



Proyecto Final DAW

José Manuel Pacheco Castro

2014



Proyecto Final DAW

Contenido

1	Descripción del Proyecto	3
2	Desarrollo Web Entorno Servidor	5
3	Desarrollo Web Entorno Cliente	15
4	Despliegue de Aplicaciones Web	20
5	Diseño de Interfaces Web	24
6	Empresa e iniciativa emprendedora	27
7	Inglés	32

Descripción del Proyecto

El Proyecto integrado final del curso DAW 2012 / 2014 consiste en el desarrollo de un sitio web para la gestión y mantenimiento de una sociedad protectora de animales.

Cualquier asociación puede hacer uso del sitio web para publicar novedades, dar a conocer a sus animales, contactar con nuevos socios para la asociación y usuarios que estén interesados en adoptar algún animal de la asociación.

Cualquier usuario puede visitar el sitio web, conocer las novedades de la asociación, registrarse en el portal, contactar con la organización, adoptar algún animal de la asociación y/o participar activamente en la asociación.

También existe una sección para la publicación de animales desaparecidos donde cualquier usuario registrado puede dar una descripción acerca del animal perdido.

La aplicación carece de un panel de administración como tal. Al administrador del sitio web se le facilita una cuenta administrativa que le permite acceder a determinadas secciones del sitio web, tales como crear / modificar entradas de noticias y fichas de nuevos animales de la asociación. A esta cuenta va asociada una dirección de email de administración donde recibe las notificaciones de contacto.

Al ser una aplicación desarrollada con el framework Symfony2 en una futura ampliación del proyecto se podría hacer uso de algún bundle de administración para facilitar la gestión del sitio web.

Para la realización de este Proyecto se han utilizado las siguientes herramientas de desarrollo:

1. La aplicación se ha desarrollado utilizando el framework PHP denominado Symfony2 (www.symfony.com), version 2.4.

Se han utilizado diversos bundles de terceros:


- FOSUserBundle, para registro y perfiles de usuario del sitio web.
- KNP Paginator Bundle para la paginación de las entradas del sitio web.
- SwiftmailerBundle para el envío de emails de confirmación / notificación.
- TwigBundle para hacer uso de las plantillas Twig, y la herencia de plantillas.

También se ha hecho uso de un pequeño script PHP para redimensionar imágenes subidas al servidor, denominado SimpleImage.php.

El editor para el desarrollo de la aplicación ha sido NetBeans 7.4, bajo servidor local con Xampp en sistema operativo Windows 8.

Para las capturas de código se ha preferido usar el editor SublimeText2 para mejor visualización de estructura de código y colores.

2. La base de datos del Proyecto se ha gestionado con Mysql, y su aplicación Phpmyadmin.
Se ha hecho uso de la capa de abstracción de BBDD Doctrine2 ORM, versión 2.2.
3. Para el diseño de la base de datos se ha utilizado la herramienta CASE MySQL Workbench,
versión 6.0 CE.

- 
4. Para la interfaz Front-End se ha hecho uso de hojas de estilo CSS3, código Javascript con el framework JQuery (JQuery UI, JQuery Validate, JQuery Minicolors), y diversos plugins para la barra de navegación y el carrusel de imágenes. Se ha hecho uso de herramientas de retoque fotográfico (Photoshop) para el diseño del logo y algunas secciones del sitio web.
El sitio web también contiene una pequeña animación al pie de página para incluir la etiqueta HTML5 Canvas.
 5. Para la documentación de las clases y metodos PHP se ha utilizado PhpDocumentor (<http://www.phpdoc.org/>), y comentarios detallados.
 6. Para la memoria del Proyecto se ha utilizado Word 2013 y una plantilla personalizada.

Desarrollo Web Entorno Servidor

1. Requisitos

- PHP Orientado a Objetos
- MySQL
- En caso de utilizarse un CMS, el alumno debe realizar un módulo independiente de suficiente entidad y dificultad desarrollado en PHP con acceso a una base de datos, que puede ser la propia del CMS o una base de datos independiente.
- De manera opcional se puede crear la aplicación con Symfony. Se valorará muy positivamente esta elección.

2. Symfony2 Framework PHP

Definición: « *Symfony es un framework PHP de tipo full-stack construido con varios componentes independientes creados por el proyecto Symfony* »

Principales características:

- Su código, y el de todos los componentes y librerías que incluye, se publican bajo la licencia MIT de software libre.
- La documentación del proyecto también es libre e incluye varios libros y decenas de tutoriales específicos.
- Aprender a programar con Symfony te permite acceder a una gran variedad de proyectos: el framework Symfony2 para crear aplicaciones complejas, el micro framework Silex para sitios web sencillos y los componentes Symfony para otras aplicaciones PHP.
- Según GitHub, Symfony es el proyecto PHP más activo, lo que garantiza que nunca te quedarás atrapado en un proyecto sin actividad. Además, el líder del proyecto, Fabien Potencier, es la segunda persona más activa del mundo en GitHub.
- Aunque en su desarrollo participan cientos de programadores de todo el mundo, las decisiones técnicas importantes siempre las toma Fabien Potencier, líder del proyecto. Esto evita el peligro de que surjan forks absurdos y la comunidad se fragmente.
- Los componentes de Symfony son tan útiles y están tan probados, que proyectos tan gigantescos como Drupal 8 están contruidos con ellos.
- En todo el mundo se celebran varias conferencias dedicadas exclusivamente a Symfony. Para que te hagas una idea del tamaño de la comunidad, la conferencia Symfony española (llamada deSymfony) es el evento PHP más grande del país.
- Sitio web del proyecto : <http://www.symfony.com>

3. Doctrine2 ORM

Doctrine 2 es un ORM (Object-Relational Mapper) para PHP. Un ORM provee un sistema de persistencia para objetos PHP.

Doctrine 2 se sitúa encima de una capa de abstracción de bases de datos (DBAL). La principal tarea del mapeador es traducir de forma transparente los objetos PHP en filas de una base de datos relacional y viceversa.

Una de las características claves es escribir queries en un dialecto propio similar al SQL, llamado Doctrine Query Language (DQL), inspirado por HQL de Hibernate.

Doctrine 2 es recomendable para aplicaciones que utilicen el paradigma de la programación orientada a objetos. El único requerimiento obligatorio es la versión 5.3.0 de PHP y opcional aunque recomendable la extensión APC.

Doctrine 2 está dividido en tres principales paquetes:

- Common
- DBAL (incluye Common)
- ORM (incluye Common+DBAL)

El paquete Common contiene componentes altamente reutilizables. El namespace raíz es Doctrine\Common.

El paquete DBAL contiene una capa de abstracción de acceso a base de datos situada encima de PDO cuyo objetivo es proporcionar una API que haga de puente entre los diferentes sistemas gestores de bases de datos. El namespace raíz es Doctrine\DBAL.

El paquete ORM contiene el mapeador objeto-relacional que provee la persistencia de objetos PHP a una base de datos. El namespace raíz es Doctrine\ORM.

Podemos descargar Doctrine2 desde: <http://www.doctrine-project.org/projects/orm/download>

La versión actual y estable del proyecto es la 2.4.

4. FOSUserBundle

El FOSUserBundle añade soporte para un sistema de usuario respaldado por la base de datos en Symfony2. Proporciona un marco flexible para la administración de usuarios que tiene como objetivo realizar tareas comunes, tales como el registro de usuarios y la recuperación de la contraseña.

Las características incluyen:

- Los usuarios pueden ser almacenados a través de ORM Doctrine, MongoDB / CouchDB ODM o Propel.
- Soporte de registro, con una confirmación por correo electrónico opcional.
- Apoyo de restablecimiento de contraseña.

5. KNP Paginator Bundle

Paginador de Symfony2 para automatizar la paginación y simplificar la clasificación de elementos del sitio web, además de otras características.

- Se pueden personalizar tanto la vista de paginación como los suscriptores de eventos.
- Posibilidad de añadir filtrado personalizado, clasificación funciona dependiendo de parámetros de la petición.
- Vista de paginación con extensiones basadas en eventos.
- Soporta múltiples paginaciones durante una misma solicitud.

En el archivo `composer.json` vemos todos los bundles que componen el proyecto de terceros que forman parte del proyecto.



```

7      "psr-0": { "": "src/" },
8      "files": ["src/SimpleImage.php"]
9  },
10     "require": {
11         "php": ">=5.3.3",
12         "symfony/symfony": "~2.4",
13         "doctrine/orm": "~2.2,>=2.2.3",
14         "doctrine/doctrine-bundle": "~1.2",
15         "twig/extensions": "~1.0",
16         "symfony/assetic-bundle": "~2.3",
17         "symfony/swiftmailer-bundle": "~2.3",
18         "symfony/monolog-bundle": "~2.4",
19         "sensio/distribution-bundle": "~2.3",
20         "sensio/framework-extra-bundle": "~3.0",
21         "sensio/generator-bundle": "~2.3",
22         "incteev/composer-parameter-handler": "~2.0",
23         "friendsofsymfony/user-bundle": "~2.0@dev",
24         "knplabs/knp-paginator-bundle": "~2.4"
25     },
26     "scripts": {
27         "post-install-cmd": [
28             "Incteev\\ParameterHandler\\ScriptHandler::buildParameters",

```

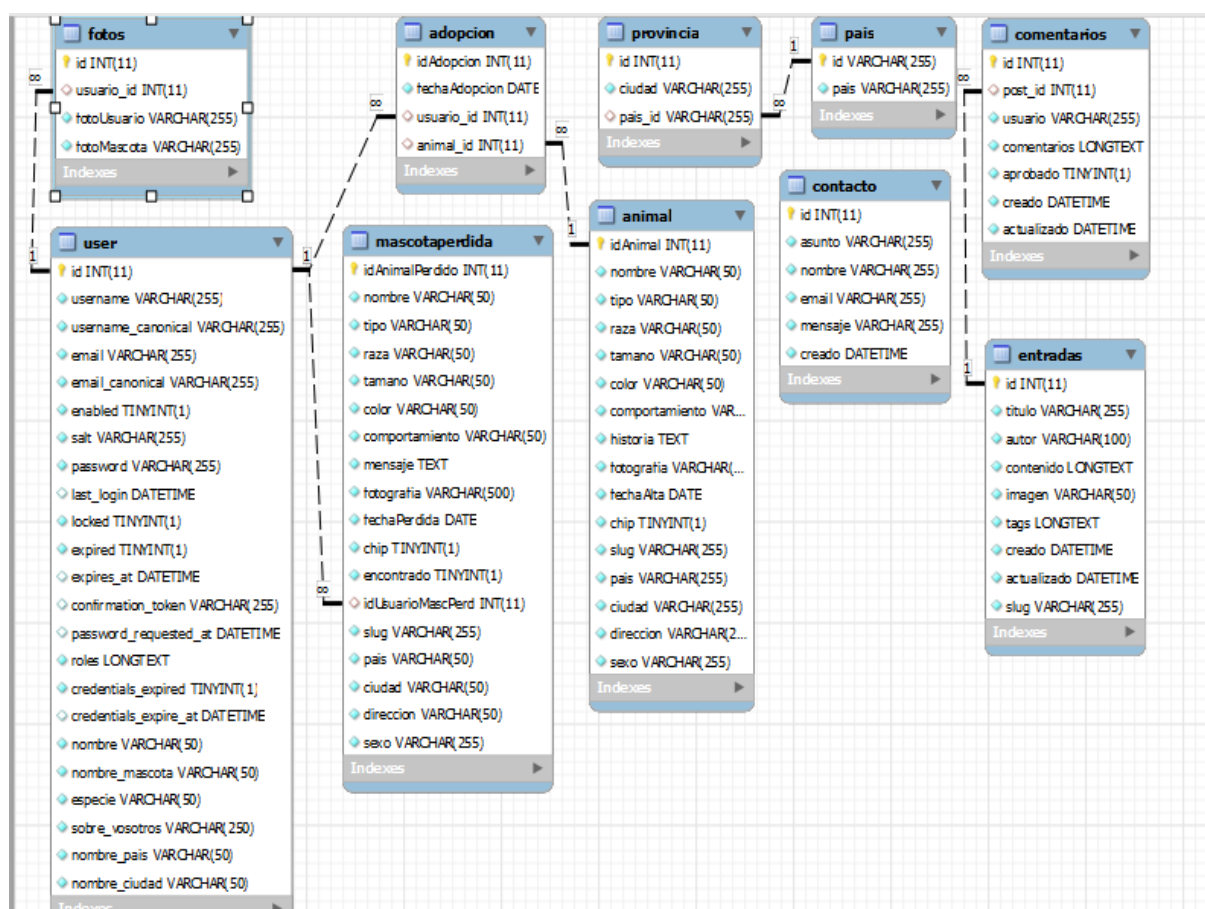
6. PetFriendsProtectorsBundle

Bajo el nombre de este bundle se ha desarrollado toda la estructura Back-End del sitio web. Se compone de un total de 7 controladores (Controller), 10 entidades (Entity), 7 formularios (Form), 29 rutas (routing.yml) y 44 vistas (views).

Las entidades utilizadas en este proyecto son:

- **Post:** Guarda entradas de noticias del sitio web creadas por el administrador.
- **Comments:** Guarda los comentarios asociados a cada noticia, escritos por los usuarios registrados del sitio web. Relacion 1 – N con Post.
- **Animal:** Contiene todas las fichas de los animales que están en la asociación. Relacion 1 – 1 con Adoption.
- **Adoption:** Esta clase entidad almacena el id del animal adoptado junto al id del usuario que lo adopta, además de la fecha.
- **User:** Clase Entidad extendida del FOSUserBundle para almacenar los usuarios registrados. Se han añadido campos nuevos como el nombre, nombre de la mascota, país y ciudad, además de la relación 1 – N con Adoption, relación 1 – N con AnimalLost y 1 – N también con UserFiles, que almacena datos fotográficos del perfil del usuario
- **AnimalLost:** Aquí se almacena las solicitudes de búsqueda de algún animal extraviado que son creadas por cualquier usuario registrado.
- **Contact:** Si algún usuario escribe al administrador a través del formulario de contacto, el contenido se almacena en esta tabla.
- **Country:** Provee un directorio de países necesario para el funcionamiento del registro de usuario del sitio web. Relacion 1 – N con Province.
- **Province:** Aquí se guardan todas las ciudades asociadas a cada país para filtrar cuando es requerido.
- **UserFiles:** Almacena las fotos del perfil y de la mascota, si tuviese, del usuario registrado.

La estructura del modelo de datos queda de la siguiente forma:



* Los nombres de las tablas se han dejado en español a propósito, salvo la de user que es extendida de FOSUserBundle.

Las entidades creadas cuentan con un sistema de validación en el servidor que proporciona Symfony2. Contiene reglas básicas como NoBlank (campos en blanco) y más potentes como patrones de tipo email o IBAN. Para ello tan solo hay que añadir las reglas en forma de anotaciones sobre cada campo que se quiera validar en el servidor.

```

Post.php
28  */
29  protected $id;
30
31  /**
32   * @ORM\Column(type="string")
33   * @Assert\NotNull(message="*Debe escribir un titulo")
34   * @Assert\Length(
35   *     min = "8",
36   *     max = "15",
37   *     minMessage = "*El titulo debe tener {{ limit }} caracteres minimo",
38   *     maxMessage = "*El titulo no debe tener mas de {{ limit }} caracteres")
39   */
40  protected $titulo;
41
42  /**
43   * @ORM\Column(type="string", length=100)
44   */
45  protected $autor;
46
  
```

El controlador principal se denomina PageController, y se encarga de crear la estructura index del sitio web, y de proporcionar el CRUD (listado, creación, edición y borrado) de la sección La Protectora. También renderiza los comentarios asociados a cada entrada y la zona lateral e inferior del sitio web, y las paginas Sobre Nosotros y Contacto.

Se compone de 12 funciones (Action) y la principal es indexAction(), que renderiza las entradas de noticias y proporciona funcionalidad extra AJAX, para paginar las entradas de forma dinámica.

```
/**
 * Listado completo de la Entidad Post
 */
public function indexAction()
{
    $request = $this->getRequest();

    $em = $this->getDoctrine()
        ->getManager();
    $entradas = $em->getRepository('JMPCPetFriendsProtectorsBundle:Post');

    $consulta = $em->createQuery("SELECT p, COUNT(c.id) as cuenta FROM JMPC\PetFriends\ProtectorsBundle\Entity\
GROUP BY p.id ORDER BY p.creado DESC");
    $entradas = $consulta->getResult();

    //Paginacion con KNP Paginator Bundle
    $paginator = $this->get('knp_paginator');
    $pagination = $paginator->paginate(
        $entradas,
        $this->get('request')->query->get('page',1),1
    );

    // Respuesta AJAX para que la paginacion no recargue la pagina
    if ($request->isXmlHttpRequest()) {
        return $this->render('JMPCPetFriendsProtectorsBundle:Page:indexAjax.html.twig', array(
            'posts' => $pagination
        ));
    }

    return $this->render('JMPCPetFriendsProtectorsBundle:Page:index.html.twig', array(
        'posts' => $pagination
    ));
}
```

Otro controlador importante del bundle es el denominado AnimalsController.

Su funcionalidad es similar al PageController, pero aquí se controla el CRUD de la Entidad Animal, que se encarga de guardar los datos de los animales de la asociación.

Contiene una función especial para listar en otra sección del sitio web a los animales que han sido adoptados por algún usuario.

```
/**
 * Listado de la Entidad Adoption
 * Adoption contiene los animales que estan asociados al usuario
 * que lo adopta
 */
public function adoptedAction()
{
    $request = $this->getRequest();

    $em = $this->getDoctrine()
        ->getManager();
    $adoptados = $em->getRepository('JMPCPetFriendsProtectorsBundle:Adoption');

    $consulta = $em->createQuery("SELECT a FROM JMPC\PetFriends\ProtectorsBundle\Entity\Adoption a");

    $adoptados = $consulta->getResult();

    //Paginacion con KNP Paginator Bundle
    $paginator = $this->get('knp_paginator');
    $pagination = $paginator->paginate(
        $adoptados,
        $this->get('request')->query->get('page',1),1
    );

    // Respuesta AJAX para que la paginacion no recargue la pagina
    if ($request->isXmlHttpRequest()) {
        return $this->render('JMPCPetFriendsProtectorsBundle:Adopted:indexAjax.html.twig', array(
            'adopteds' => $pagination
        ));
    }

    return $this->render('JMPCPetFriendsProtectorsBundle:Adopted:index.html.twig', array(
        'adopteds' => $pagination
    ));
}
```

El siguiente controlador que se ha desarrollado es el LostController, para controlar la Entidad AnimalLost.

El controlador UserController se encarga de redireccionar correctamente al usuario a la pagina principal al hacer login, y/o enviar los errores de acceso si los hubiese. Tambien contiene la funcionalidad para actualizar las fotos del perfil, generadas de forma automática al hacer el registro de usuario.

Los controladores ProfileController y RegistrationController son extendidos de FOSUserBundle, para poder agregar funcionalidades extra en el registro y en el perfil. Una funcionalidad extra que se añade al RegistrationController es la de generar un nuevo objeto de la Entidad Fotos para el perfil de usuario cuando se efectúa un nuevo registro.

Cada controlador hace uso de una clase Form, para generar los distintos formularios del sitio web. Se denominan PostType, LostType, ContactType, CommentsType, AnimalType, ProfileFormType y RegistrationFormType. Estos dos últimos son extendidos de FOSUserBundle para añadir campos extra como nombre completo, ciudad o país, necesarios para la funcionalidad del sitio web.

En cuanto a las rutas, se han utilizado un total de 29 rutas. La mayoría son usadas para renderizar las diferentes zonas del sitio web, pero también existen rutas para incluir la funcionalidad AJAX. Estas devuelven un objeto `Response()`, que puede contener HTML, JSON o Texto plano y permite actualizar zonas del sitio web sin recargar la pagina completa. Un ejemplo de este tipo de ruta seria la siguiente:

```

routing.yml
124 jmpc_pet_friends_validate:
125     pattern: /validate
126     defaults: { _controller: JMPCPetFriendsProtectorsBundle:Page:validate }
127
128 jmpc_pet_friends_validate_email:
129     pattern: /validate_email
130     defaults: { _controller: JMPCPetFriendsProtectorsBundle:Page:validateEmail }
131
132 jmpc_pet_friends_update_files:

```

Esta ruta se asocia al Action `validateAction()`, del `PageController`. Es capaz de determinar que los datos que se le envían desde la vista provienen de una función AJAX y tras procesar los datos, devolver una respuesta.

La forma de validar si es AJAX es con el método `isXmlHttpRequest()`. Después se rescata el campo que queremos procesar mediante `request->get(etiqueta name del formulario)[data]`, y devolvemos un nuevo objeto `Response()` con el resultado obtenido.

En este ejemplo, recibimos el nombre de usuario que esta escribiendo un visitante en el formulario de registro. Si el nombre de usuario esta en la BBDD ya registrado por otro usuario, se le envía un 1, y no envia el formulario. En caso contrario devuelve 0.

```

PageController.php
70 /**
71  * Funcion de respuesta de llamada AJAX para validar el nombre de usuario
72  * en el registro de usuarios
73  * Devuelve 0 o 1
74  */
75 public function validateAction()
76 {
77
78     $request = $this->getRequest();
79     $em = $this->getDoctrine()->getManager();
80
81     if ($request->isXmlHttpRequest()) {
82         $username = $request->request->get('fos_user_registration_form')['username'];
83         if ($username != null) {
84             $consulta = $em->createQuery("SELECT u FROM JMPC\PetFriends\ProtectorsBundle\Entity\User u WHERE u.username = :username");
85             $resultado = $consulta->getResult();
86             $response = new Response();
87
88             if ($resultado != null) {
89                 $response->setContent(1);
90                 $response->headers->set('Content-Type', 'text/plain');
91             }
92             else {
93                 $response->setContent(0);
94                 $response->headers->set('Content-Type', 'text/plain');
95             }
96             return $response;
97         }
98     }
99 }

```


El proyecto contiene un total de 44 vistas (views). La vista principal se encuentra en `app/Resources/views/base.html.twig`, que contiene la estructura básica de bloques.

```
base.html.twig
1 |[{# app/Resources/FOSUserBundle/views/base.html.twig #}]
2 |[{# Plantilla principal del sitio web PetsFriends #}]
3
4 <!DOCTYPE html>
5 <html lang="en">
6   <head>
7     <title>PetsFriends {%- block title %}{%- endblock %}</title>
8     {%- block meta %}{%- endblock %}
9     {%- block stylesheets %}{%- endblock %}
10
11    {%- block javascript %}{%- endblock %}
12  </head>
13  <body class="page1">
14    <div class="wrapper">
15      <!-- ===== header & gallery ===== -->
16
17      <header>
18        <div class="container_12">
19          <div class="grid_12 prefix_1 suffix_1">
20            {%- block login %}{%- endblock %}
21            <h1>
22              <div class="box_logo">
23                <a href="{{ path('jmpc_pet_friends_protectors_homepage') }}">
24                  
25                </a>
26              </div>
27            </h1>
28            {%- block menu_block %}{%- endblock %}
29            <div class="clear"></div>
30          </div><!-- End grid_12 prefix_1 suffix_1 -->
31        </div><!-- End container_12 -->
32      </header>
33      {%- for type, messages in app.session.flashbag.all() %}
34        {%- for message in messages %}
35          <div class="flash_notice">
```

Las demás plantillas del sitio web heredan la estructura de `base.html.twig`, añadiendo las distintas secciones. Las plantillas están divididas en bloques según el controlador que usen, por ejemplo, la carpeta `views/Page` contiene todas las plantillas que se renderizan en el controlador `PageController`.

También existe un número de plantillas que son sobrescritas del `FOSUserBundle`, para poder adecuarlas al diseño del sitio web, y añadir funcionalidades. Se encuentran en la ruta `app/Resources/views/FOSUserBundle`. Se incluyen aquí por especificaciones de la documentación del bundle.

La pagina principal del sitio web queda así:



Pets Friends

La Protectora | Nuestros amigos | Adoptados | Mascotas perdidas | Como adoptar

Acceso


Nombre de usuario:

Contraseña:

Recordar ☐


Entrar

Si no estas registrado haz click aqui



Miércoles, 18 de junio de 2014

Noticia Cinco



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Morbi libero dolor, adipiscing et cursus vel, egestas ut sem. Fusce sed facilisis libero, a molestie ante. Duis eget sapien id nisi varius aliquet. Aliquam malesuada quam lacus, consectetur volutpat arcu ultrices a. Nulla hendrerit ligula sit

Comentarios: 4

Enviado por admin a las 02:53PM

Etiquetas: alaskan, malamute, perro

Continúa...

Comentarios recientes

18 JUNE Admin ha comentado en Noticia Cinco hace 20 horas

Comentario 04

18 JUNE Admin ha comentado en Noticia Cinco hace 20 horas

Comentario 03

18 JUNE Admin ha comentado en Noticia Cinco hace 20 horas

Comentario 02


18 JUNE Admin ha comentado en Noticia Cinco hace 20 horas

Comentario 01

Etiquetas mas usadas

alaskan malamute bobbail perros bulldog perro ingles

Animales de la protectora




RockyIII

Perro

Pastor aleman

Macho




Piluka

Perro

Bobbail

Hembra







Tango

Perro

Alaskan malamute

Macho

PetsFriends © 2014

Desarrollo Web Entorno Cliente

1. Requisitos

- Utilizar la librería JQuery (Validación de todos los formularios)
- Utilizar la librería JQuery UI (Cuadros de diálogos)
- Utilizar las funciones JQuery AJAX (Buscadores para búsquedas con filtro, eliminación de registros en los listados, incremento/Decremento de cantidades en los listados)

2. JQuery Validation

Para el sitio web he hecho uso de JQuery Validation. Este plugin sirve para que el usuario no introduzca datos erróneos a la hora de rellenar cualquier formulario del sitio web.

El plugin puede ser descargado desde su sitio web: <http://plugins.jquery.com/validation/>, donde se incluye la documentación y ejemplos de uso.

En PetsFriends se ha hecho uso de la validación Front-End en los formularios de registro de usuario y edición del perfil.

Registro de Usuario

Email: checo1992@gmail.com ✓

Nombre de usuario: checo ✓

Entre 6 y 10 caracteres

Contraseña:

Repita la contraseña:

Contraseña no coincide

Nombre completo:

Nombre

Campo obligatorio

Apellido

Campo obligatorio

Pais: Selecciona pais... ▼

Campo obligatorio

Ciudad: Selecciona ciudad... ▼

Campo obligatorio

Nombre de tu mascota:

Mascota

Campo obligatorio

Especie: Perro ▼

Miércoles, 18 de junio de 2014

El código usado para validar los campos se encuentra en js/ajax.js, y es el siguiente:

```

ajax.js
x
422 /*****
423  */ Funcion JQuery Validate que valida los campos de */
424  */ registro de usuario
425  */
426 $(document).ready(function(e) {
427     $("#fos_user_registration_register").validate();
428     $("#fos_user_registration_form_email").rules("add", {
429         required: true,
430         email: true, messages: {
431             required: "Campo obligatorio",
432             email: "Campo con formato e-mail"
433         });
434     $("#fos_user_registration_form_username").rules("add", {
435         required: true,
436         rangelength: [6,10], messages: {
437             required: "Campo obligatorio",
438             rangelength: "Entre 6 y 10 caracteres"
439         });
440     $("#fos_user_registration_form_plainPassword_first").rules("add", {
441         required: true,
442         rangelength: [6,15], messages: {
443             required: "Campo obligatorio",
444             rangelength: "Entre 6 y 15 caracteres"
445         });
446     $("#fos_user_registration_form_plainPassword_second").rules("add", {
447         required: true,
448         equalTo: "#fos_user_registration_form_plainPassword_first", messages: {
449             required: "Campo obligatorio",
450             equalTo: "Contraseña no coincide"
451         });
452     $("#fos_user_registration_form_nombre").rules("add", {
453         required: true,
454         rangelength: [10,25], messages: {
455             required: "Campo obligatorio",

```

3. JQuery UI

JQuery UI proporciona funcionalidades extra para la creación de ventanas modales con efectos, formularios personalizados, y de campos específicos como, por ejemplo, DatePicker para incluir un calendario en campos de tipo fecha.

La página principal de JQuery UI es <http://jqueryui.com/>

Para el proyecto se han utilizado ventanas modales en el registro y en el perfil, y calendario DatePicker en formularios para creación / edición de entradas del sitio web.

En la siguiente captura se muestra un DatePicker personalizado de JQuery UI

La imagen muestra una interfaz de usuario para el registro. A la izquierda, hay un formulario con los campos: 'Fecha de registro' (con el valor 19-06-14), 'País', 'Ciudad' y 'Dirección'. A la derecha, hay un botón 'Volver'. En el centro, se muestra un calendario modal de JQuery UI para el mes de junio de 2014. El calendario tiene una estructura de tabla con días de la semana (L, M, X, J, V, S, D) y números de días (1 a 30). El día 19 está seleccionado.

Para crear las ventanas modales he utilizado el siguiente código Javascript:

```

ajax.js
58  /*****
59  /* Funciones para cargar JQuery UI Ventana Modal */
60
61  $(document).ready(function(){
62
63      // Damos formato a la Ventana de Diálogo
64      $('#dialogoPerfilError').dialog({
65          // Indica si la ventana se abre de forma automática
66          autoOpen: true,
67          // Indica si la ventana es modal
68          modal: true,
69          // Largo
70          width: 350,
71          closeOnEscape: false,
72          // Alto
73          height: 'auto',
74          position: {my: "center", at: "center", of: window},
75          resizable: false,
76          draggable: false,
77          show: { effect: "puff", duration: 800 },
78          hide: { effect: "puff", duration: 1000 },
79          open: function(event, ui) { $(".ui-dialog-titlebar-close", ui.dialog).hide(); }
80      });
81
82      $('#dialogoPerfil').dialog({
83          // Indica si la ventana se abre de forma automática
84          autoOpen: false,
85          // Indica si la ventana es modal
86          modal: true,
87          // Largo
88          width: 350,
89          // Alto
90          height: 'auto',
91          position: {my: "center", at: "center", of: window},

```

Para cargar el DatePicker se implementa de la siguiente manera:

```

ajax.js
165  /*****
166  /* Funcion para cargar JQuery UI Datepicker */
167
168  $(document).ready(function(){
169      $('#form_animal_date').datepicker({
170          dateFormat: 'dd-MM-yy'
171      });
172  });
173
174

```

4. JQuery AJAX

La funcionalidad AJAX permite poder recargar ciertas zonas del sitio web, recibir datos en algún campo de selección o hacer llamadas a la base de datos de una manera asíncrona, mientras el usuario esta visitando el sitio web y sin tener que recargar la página que esta viendo / utilizando de manera completa con cada acción. Permite crear sitios webs muy dinámicos.

En JQuery la función AJAX se ejecuta mediante la función `$.ajax()`. Los parámetros minimos que necesita son la url donde se envían los datos, type (POST o GET), data (aquí enviamos los datos que queremos modificar cuando se devuelve la llamada), y una función `success(data)`, que realiza la acción en la pagina. Opcionalmente se incluye la función `beforeSend()`, que sirve para realizar acciones mientras se completa la llamada AJAX, como mostrar un 'loading' por ejemplo.

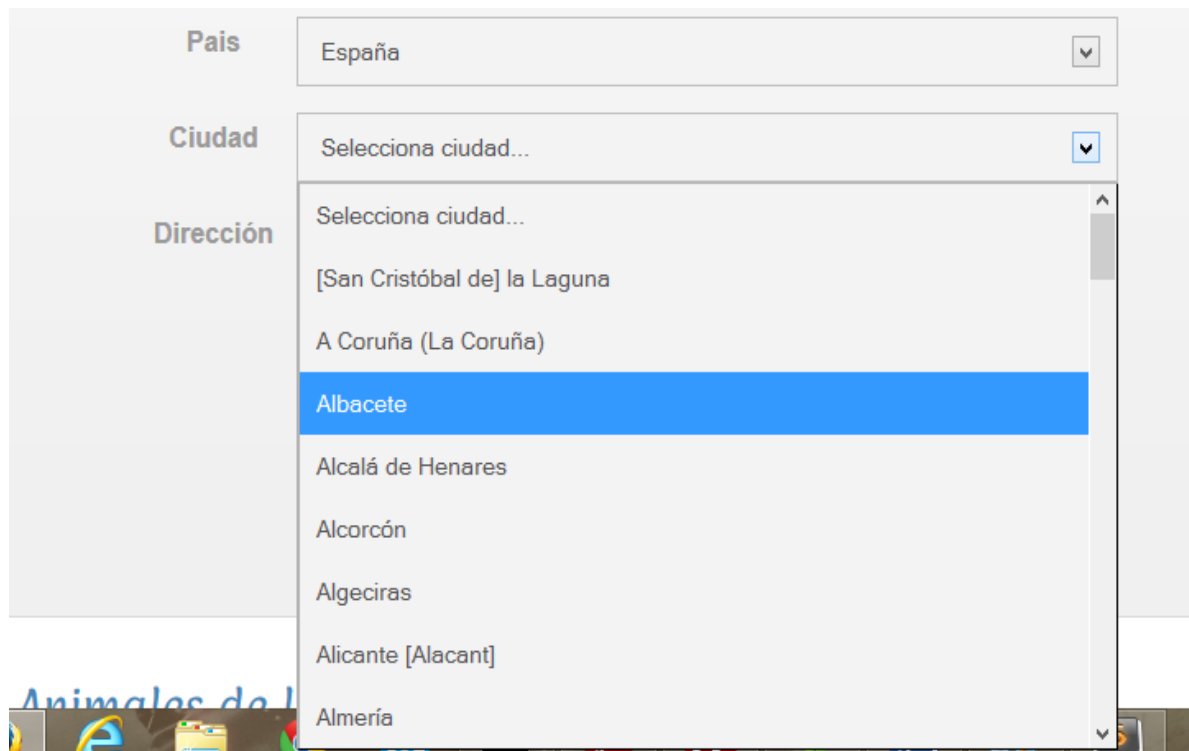
Para el proyecto final he utilizado AJAX para validar en tiempo real si un nombre de usuario o su email están ya registrados en la BBDD. Tambien he usado AJAX para cargar las ciudades de un determinado país elegido por el usuario en un select form. Otra funcionalidad AJAX permite cambiar al usuario su foto de perfil de forma dinámica. Por último se ha usado AJAX junto con la paginación de las entradas del sitio web.

Por ejemplo para cargar las ciudades, el código AJAX seria el siguiente:

```

132  /*****
133  /* Funcion para cargar Ciudades segun el Pais */
134
135  $(document).ready(function(){
136      var $pais = $('country');
137
138      $pais.change(function() {
139
140          var $form = $(this).closest('form');
141
142          var data = {};
143          data[$pais.attr('name')] = $pais.val();
144
145          $.ajax({
146              url : $form.attr('action'),
147              type: $form.attr('method'),
148              data : data,
149              dataType: 'json',
150              success: function(data) {
151                  $('city')
152                  .empty()
153                  .append('<option value="">Selecciona ciudad...</option>')
154                  .find('option:first')
155                  .attr("selected","selected");
156                  $.each(data.ciudades, function(i, value) {
157                      $('city').append('<option value='+value.id+'>'+value.ciudad+'</option>');
158                  });
159              }
160          });
161      });
162  });
163  */

```



The image shows a web form with three dropdown menus. The first menu, labeled 'Pais', has 'España' selected. The second menu, labeled 'Ciudad', is open and shows a list of cities. The third menu, labeled 'Dirección', is also open and shows a list of cities. The city 'Albacete' is selected in both the 'Ciudad' and 'Dirección' menus.

Field	Value
Pais	España
Ciudad	Albacete
Dirección	Albacete

Al elegir el país, nos carga de forma automática las ciudades de la BBDD relacionadas. Así impedimos que se elija una ciudad que no pertenece a ese país.

Todas las funciones Javascript empleadas en el sitio web se encuentran en el fichero `js/ajax.js`

Despliegue de Aplicaciones Web

1. Requisitos

- El proyecto estará alojado en un hosting compartido gratuito. Se recomienda la utilización de hostinger.es por sus buenas prestaciones y fiabilidad.
- El proyecto estará incluido en un directorio llamado /proyecto que pende del directorio raíz.
- Al escribir el nombre de dominio (por ejemplo nombre_alumno.hol.es) se redireccionará al directorio raíz.
- Se creará un directorio denominado "/documentación" protegido por contraseña "basic" que contendrá toda la documentación que se genere para el desarrollo del proyecto, una copia de seguridad del sitio y la base de datos. Se facilitará al usuario la navegación por ese directorio. Al directorio se podrá entrar tecleando en el navegador documentacion.nombre_alumno.hol.es
- Utilizando los medios necesarios se impedirá que otros usuarios hagan "hotlinking" de las imágenes del sitio.
- Se creará un usuario ftp denominado profesor que tendrá acceso directo al directorio /documentación.
- Todos los pasos para elaborar los pasos anteriores serán debidamente documentados y guardados en el directorio anteriormente citado. En este documento se completará la siguiente información: Nombre del dominio seleccionado, nombre de usuario y contraseña del hosting y nombres de usuarios y contraseñas del servidor ftp.

2. Datos de acceso

Nombre del dominio	http://checo1983.alwaysdata.net
Usuario FTP Profesor	checo1983_profesor
Contraseña FTP Profesor	ciphron
Usuario Documentación	checo
Contraseña Documentación	ciphron
Versión de PHP	5.5
Versión MySQL	5.1.61
Versión Apache	Apache 2.2.14

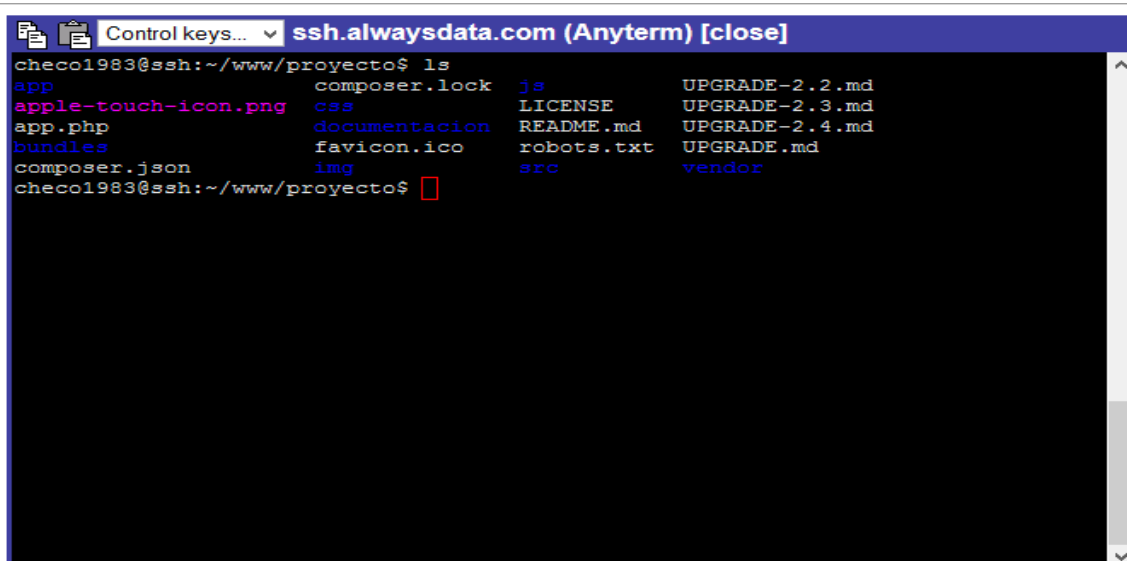
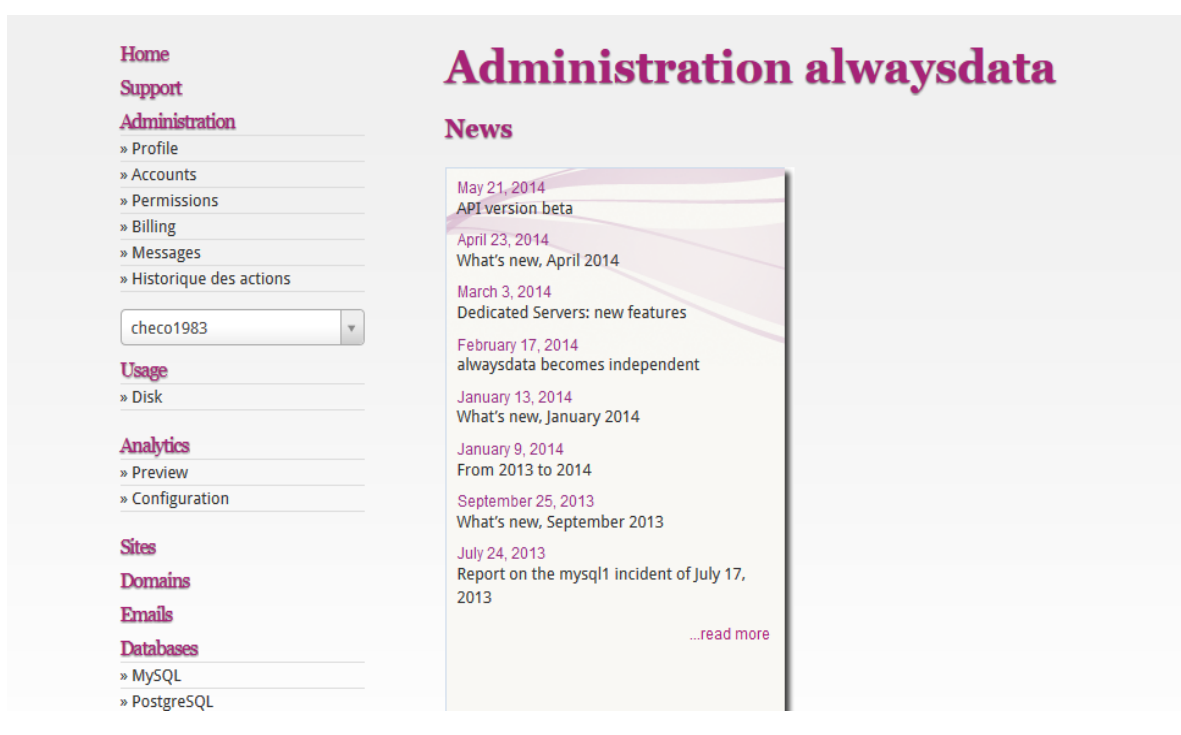
3. Alojamiento del proyecto

Mi proyecto esta desarrollado en PHP con el framework Symfony2. Todo el desarrollo y las pruebas las he realizado con el paquete de aplicaciones Xampp bajo Windows. Al subirlo al servidor Hostinger en modo producción, correctamente configurado y probado en el servidor local, me aparece la página totalmente en blanco.

Tras probar en otros dos hostings gratuitos con idéntico resultado, me dispongo a montar un servidor virtual con Ubuntu, y tampoco consigo que funcione ni en modo de desarrollo ni por supuesto en producción.

Finalmente encuentro un servidor alojado en Francia www.alwaysdata.com

Este servidor no cuenta con un panel de control tan sencillo de manejar, pero permite acceder via SSH para poder configurar el servidor a nuestro gusto.



4. Redirección

En el caso de que queramos realizar una redirección desde otra carpeta hacia el index de nuestro proyecto lo hacemos de la siguiente forma.

Creamos un archivo .htaccess en el directorio raíz y escribimos lo siguiente:

```
#redireccion http://checo1983.alwaysdata.net/proyecto
RewriteEngine On
RewriteRule ^(.*)$ http://checo1983.alwaysdata.net/proyecto/$1 [R=301,L]
```

5. Directorio Documentación Encriptado

Creamos dos archivos, uno .htaccess y el otro .htpasswd. En el de password añadimos el usuario y la contraseña encriptada, y lo colocamos en la raíz del proyecto por ejemplo.

Seria algo asi:

```
checo:MMUdRT74Bb98c
```

En el archivo .htaccess debemos añadir lo siguiente:

```
AuthUserFile /www/proyecto/.htpasswd
AuthType Basic
AuthName "Restringido"
require valid-user
```

Y guardarlo en la carpeta que queramos proteger, en nuestro caso documentación.

6. Protección Hotlinking


El hotlinking es usar una imagen de una página web en otra usando la url del sitio original. Esto provoca que la imagen se descargue desde tu servidor y haciendo que consumas transferencia mensual de datos del hosting que tengas contratado.

Para proteger nuestras imágenes añadimos estas líneas a nuestro .htaccess

```
RewriteEngine On
RewriteBase /
RewriteCond %{HTTP_REFERER}!^http://(.+\.)?http://checo1983.alwaysdata.net/proyecto
[NC]
```

7. Acceso FTP para el profesor





Por último creamos un usuario FTP profesor que tenga acceso directo a la carpeta documentación, accediendo a Archivos / Cuentas FTP.

 The **checo1983_profesor** FTP user was successfully created.

FTP users

Hôte FTP : ftp.alwaysdata.com

+ Add FTP user

Name	Root directory	Edit	Delete
checo1983	/		
checo1983_profesor	/proyecto/documentacion		

Diseño de Interfaces WEB

1. Requisitos

- Utilizar la hoja de estilos CSS3 para dar formato a la interfaz, en el caso de ser utilizado algún framework tipo Bootstrap, crear el código para desarrollar alguno de estos elementos: - Galería de imágenes. - Banner.
- El diseño de la interfaz deberá ser "Responsive" utilizando Media Queries.
- Insertar un elemento multimedia de cada tipo: vídeo, sonido, canvas.
- Utilizar Photoshop para crear una cabecera con un fondo y un título en la cual se aplicarán deferentes efectos y filtros.
- Crear una imagen para el fondo de la página web con un motivo, aplicándole efectos y filtros.

2. Interfaz de PetsFriends

Para el desarrollo de la interfaz del sitio web he hecho uso de una plantilla base, denominada Pet Club Free CSS Template <http://www.free-css.com/free-css-templates/page172/pet-club>, a la cual he ido adaptando y añadiendo estilos personalizados hasta conseguir el resultado final del sitio web.

Los colores que predominan en la interfaz son el verde, el naranja, el negro y el gris claro.

La tipografía empleada es Arial en general, los botones emplean Noto-Sans y los títulos son de la fuente Paprika. Ambas fuentes son de Google Font. El color de la fuente usada presenta blanco, negro y grisáceo.

Para la cabecera he utilizado fondo negro sobre letra en blanco, con un logo creado en Photoshop, y un acceso login en la zona superior izquierda.

Debajo de la barra de navegación se ha incluido una animación CSS3. Las fotos de la galería se van sucediendo de forma automática cada 5 segundos.

El background del sitio web combina blanco y gris. Los iconos utilizados son de tonos verdes y naranjas.

El carrusel de imágenes inferior esta creado con un plugin JS, y estilizado usando los estilos de la plantilla inicial.

Al pie de página se ha añadido una pequeña animación canvas, con efecto de ola.

3. Logo de PetsFriends

Para el logo he utilizado un background negro de 500x213 px. He creado cuatro capas adicionales. En la primera capa he añadido la palabra PETS, con un efecto difuminado. A la primera letra le he dado un tono Verde para destacarla del resto.



En la segunda capa he añadido la palabra FRIENDS, de igual forma que en la primera capa, pero esta vez he usado Naranja para la primera letra. En las dos siguientes capas he utilizado los iconos característicos del sitio web, para acoplarlos al conjunto del logotipo. El resultado obtenido es el que se muestra a continuación:



4. Animación Canvas

Para la etiqueta Canvas se ha desarrollado una pequeña animación en la parte inferior, imitando el efecto de una ola con movimiento de izquierda a derecha.

El código para la animación es el siguiente:

```

parameters.yml  x  canvas.js  x
3  /*      Funciones de Canvas.js      */
4  /*      Contiene el código javascript de la animación      */
5  /*      canvas del sitio web PetsFriends      */
6  /*      */
7  /*      */
8
9  var velocidad = 0.2; /* Velocidad y sentido de la ola */
10 var altoOla = 50; /* Altura de la ola */
11 var anchoOla = 0; /* Ancho de la ola */
12
13 var myAarray = new Array; /* Almacena cada paso de la animación */
14
15 /* Aquí pintamos el canvas en cada iteración */
16 function pintarCanvas() {
17     var contador = 0;
18     var posicion = 0;
19
20     var incrementoOla = Math.PI / 90 + anchoOla; /* Curvatura de la ola */
21
22     /* Tamaño del contenedor canvas */
23     var canvas = document.getElementById("canvas")
24     canvas.width = document.getElementById("bottom_block").clientWidth;
25     canvas.height = document.getElementById("bottom_block").clientHeight-20;
26
27     for (i = 0; i <= canvas.width; i++) {
28         y = Math.sin(contador + velocidad);
29         contador += incrementoOla;
30         var fin = 150 + y * altoOla; //Obtiene el final de la línea a pintar
31
32         //Entra la primera vez para iniciar el canvas
33         if (myAarray[i] == undefined || myAarray[i] == '' ) {
34             myAarray[i] = canvas.getContext('2d');
35         }
36
37         //Pinta el canvas
38         myAarray[i].beginPath();
39         myAarray[i].linewidth="5";
40         myAarray[i].strokeStyle="white";
41         myAarray[i].moveTo(0+posicion,0);
42         myAarray[i].lineTo(posicion,fin);
43         myAarray[i].stroke();
44         posicion += 5;
45     }
46
47     velocidad += 0.1;
48 }
49
50 setInterval('pintarCanvas()',50);

```

Para completar este módulo he creado un pequeño video de presentación del sitio web con PowToons. La versión gratuita no permite la descarga de los videos creados, por lo que dejo el enlace en esta documentación para poder verlo.

<http://www.powtoon.com/embed/bxNnoA3lSux/>

Empresa e Iniciativa Emprendedora

1. Requisitos

- Idea de negocio: Su inclusión o no dentro de los nuevos yacimientos de empleo.
- Descripción de la empresa: denominación social, evolución de la empresa, ubicación, número de trabajadores, organigrama, actividad social, volumen de trabajo...
- Indicar la forma jurídica de la empresa.
- Localizar el establecimiento y realizar un breve estudio de la competencia.
- Financiación necesaria (según la forma jurídica) y forma de obtención de la misma, si fuera posible obtener ese dato.
- Relación de los trámites necesarios para la constitución y puesta en marcha del negocio.
- Valoración personal de la FCT y su relación con lo estudiado en clase. Se trata de comprobar que lo que hemos visto en el módulo tiene su aplicación práctica en el mundo laboral, o no.
- Fecha final de entrega 16 de mayo.

8. Idea de negocio

CIPHON GmbH se encuentra actualmente dentro de las siguientes tres áreas de negocio:

- Seguridad IT
- Comunicaciones
- Ingeniería


La seguridad informática es el área de negocio donde CIPHON comienza los primeros años, y aún es un campo muy importante hoy en día. La experiencia adquirida se expande, por supuesto, a las otras actividades.

Además de consejos generales sobre Seguridad IT que ofrecen a sus clientes, realizan pruebas de penetración y evaluación de redes. Posee clientes extremadamente sensibles en temas de seguridad como la NBank (Importante Banco Alemán)

CIPHON posee un sistema de comunicaciones basado en VoicelP (Voz a través de internet), denominado PBX VOIPUP que no sólo proporciona seguridad, sino también un alto grado de facilidad de uso. Es conveniente para las empresas independientemente de su número de empleados, ofrece un manejo intuitivo con una gran flexibilidad y capacidad de adaptación.

En Comunicaciones también hacen uso del sistema de tickets OTRS. Este sistema se utiliza en las empresas IT por la sencillez y la estructuración del sistema. Permite a cada cliente notificar las incidencias en sus sistemas informáticos, en forma de tickets. Cuando llega la notificación, el ingeniero dispone de todos los datos del cliente y puede contactar con él para ampliar la información sobre la incidencia, o comenzar a solucionarla.

La tercera área de negocio es la ingeniería. En el departamento de desarrollo se realizan aplicaciones desde cero, y también ajustan las aplicaciones de código abierto existentes a las características individuales de cada cliente.



Esta cartera de soluciones y el servicio se completa con consultoría IT, así como servicios de Linux y Unix. Para satisfacer las demandas de los clientes en este sector tan dinámico de las necesidades de software de código abierto, trabajan constantemente en la ampliación de su cartera de clientes.

CIPHON GmbH tiene una idea muy definida sobre el área de negocio. Es un sector con gran competitividad, pero muy rentable y con gran expansión. El sector de las IT es un sector de gran futuro, cada vez es más necesario en nuestras vidas la comunicación.

9. Descripción de la empresa

CIPHON GmbH como empresa está en continuo crecimiento desde su fundación en 2003, por lo que siempre están buscando nuevos empleados calificados, creativos y motivados.

Posee un gran valor atractivo para sus clientes, también un gran énfasis en las necesidades de nuestros empleados a la empresa: un ambiente de trabajo agradable y acogedor, informal, y donde el trabajo se caracteriza por un alto grado de compañerismo.

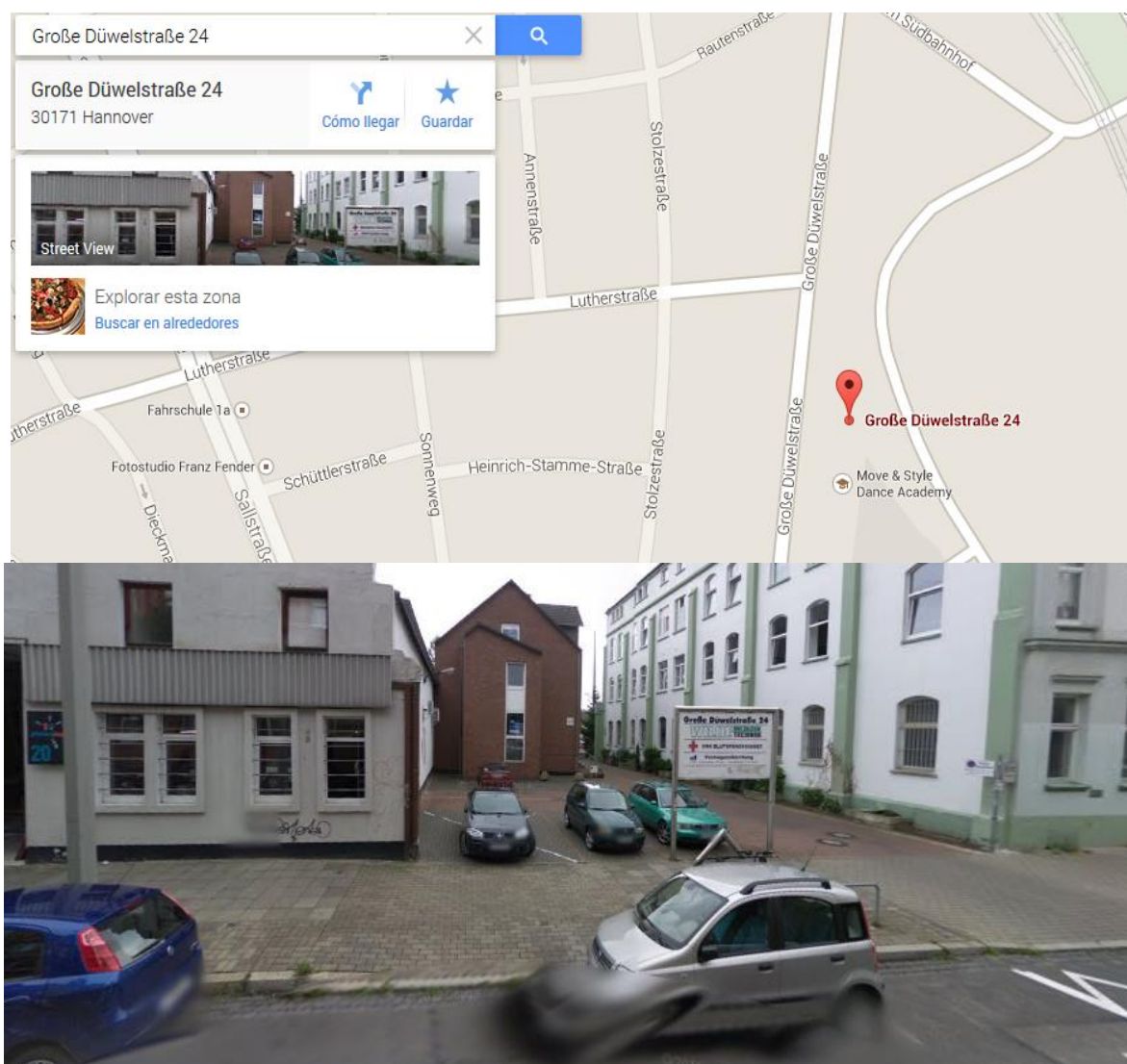
El equipo de CIPHON GmbH es diverso: se compone en gran parte de un grupo joven y dinámico los desarrolladores de aplicaciones e integradores, que forman la columna vertebral de la empresa. Éstos son apoyados por especialistas en las áreas de ventas, marketing, administración y diseño.

A la cabeza de la empresa se encuentra Sebastián Horzela, fundador de la compañía y actual CEO. Durante los dos primeros años de vida de la empresa, fue el único socio. Después aparece la figura de Kai Hoerner, socio capitalista y jefe del departamento de desarrollo. En este departamento trabaja codo a codo con los programadores informáticos de la empresa.

La empresa se completa con un área especializada en monitorización de servidores web Linux, y como he comentado anteriormente, marketing y administración.

10. Localización y forma jurídica de la empresa

La empresa CIPHON GmbH tiene su sede única en la ciudad de Hannover (Alemania), en concreto en Große Düwelstr. 24, 30171 Hannover. La zona se denomina Braunschweiger Platz donde llegan hasta cinco líneas de tranvía y diversos autobuses. La localización exacta se detalla a continuación en el siguiente mapa:



En relación a su forma jurídica, la empresa CIPHON GmbH es una sociedad de responsabilidad limitada, de ahí las siglas (GmbH). En Alemania también se denomina sociedad unipersonal de responsabilidad limitada (UG haftungsbeschränkt).

Esta forma jurídica alemana permite establecer una sociedad sin necesidad de un capital nominal mínimo específico. Está pensada para satisfacer las necesidades de los socios fundadores que en sus comienzos solo cuentan con un capital nominal bajo.

La GmbH es una sociedad mercantil con una estructura empresarial y su personalidad legal propia. A diferencia de las sociedades personalistas, el foco no se centra en una asociación de personas, sino en la contribución de capital. Puede establecerse para cualquier propósito admisible. Dispone de un capital nominal que queda determinado por el memorándum y los estatutos de asociación y se corresponde con las contribuciones originales de capital que deben efectuar los accionistas. Si la empresa contrae deudas, la responsabilidad frente a los acreedores recae exclusivamente en la empresa.

La GmbH es la figura más simple de una empresa en Alemania. Una ventaja todavía más relevante de su forma jurídica es su flexibilidad. Existe una trascendental libertad en relación a la redacción del acuerdo de accionistas. La GmbH es igual de adecuada para pequeñas empresas, medianas empresas familiares y grandes empresas.

La GmbH es una entidad legalmente independiente. Está organizada como una sociedad de tipo capitalista, independiente de los miembros y dispone de una organización con al menos dos órganos independientes: el/los director/es administrativos y la junta de accionistas. Es admisible la designación de un consejo de vigilancia, aunque no es necesario en todos los casos.

El capital nominal de la GmbH asciende a como mínimo 25.000 euros. El capital nominal de la UG (de responsabilidad limitada) asciende a como mínimo 1 euro. Dado que este último es, sin embargo, para acumular progresivamente el capital mínimo de la GmbH normal, podría no distribuir completamente los beneficios, aunque debe transferir un cuarto de los ingresos netos del año a una reserva legal hasta que se alcance el capital mínimo legal de 25.000 euros.

11. Trámites necesarios para la puesta en marcha

No se prescribe ningún número mínimo o máximo de accionistas. También es posible la constitución con un único socio. Los socios fundadores de una GmbH pueden ser personas naturales y jurídicas alemanas y extranjeras así como sociedades personalistas mercantiles (oHG, KG y EEIG) y sociedades de derecho civil (GbR).

Es necesaria la legitimación notarial del acuerdo de la GmbH así como el registro del establecimiento de la empresa. El acuerdo de los accionistas debe incluir como mínimo la siguiente información: la razón social y la sede social de la empresa, el objeto de la empresa, el montante del capital nominal, el montante de las contribuciones del capital original que deben ser efectuadas por cada accionista (contribución del capital original).

El establecimiento con el registro estándar más económico sólo es posible:

- Si la empresa se constituye con un máximo de tres accionistas. A partir de cuatro accionistas, el establecimiento sólo es posible a través de un acuerdo de accionistas individual y legitimizado ante notario.
- Si los accionistas se ponen de acuerdo sobre un máximo de un director administrativo. En ese caso, dicho director administrativo cuenta con poder de representación de la empresa por sí solo.
- Si el director administrativo está exento de la prohibición de la autocontratación (p. ej., el director administrativo debe suscribir transacciones de la UG (de responsabilidad limitada) o la GmbH con él mismo como persona privada o como representante de otra persona).
- El establecimiento a través del registro estándar sólo es posible en caso de constitución por suscripción al contado. La constitución por contribución de capital que no sea en metálico no será posible dentro del marco de la constitución a través del registro estándar.

El capital nominal de la GmbH, como se ha expuesto anteriormente, asciende a un mínimo 25.000 euros.

Las contribuciones del capital original pueden aportarse con dinero (constitución por suscripción al contado) o a través de contribuciones en especie (constitución por contribuciones de capital que no son en metálico). Debe depositarse al menos un cuarto de cada contribución de capital original en metálico. La solicitud de inscripción en el registro mercantil podría no producirse a menos que las contribuciones de capital hayan alcanzado como mínimo la mitad del capital nominal mínimo, es decir, 12.500 euros.

La razón social es el nombre bajo el cual la GmbH queda registrada en el registro mercantil y bajo el que aparecerá en las transacciones comerciales. La razón social de la GmbH puede hacer referencia a la actividad de la empresa (nombre objetivo de la empresa), incluir el nombre de uno o más accionistas (nombre personal de la empresa) o basarse en una denominación inventada. Es posible combinar dichos elementos; el nombre objetivo de la empresa deberá ir provisto de una adición que lo distinga. La adición Sociedad Limitada o la abreviatura GmbH es un componente obligatorio para la razón social. La adición de responsabilidad limitada no debe abreviarse.

12. Valoración personal de la FCT

Mi valoración es totalmente positiva, superando todas las expectativas que tenía antes de empezar. La acogida y presentación como un miembro más del equipo son de agradecer puesto que parece que el personal en prácticas tradicionalmente ha sido infravalorado, en cambio desde un principio me he sentido integrado en la empresa. En las empresas alemanas, la acogida de empleados en prácticas es muy buena. Siempre intentan ayudarte en todo lo posible, y aconsejándote como poder realizar tu trabajo de manera más eficiente.

También es positivo el hecho de que no haya tenido que hacer el trabajo con menos responsabilidad o que nadie quería hacer, el reparto de tareas se hace por igual tanto entre personal en prácticas como personal de la empresa. Aunque CIPHRON GmbH sea una empresa pequeña, gestiona proyectos de gran envergadura y tuve la posibilidad de conocer la forma de trabajar en cada uno de ellos. Por lo tanto el aprendizaje ha sido mayor de lo que esperaba. Aunque quizás no haya profundizado en ningún departamento en concreto, el hecho de ver como se coordinan será muy útil para mi futura inclusión en el sector.

Recomiendo CIPHRON GmbH para futuros estudiantes en prácticas puesto que es una empresa que se implica en nuestra formación y valora el esfuerzo de cada uno. Se trata de una empresa sólida que siempre busca nuevos talentos, que está en pleno crecimiento, por lo que es una buena oportunidad para entrar en el sector en una empresa reconocida.

Es flexible con los alumnos en prácticas que tengan algún tipo de compromiso, por ejemplo, si tienes algún imprevisto, o visitar al médico. Tienen un horario muy flexible, e incluso poseen trabajadores a tiempo parcial que desempeñan su tarea desde su propia casa. Todo son facilidades para que puedas asistir a todo con causa justificada.

Inglés

1. Requisitos

- Hacer un listado de las funcionalidades de la aplicación en inglés y español y añadirlo con enlace en la aplicación web.

2. Proyecto PetsFriends en español

El objetivo del sitio web es proveer a cualquier tipo de asociación protectora de animales, un espacio en internet para darse a conocer, publicar noticias, eventos e información relacionada con sus animales. También permite a cualquier usuario solicitar la adopción de algún animal de la asociación y / o informar sobre la desaparición de algún animal al resto de la asociación.

El sitio web contiene áreas de acceso limitado solo al administrador del sitio web.

El usuario anónimo puede visitar la sección de noticias, de animales y mascotas perdidas, y escribir en el formulario de contacto, además de registrarse.

El usuario registrado puede escribir comentarios en las noticias, solicitar adopciones, y añadir nuevas entradas en la zona de mascotas perdidas.

El administrador del sitio tiene acceso a toda la web. Puede crear nuevas noticias, puede dar de alta a nuevos animales de la asociación y en su cuenta email recibe todas las notificaciones sobre el sitio web.

3. Proyecto PetsFriends en inglés

The purpose of the web site is to provide to any type of protective association of animals, a space in Internet to be announced, to publish news, events and information related to animals. It allows to any user to request the adoption of some animal of the association and / or to report on the disappearance of some animal to the rest of the association.

The web site contains areas of limited access only for the administrator of the web site.

The anonymous user can visit the section of news, animals and lost pets, and to write in the form of contact, besides to be registered.

The registered user can write comments in the news, request adoptions, and add new entries in the area of lost pets.

The administrator of the site has access to all the web. He can create news, sign up to new animals of the association and in his account e-mail, he can receive all the notifications on the web site.