



PROTECTORA DE ANIMALES

Alberto Ortega Padilla

Proyecto Final DAW

1. Manual del Usuario

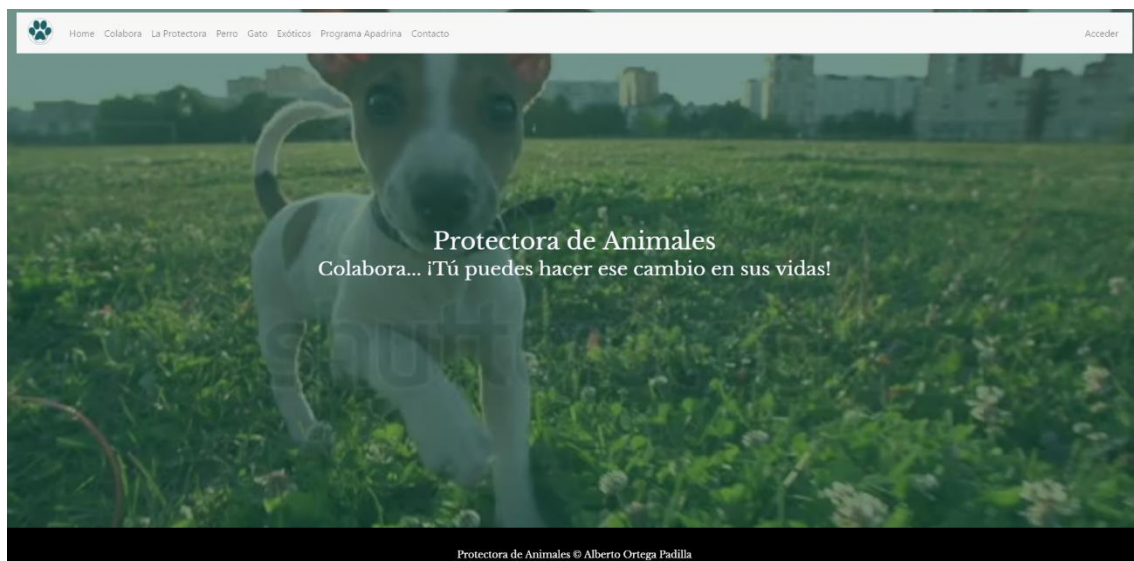
1.1. Descripción de la aplicación.

La aplicación web protectora de animales, es una plataforma cuya finalidad es facilitar la adopción, apadrinamiento, voluntariado y donación en una asociación que su función sea un refugio de animales.

La aplicación se divide en dos partes: una parte publica, que muestra los datos de la protectora, contacto, la selección de animales que tienen en el refugio (gatos, perros, exóticos, apadrina), para ser apadrinados o adoptados, etc. Después, la parte privada, que se accede desde el apartado del logueado, donde accede a la administración de los animales que se muestran en la parte pública.

1.2. Funcionalidades y características.

- Portada con video de fondo.



- Información de colaboración, historia de la protectora y contacto.

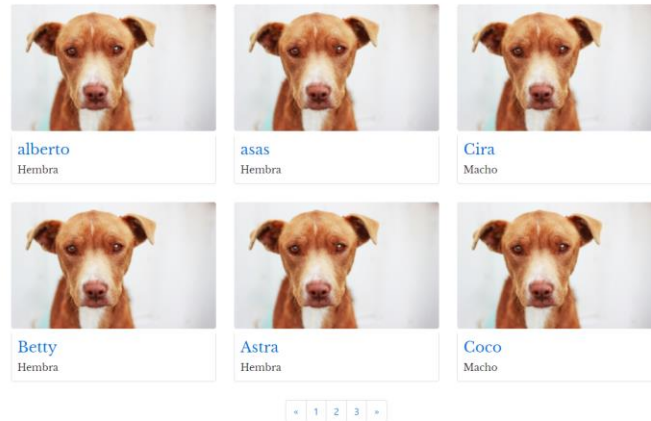


- Listado con paginación de gatos, detalle de gato con imágenes en Scroll y video oculto para la muestra de la página (para que tenga menos peso de carga) con botón de mostrar y ocultar video del animal (video conectado a una cuenta en YouTube).



- Listado con paginación de perros, detalle de perro con imágenes en Scroll y video oculto para la muestra de la página (para que tenga menos peso de carga) con botón de mostrar y ocultar video del animal (video conectado a una cuenta en YouTube).

PERROS EN ADOPCIÓN

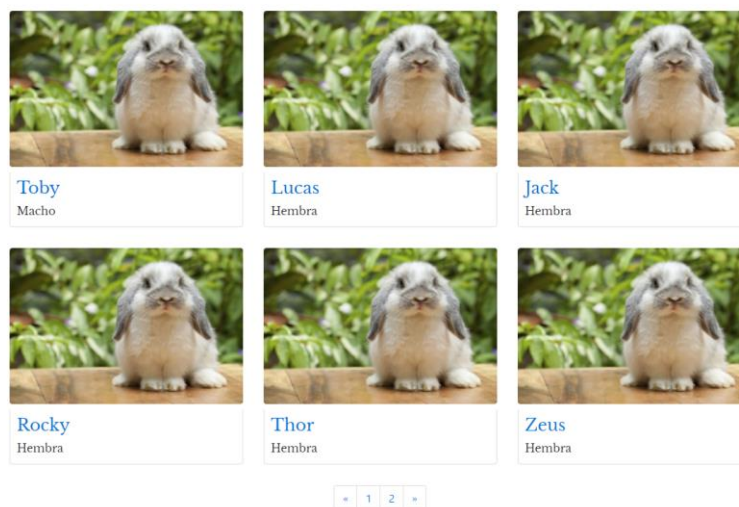


Protectora de Animales © Alberto Ortega Padilla

- Listado con paginación de exóticos, detalle de exótico con imágenes en Scroll y video oculto para la muestra de la página (para que tenga menos peso de carga) con botón de mostrar y ocultar video del animal (video conectado a una cuenta en YouTube).

▪

EXÓTICOS EN ADOPCIÓN



Protectora de Animales © Alberto Ortega Padilla

- Logueado para la parte administrativa de los animales.

Home Colabora La Protectora Perro Gato Exóticos Programa Apadrina Contacto Acceder

Login not Valid!!

Login

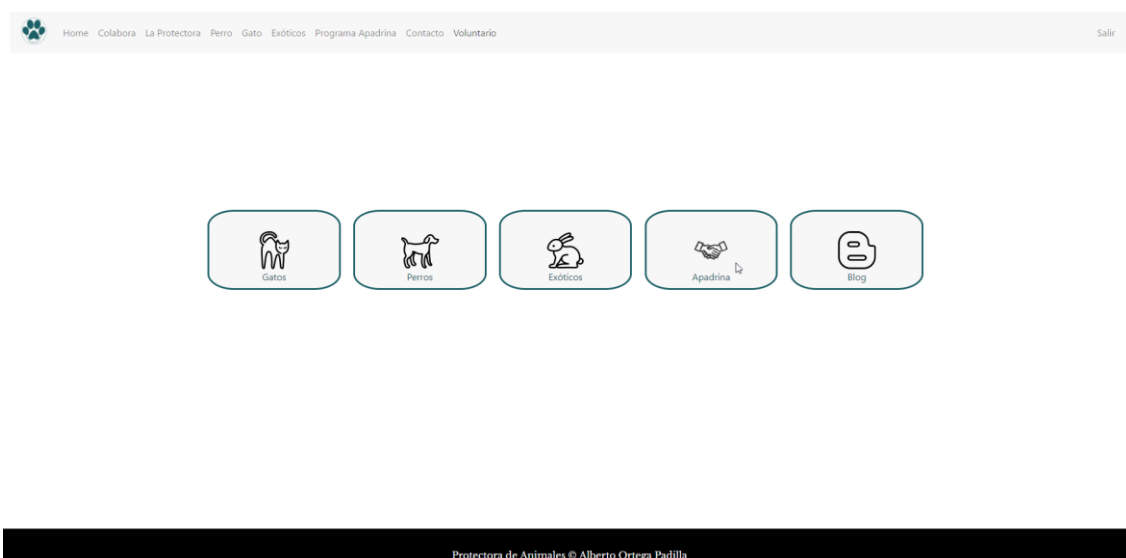
Email:
jane.doe@example.com

Password:
Password

Login

Protectora de Animales © Alberto Ortega Padilla

- Menú de los módulos animales (solamente activos los 3 primeros) y destacar que este apartado privado esta responsive hasta Tablet.



- Tabla paginada de gatos, con opciones de añadir, eliminar, modificar y buscador:
 - ✓ Añadir se encuentra oculto como en el video en detalle del animal, con botón de añadir y ocultar el formulario con validación de cada input.
 - ✓ Botón de eliminar animal.
 - ✓ Botón de modificar, que abre externamente un formulario con los detalles de la base de datos de ese gato ya introducidos, obligando al usuario a revisar antes de aceptar los cambios y comprobar si otros están correctos.

Home Colabora La Protectora Perro Gato Exóticos Programa Apadrina Contacto Voluntario Sair

¿Que quieres ver?

Buscar Añadir

Listado Gatos

NOMBRE	RAZA	GENERO	TAMAÑO	STATUS	EDAD	TEXTO	CONTACTO	TELEFONO	CORREO	VIDEO	CREADO		
Suli	Mestizo	Hembra	Mediano	Adopción	1	Qué dise u...	Alberto	123456787	alb@aorte...	<iframe w...	Alber	Editar	Eliminar
Felipe	Europeo	Macho	Mediano	Adoptado	3	Aliqua me...	Luna	578433910	luna@aort...	<iframe w...	Alber	Editar	Eliminar
Tana	Ruso	Hembra	Mediano	Adopción	3	Caballo bl...	África	878926900	afri@aort...	<iframe w...	Alber	Editar	Eliminar
Sheldon	Siamés	Macho	Mediano	Adopción	4	Lorem fist...	Adrián	123445678	adri@aort...	<iframe w...	Alber	Editar	Eliminar
Isabela	Mestizo	Hembra	Mediano	Adopción	3	Lorem fist...	Luna	578433910	luna@aort...	<iframe w...	Alber	Editar	Eliminar
Lynx	Siamés	Hembra	Pequeño	Adopción	1	Lorem fist...	Lorena	382910405	lore@aort...	<iframe w...	Alber	Editar	Eliminar

1 2

Protectora de Animales © Alberto Ortega Padilla

- Tabla paginada de perros, con opciones de añadir, eliminar, modificar y buscador:
 - ✓ Añadir se encuentra oculto como en el video en detalle del animal, con botón de añadir y ocultar el formulario con validación de cada input.
 - ✓ Botón de eliminar animal.
 - ✓ Botón de modificar, que abre externamente un formulario con los detalles de la base de datos de ese perro ya introducidos, obligando al usuario a revisar antes de aceptar los cambios y comprobar si otros están correctos.

¿Que quieres ver?

Listado Perros

NOMBRE	RAZA	GENERO	TAMAÑO	STATUS	EDAD	TEXTO	CONTACTO	TELEFONO	CORREO	VIDEO	CREADO		
alberto	sdddfs	Hembra	Grande	Adoptado	2	1szx	esd	12344567...	es@es.com	<iframe w...	Alber	Editar	Eliminar
asas	ssasa	Hembra	Grande	Adoptado	1	assasa	asa	123456789	asta@asta...	asss	Alber	Editar	Eliminar
Cira	Persa	Macho	Grande	Adopción	1	Gato gord...	Tomm	76585483	tomm@t.c...	as	Alber	Editar	Eliminar
Betty	Persa	Hembra	Pequeño	Adopción	2	Gato gord...	Tomm	765854834	tomm@t.c...		Alber	Editar	Eliminar
Astra	Persa	Hembra	Pequeño	Adopción	2	Gato gord...	Tomm	765854834	tomm@t.c...		Alber	Editar	Eliminar
Coco	Persa	Macho	Pequeño	Adopción	2	Gato gord...	Tomm	765854834	tomm@t.c...		Alber	Editar	Eliminar

« 1 2 3 »

- Tabla paginada de exóticos, con opciones de añadir, eliminar, modificar y buscador:
 - ✓ Añadir se encuentra oculto como en el video en detalle del animal, con botón de añadir y ocultar el formulario con validación de cada input.
 - ✓ Botón de eliminar animal.
 - ✓ Botón de modificar, que abre externamente un formulario con los detalles de la base de datos de ese exótico ya introducidos, obligando al usuario a revisar antes de aceptar los cambios, si otros están correctos.

¿Que quieres ver?

Listado Exoticos

NOMBRE	CLASE	RAZA	GENERO	TAMAÑO	ESTADO	EDAD	TEXTO	CONTACTO	TELEFONO	CORREO	VIDEO	CREADO		
Toby	Conejo	Belier	Macho	Mediano	Adopción	2	Gato gor...	Tomm	765854834	tomm@t...	algo	Alber	Editar	Eliminar
Lucas	Hámsteres		Hembra	Pequeño	Adopción	2	Gato gor...	Tomm	765854834	tomm@t...		Alber	Editar	Eliminar
Jack	Hurones		Hembra	Pequeño	Adopción	2	Gato gor...	Tomm	765854834	tomm@t...		Alber	Editar	Eliminar
Rocky	Hurones		Hembra	Pequeño	Adopción	2	Gato gor...	Tomm	765854834	tomm@t...		Alber	Editar	Eliminar
Thor	Hámsteres		Hembra	Pequeño	Adopción	2	Gato gor...	Tomm	765854834	tomm@t...		Alber	Editar	Eliminar
Zeus	Conejo		Hembra	Pequeño	Adopción	2	Gato gor...	Tomm	765854834	tomm@t...		Alber	Editar	Eliminar

« 1 2 »

1.3. Requisitos mínimos.

Como es a través del navegador, el único requisito es que el navegador, pero que toda la funcionalidad de la interfaz funcione. Otro requisito es que para la parte publica se puede visualizar en móvil, Tablet, y ordenador. Mientras que la parte privada, solo esta optimizada para ordenador y Tablet.

2. Manual técnico.

2.1. Entorno servidor.

2.1.1.1. Modelo de datos.

La base de datos está hecha en MySQL (sistema de gestión de bases de datos relacional), administrada en Apache, en PhpMyAdmin.

2.1.1.2. Estructura y funcionamiento.

Este entorno está hecho con PHP y con el Framework Symfony 3, en el cual se basa en el modelo, vista, controlador, pero como este entorno es solo servidor, no tiene interfaz, solamente uso el apartado del modelo y del controlador.

La estructura de este apartado es:

- ✓ **App:** configuración, y traducciones de la aplicación.
- ✓ **Bin:** este directorio reside el ejecutable de la consola, que sirve para llamar a las utilidades del cliente desde nuestro terminal.
- ✓ **Src:** es el código fuente de la aplicación, donde contiene los entities, bundles, controladores, etc.
- ✓ **Test:** almacena los sources de test, básicamente una réplica de src.

- ✓ **Var:** directorio temporal, aquí se almacena el log, cache, sesión, etc.

- ✓ **Vendor:** contiene las librerías externas que utilizo, así como el core de Symfony, básicamente es el directorio que utiliza el composer para mantener las dependencias.

- ▶ app
- ▶ bin
- ▼ src
 - ▼ AppBundle
 - ▼ Controller
 - BlogController.php
 - CatController.php
 - DefaultController.php
 - DogController.php
 - ExoticController.php
 - GenderController.php
 - RoleController.php
 - SizeController.php
 - SponsorController.php
 - StatusController.php
 - TypeAnimalController.php
 - TypeBlogController.php
 - TypeExoticController.php
 - UserController.php
 - ▼ Resources
 - ▼ config
 - ▶ routing
 - routing.yml
 - ▼ Services
 - Helpers.php
 - JwtAuth.php
 - AppBundle.php
 - ▼ BackendBundle
 - ▶ Controller
 - ▼ Entity
 - Blog.php
 - Cat.php
 - Dog.php
 - Exotic.php
 - Gender.php
 - ImgAnimal.php
 - ImgBlog.php
 - ImgCat.php
 - ImgDog.php
 - ImgExotic.php
 - Role.php
 - Size.php
 - Sponsor.php
 - Status.php
 - TypeAnimal.php
 - TypeBlog.php
 - TypeExotic.php
 - User.php
 - ▶ Resources
 - ▶ Tests

- El controlador atiende las peticiones, llamadas y retorna el resultado desde la base de datos, al cliente, en este caso a Angular v4.

```
<?php
namespace AppBundle\Controller;

use Symfony\Bundle\FrameworkBundle\Controller\Controller;
use Symfony\Component\HttpFoundation\Request;
use Symfony\Component\Validator\Constraints as Assert;
use BackendBundle\Entity\Cat;

class CatController extends Controller {

    public function newAction(Request $request) {
        $helpers = $this->get("app.helpers");

        $hash = $request->get("authorization", null);
        $authCheck = $helpers->authCheck($hash);

        if ($authCheck == true) {
            $identity = $helpers->authCheck($hash, true);

            $json = $request->get("json", null);

            if ($json != null) {
                $params = json_decode($json);

                $createdAt = new \DateTime('now');
                $updatedAt = new \DateTime('now');

                $name = (isset($params->name) && preg_match('/^[a-z áéíóñúâëîôñ]+$/i', $params->name)) ? $params->name : null;
                $breed = (isset($params->breed) && preg_match('/^[a-z áéíóñúâëîôñ]+$/i', $params->breed)) ? $params->breed : null;
                $age = (isset($params->age) && ctype_digit($params->age)) ? $params->age : null;
                $description = (isset($params->description)) ? $params->description : null;
                $contact_person = (isset($params->contact_person) && preg_match('/^[a-z áéíóñúâëîôñ]+$/i', $params->contact_person)) ? $params->contact_person : null;
                $phone = (isset($params->phone) && ctype_digit($params->phone)) ? $params->phone : null;
                $email = (isset($params->email)) ? $params->email : null;
                $url_video = (isset($params->url_video)) ? $params->url_video : null;
                $id_gender = (isset($params->id_gender)) ? $params->id_gender : null;
                $id_size = (isset($params->id_size)) ? $params->id_size : null;
                $id_user = ($identity->sub != null) ? $identity->sub : null;
                $id_status = (isset($params->id_status)) ? $params->id_status : null;

                $emailContraint = new Assert\Email();
                $emailContraint->message = "This email is not valid !!";
                $validate_email = $this->get("validator")->validate($email, $emailContraint);
            }
        }
    }
}
```

- El apartado del modelo, es la lógica y la que interactúa con la Base de Datos.

```

    * @return Cat
    */
    public function setBreed($breed)
    {
        $this->breed = $breed;

        return $this;
    }

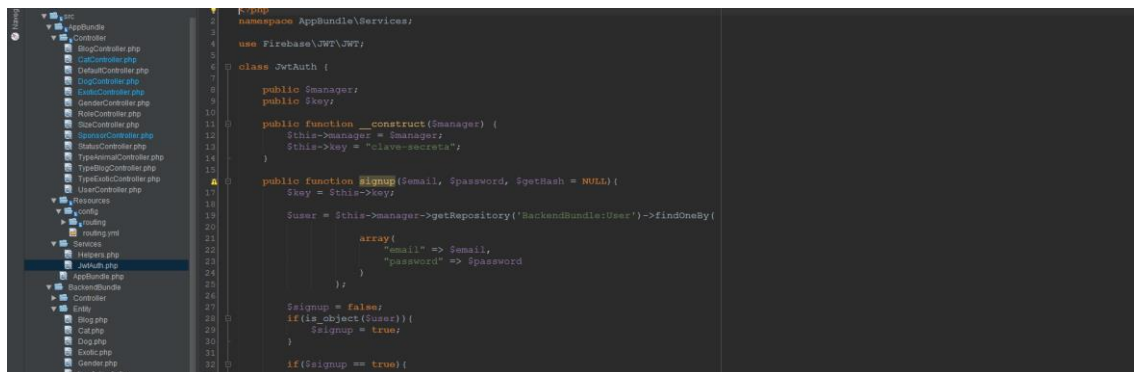
    /**
     * Get breed
     *
     * @return string
     */
    public function getBreed()
    {
        return $this->breed;
    }

    /**
     * Set name
     *
     * @param string $name
     *
     * @return Cat
     */
    public function setName($name)
    {
        $this->name = $name;

        return $this;
    }

    /**
     * Get name
     *
     * @return string
     */
    public function getName()
    {
        return $this->name;
    }
}
```

- Los servicios tratan la autorización por logueo, el Json, chequeo del Token.



2.2. Entorno cliente.

2.2.1.1. Estructura y funcionamiento

Este entorno está hecho con TypeScript y con el Framework Angular v4, en el cual se basa en la vista de la aplicación, es la interfaz de la aplicación, y a través de las peticiones por https, llama al entorno servidor de Symfony y se transmiten los datos en Json.

Antes de explicar la estructura, destacar que para poder utilizar este entorno es necesario instalar globalmente Node.js y Angular Cli.

La estructura de este apartado es:

- ✓ **Node_module:** esta carpeta contiene todas las librerías del package.json que se utilizarán después para la aplicación.
- ✓ **Src:** contiene el código fuente de la aplicación, dividida en carpetas, y cada carpeta contiene un componente, que contiene las funciones de la aplicación, el html, que contiene las vistas del componente, el scss, que contiene el css de este componente en concreto y un spec que es el test del componente.

El inicio de la aplicación empieza en index.html y a continuación, a partir de los selectores, en un principio app.component.html, y dependiendo de si utilizas, como es en mi caso, app.route, a partir

del routing, vas pasando de un componente a otro, formando lo que sería la aplicación.

La carpeta assets, contiene las imágenes, videos, js, etc. de la aplicación.

- ✓ Quedaría la pre-configuración de la aplicación, como puede ser angular-cli.json, karma.conf.js, package.json, protactor.conf.js, tsconfig.json, tslint.json.

- node_modules
- src
 - app
 - home
 - login
 - pipes
 - shared
 - voluntary
 - app.component.css
 - app.component.html
 - app.component.spec.ts
 - app.component.ts
 - app.module.ts
 - app.route.ts
 - assets
 - img
 - video
 - .gitkeep
 - environments
 - ★ favicon.ico
 - index.html
 - main.ts
 - polyfills.ts
 - styles.css
 - test.ts
 - {...} tsconfig.app.json
 - {...} tsconfig.spec.json
 - typings.d.ts
 - .angular-cli.json
 - .editorconfig
 - .gitignore
 - karma.conf.js
 - package.json
 - protractor.conf.js
 - README.md
 - {...} tsconfig.json
 - {...} tslint.json

- ✓ **Pipes:** la carpeta pipes que se encuentra dentro de /src/app/ son los que nos permite alterar la forma en la que se van a visualizar los datos, son una especie de filtros de datos.
- ✓ **Shared:** aquí se encuentra en la misma ruta que los pipes, y contienen los modelos, directivas y servicios de la aplicación.

3. Tecnologías utilizadas.

- **Entorno Servidor:**

- ✓ **Symfony 3:**

Para instalar Symfony, es necesario instalar un instalador de paquetes llamado Composer. Contando las dependencias de serie, que trae un proyecto en blanco, le añadí nuevas librerías en las dependencias que son:

- Firebase/php-jwt: para la autenticación de los tokens, la traducción de jwt es Json Web Token.
 - Knplabs/knp-paginator-bundle: que lo utilizo para la paginación de los datos traídos desde la Base de Datos.

- **Entorno Cliente:**

- ✓ **Node.js:**

Es un entorno de ejecución para JavaScript construido con el motor de JavaScript V8 de Chrome.

- ✓ **Angular v4:**

Como en symfony, necesita otros componentes para instalar una aplicación desde cero. En este caso, necesita node.js y angular cli.

Una vez instalados, le añado nuevas librerías a las dependencias del package.json y son:

- Bootstrap: para el css de la aplicación.
 - JQuery: para utilizar sus librerías.
 - ✓ **Inkscape:** para el logo de la aplicación transformado en svg y para el favicons, después transformado a la extensión .ico desde internet.
 - ✓ **CSS:** el css está dividido entre el predefinido de Bootstrap y propio, como es el ejemplo de las imágenes en scroll, mezclado el css de bootstrap para la animación, pero después utilizando el css propio de @media del media queries. Otros casos son como el menú de los módulos de los animales en la parte privada, las tablas, etc.
- **Redes:**

La aplicación, al estar dividida en dos partes, están alojados en hostings distintos, el del servidor está alojado en uno que corre con Apache y PhpMyAdmin y el entorno cliente está alojado en otro que utiliza node.js, cosa imprescindible para ejecutar la interfaz hecha por Angular v4.