede visualizar todo el contenido del proyecto en una ventana emergente para poder volver fácilmente al mapa sin modificar las selecciones.

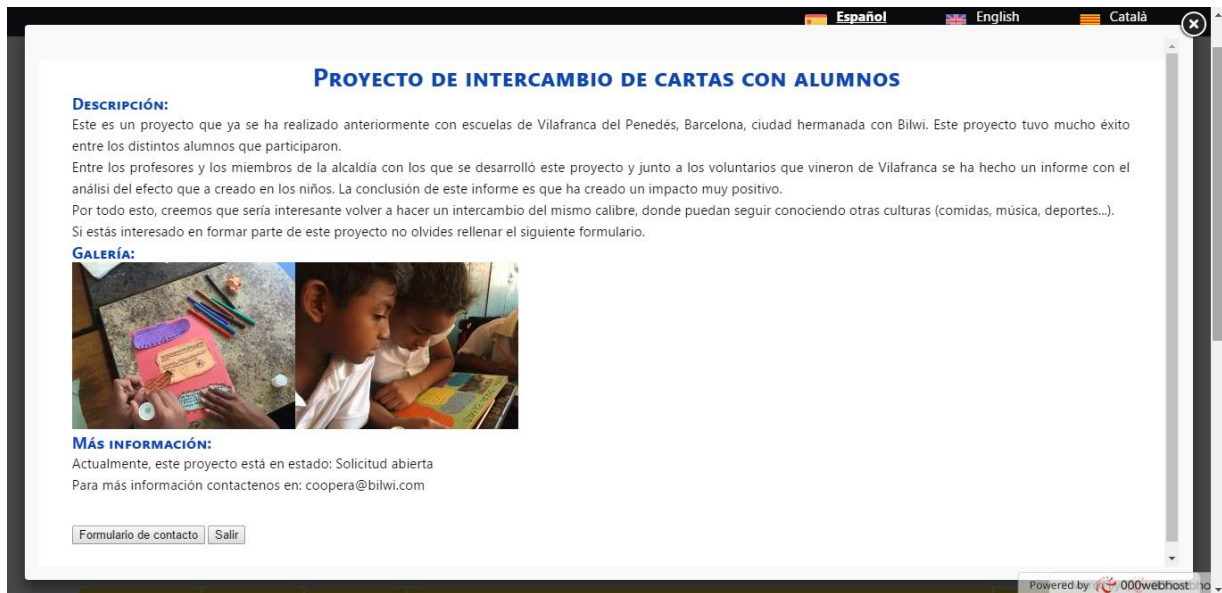


Ilustración 8.40 Contenido de un proyecto en la web (con Fancybox)

Si se quiere enviar el 'Formulario de contacto' se muestra la siguiente página:

The image shows a contact form page with the title 'FORMULARIO DE CONTACTO'. It begins with the question '¿Tienes dudas? ¿Quieres contactar con nosotros? Utiliza nuestro formulario y te contestaremos lo antes posible.' The form contains several input fields: 'Nombre o Razón Social', 'Teléfono', 'E-mail', 'Comentarios', and 'Código de seguridad'. A 'Enviar petición' button is located at the bottom left. A CAPTCHA image with the text 'KVZha' is positioned below the security code field. At the bottom of the page, there is a dark bar with the text 'También nos puedes encontrar en:' followed by icons for Twitter, Instagram, Facebook, and YouTube.

Ilustración 8.41 Formulario de contacto de la web

Cuando el usuario rellena todos los datos necesarios y los envía, esto se recibe en el correo de Bilwi dedicado para gestionar todas las solicitudes. Para facilitar esta tarea, el asunto del correo que se reciba siempre será la referencia del proyecto.

8.1.5 Vocabulario

Alguno de los términos que se han usado en este Manual de Usuario son los siguientes:

Módulo: engloba todo un conjunto de información que, por lo general, se muestra en una nueva pestaña en la web (servicios, noticias...)

Entrada: cada una de las informaciones que se gestionan para un módulo, por ejemplo, en el módulo de noticias, cada una de las noticias que se añaden (editas, borran...) son una entrada.

Gestor: toda aquella persona que puede acceder al administrador para gestionar la información

Usuario: toda aquella persona que accede a la web

8.2 Anexo 2: Diagramas

	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13	W14	W15	W16	W17	W18	W19	W20	W21	W22	W23	W24	W25	W26	W27	W28	W29	W30	W31	W32	W33	W34	W35	W36										
Documentación	█																																											
Estructurar la web		█																																										
Administrador																																												
Gestión de usuarios y permisos																																												
Crear multi-idioma																																												
Crear galería para los módulos																																												
Módulo noticias																																												
Acciones																																												
Formulario a rellenar (conexión con BD)																																												
Módulo proyectos																																												
Acciones																																												
Formulario a rellenar (conexión con BD)																																												
Módulo servicios																																												
Acciones																																												
Formulario a rellenar (conexión con BD)																																												
Módulo inicio																																												
Acciones																																												
Formulario a rellenar (conexión con BD)																																												
Acción: incluir logo																																												
Asociar logos a servicios (tipo)																																												
Asociar logos a proyectos (tema y actividad)																																												

	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13	W14	W15	W16	W17	W18	W19	W20	W21	W22	W23	W24	W25	W26	W27	W28	W29	W30	W31	W32	W33	W34	W35	W36						
Diseño base de datos																																								
Diseñar tablas																																								
BD contenido general																																								
BD contenido según idioma																																								
Crear las tablas de cada módulo																																								
Portal "Proyectos de cooperación"																																								
Insertar mapa OSM																																								
Descargar librerías off-line																																								
Crear marcadores por proyecto																																								
Poder seleccionar marcadores (según tema y/o actividad)																																								
Asociar logo marcador según características del proyecto																																								
Tabla con el listado de los proyectos																																								
Crear fancybox para mostrar proyecto seleccionado																																								
Asociar con el formulario los datos de la ONG interesada																																								
Portal "Servicios"																																								
Agrupar los servicios en bloques (tipo)																																								
Crear listado de servicios																																								
Contenido de los servicios dinámico, con Ajax																																								
Portal "Noticias"																																								
Mostrar las noticias																																								
Listar noticias en temas y fechas																																								
Crear paginador para mostrar noticias																																								
Crear accesos directos con Facebook y Twitter																																								
Función para mostrar noticia completa																																								

	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13	W14	W15	W16	W17	W18	W19	W20	W21	W22	W23	W24	W25	W26	W27	W28	W29	W30	W31	W32	W33	W34	W35	W36					
Portal "Inicio"																																							
Resumen del contenido de la web																																							
Incluir información de la alcaldía																																							
Mostrar contenido de la información, con Ajax																																							
Diseño gráfico																																							
Diseño genérico																																							
Diseño proyectos																																							
Diseño servicios																																							
Diseño noticias																																							
Diseño inicio																																							
Adaptar el diseño en responsive																																							
Traducir																																							
Traducir web																																							
Traducir Administrador																																							
Diseño de los logos de la web																																							
Diseños básicos y sencillos																																							
Diseños a corde la web																																							
Redacción de la memoria																																							
Revisión de errores del programa																																							
Revisión de la memoria																																							
Entrega de la memoria																																							

Tabla 8.1 Diagrama de Gantt real. Como se han distribuido las tareas desde setiembre hasta mayo

	Tarea 1	Horas	Tarea 2	Horas	Tarea 3	Horas	Tarea 4	Horas	Tarea 5	Horas	TOTAL HORAS
Wk1	Reunión y definición del proyecto	6									6
Wk2											0
Wk3	Documentación	8									8
Wk4	Documentación	6	Estructura web	8							14
Wk5	Documentación	6	Estructura web	8							14
Wk6	Documentación	6	Estructura web	8							14
Wk7	Documentación	6	Estructura web	8							14
Wk8	Documentación	6	Estructura web	8							14
Wk9	Documentación	6	Estructura web	8							14
Wk10	Documentación	6	Estructura web	6	Insertar mapa	4					16
Wk11	Documentación	6	Estructura web	6	Insertar mapa	6					18
Wk12	Documentación	5	Estructura web	6	Insertar mapa	6					17
Wk13	Documentación	3	Marcadores del proyecto	15							18
Wk14	Documentación	3	Marcadores del proyecto	15							18
Wk15	Documentación	2	Marcadores del proyecto	15							17
Wk16	Librerías off-line (mapbox)	12	Mostrar noticias	6							18
Wk17	Librerías off-line (mapbox)	12	Listar noticias	6							18
Wk18	PREPARACIÓN EXAMENES FINALES										0
Wk19											0
Wk20											0

Wk21	Librerías off-line (layers)	20								20	
Wk22	Librerías off-line (layers)	20								20	
Wk23	Librerías off-line (layers)	20								20	
Wk24	Librerías off-line (layers)	15	Listar servicios	10						25	
Wk25	Librerías off-line (layers)	15	Listar servicios	10						25	
Wk26	Programar módulos del administrador	15	Diseño gráfico servicios	5	Diseño gráfico proyectos	5				25	
Wk27	Programar módulos del administrador	25								25	
Wk28	Programar módulos del administrador	25								25	
Wk29	Programar módulos del administrador	25								25	
Wk30	Programar módulos del administrador	8	Añadir logos des de Administrador	12						20	
Wk31	Programar Ajax (servicios, inicio)	6	Crear formulario proyectos	6	Conectar logos BD a la web	8			Diseño gráfico de la web	14	34
Wk32	Programar Ajax (servicios, inicio)	12	Programar Fancybox (proyectos)	12	Conectar logos BD a la web	12			Diseño gráfico de la web (genérico y noticias)	14	50
Wk33	Organizar información inicio	8	Crear inicio de la web	24					Diseño gráfico de la web	14	46
Wk34	Organizar información inicio	8	Diseño gráfico inicio	14	Traducir	6	Revisión de errores	10			38
Wk35	Diseño responsive	8	Diseño gráfico inicio	7	Traducir	6	Revisión de errores	10			31
Wk36	Revisión de la memoria	16					Revisión de errores	10			26

Tabla 8.2 Repartición de las horas de trabajo

Horas por semana	
Wk1	6 horas
Wk2	0 horas
Wk3	8 horas
Wk4	14 horas
Wk5	14 horas
Wk6	14 horas
Wk7	14 horas
Wk8	14 horas
Wk9	14 horas
Wk10	16 horas
Wk11	18 horas
Wk12	17 horas
Wk13	18 horas
Wk14	18 horas
Wk15	17 horas
Wk16	18 horas
Wk17	18 horas
Wk18	0 horas

Horas por semana	
Wk19	0 horas
Wk20	0 horas
Wk21	20 horas
Wk22	20 horas
Wk23	20 horas
Wk24	25 horas
Wk25	25 horas
Wk26	25 horas
Wk27	25 horas
Wk28	25 horas
Wk29	25 horas
Wk30	20 horas
Wk31	34 horas
Wk32	50 horas
Wk33	46 horas
Wk34	38 horas
Wk35	31 horas
Wk36	26 horas

TOTAL HORAS:	693 horas
---------------------	------------------

Tabla 8.3 Suma de las horas por semana, en función de la tabla 8.2

Tarea 1	Horas	Tarea 2	Horas	Tarea 3	Horas	Tarea 4	Horas	Tarea 5	Horas	TOTAL HORAS
Estructura web	6									6
Estructura web	6	Insertar mapa	6							12
		Insertar mapa	6							6
Marcadores del proyecto	15									15
Marcadores del proyecto	15									15
Mostrar noticia	6									6
Listar noticias	6									6
PREPARACIÓN EXAMENES FINALES										0
PREPARACIÓN EXAMENES FINALES										0
PREPARACIÓN EXAMENES FINALES										0
PREPARACIÓN EXAMENES FINALES										0
PREPARACIÓN EXAMENES FINALES										0
Librerías off-line (layers)	10									10
Librerías off-line (layers)	10									10
Librerías off-line (layers)	10									10
Librerías off-line (layers)	10									10
Librerías off-line (layers)	10									10
Diseño gráfico servicios	5	Listar servicios	10							15
Diseño gráfico proyectos	5	Listar servicios	10							15
Programar módulos del administrador	25									25
Programar módulos del administrador	25									25
Programar módulos del administrador	8	Añadir logos des de Administrador	12	Conectar logos BD a la web	8			Diseño gráfico de la web	7	35
Programar Ajax (servicios, inicio)	6	Crear formulario proyectos	6	Conectar logos BD a la web	12			Diseño gráfico de la web	7	31
Programar Ajax (servicios, inicio)	12					Crear inicio de la web	20	Diseño gráfico de la web	4	36
Organizar información inicio	8	Programar Fancybox (proyectos)	12							20
Organizar información inicio	8	Diseño gráfico inicio	14	Traducir	6	Revisión de errores	10			38
Diseño responsive	8	Diseño gráfico inicio	7	Traducir	6	Revisión de errores	10			31
Revisión de la memoria	16					Revisión de errores	10			26

Tabla 8.4 Repartición de las horas de trabajo, sin tener en cuenta las horas de documentación y formación

Horas dedicadas cada semana	
Wk11	12 horas
Wk12	12 horas
Wk13	0 horas
Wk14	15 horas
Wk15	15 horas
Wk16	6 horas
Wk17	6 horas
Wk18	0 horas
Wk19	0 horas
Wk20	0 horas
Wk21	10 horas
Wk22	10 horas
Wk23	10 horas
Wk24	10 horas
Wk25	10 horas
Wk26	15 horas
Wk27	15 horas
Wk28	25 horas
Wk29	25 horas
Wk30	35 horas
Wk31	31 horas
Wk32	36 horas
Wk33	20 horas
Wk34	38 horas
Wk35	31 horas
Wk36	26 horas
TOTAL:	413 horas

Tabla 8.5 Suma de las horas por semana, en función de la tabla 8.4

9. Glosario

9.1 Conceptos usados durante la tesis

Cliente: los interesados en usar esta web, instituciones del tercer mundo con pocos recursos.

Módulos: los distintos portales que se pueden visitar en la web en cualquier momento. Entre ellos no hay ninguna relación, por ello, el contenido se carga de nuevo cada vez que se accede a uno de ellos borrando lo que había anteriormente.

Usuario: la persona que accede a la web para buscar información

Situm: es el nombre con el que se ha bautizado al administrador

9.2 Lista de acrónimos

Acrónimo	Significado
AUCCOOP	Associació d'Universitaris per a la Cooperació de la UPC
AJAX	Asynchronous JavaScript And XML
BD	Bases de Datos
CCD	Centre de Cooperació per al Desenvolupament de la UPC
CMS	Content Management System
CSS	Cascading Style Sheets
ETSETB	Escola Tècnica Superior d'Enginyeria en Telecomunicacions de Barcelona
GB	GigaByte
HTLM	HyperText Markup Language
IP	Internet Protocols
IRPF	Impuesto sobre la Renta de las Personas Físicas
IVA	Impuesto al Valor Agregado
JS	JavaScript
OSM	Open Street Maps
PHP	Hypertext Preprocessor
SEO	Search Engine Optimization
SQL	Structured Query Language (lenguaje de consulta estructurada)
TxT	Tecnologia per Tothom
UPC	Universitat Politècnica de Catalunya

Tabla 9.1 A la izquierda los acrónimos mencionados en esta tesis y a la derecha los significados