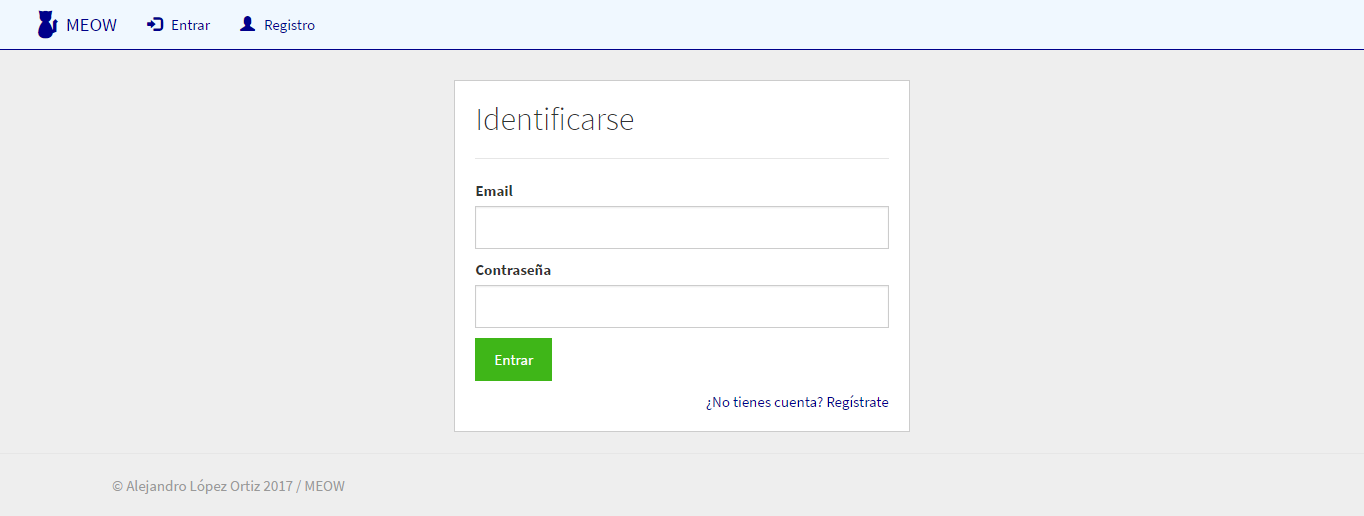


MEOW

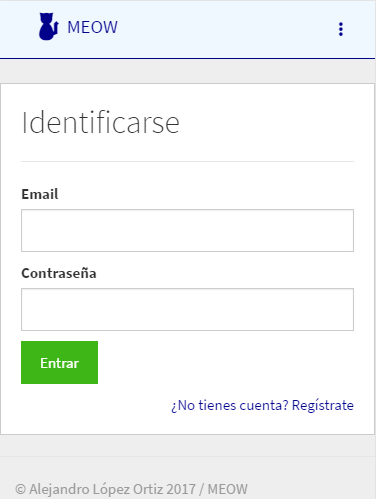
**Alejandro López Ortiz**

**1. MANUAL DE USUARIO**

Vamos a poder registrarnos o loguearnos en un principio.

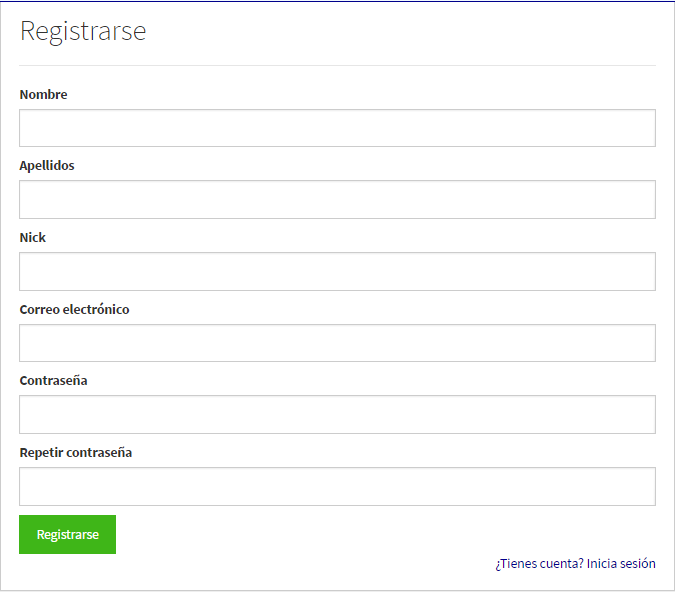


Versión móvil

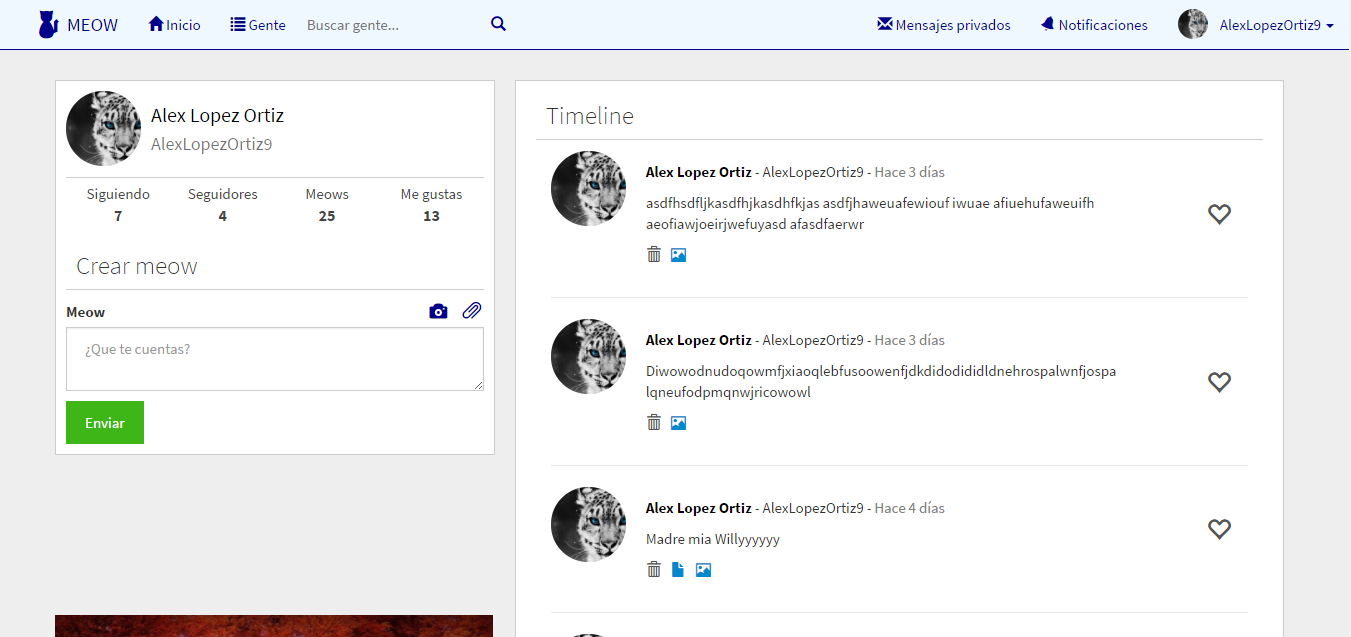


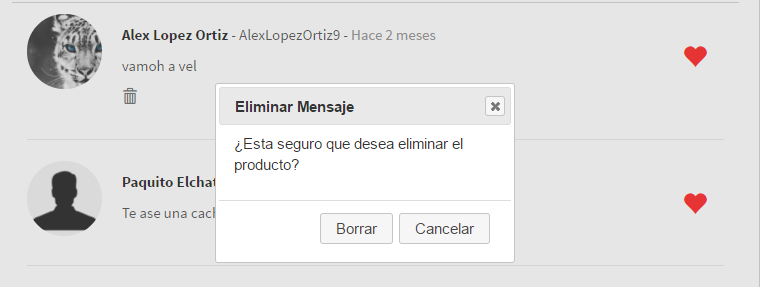
En el proceso de registro podemos observar que tenemos el formulario a rellenar con todos los datos necesario para crear el usuario, en el campo de nick realizamos una petición ajax cada vez que salimos del campo para verificar si ese nick introducido esta disponible o no.

Al momento de registrarnos podemos iniciar sesión con nuestro usuario.

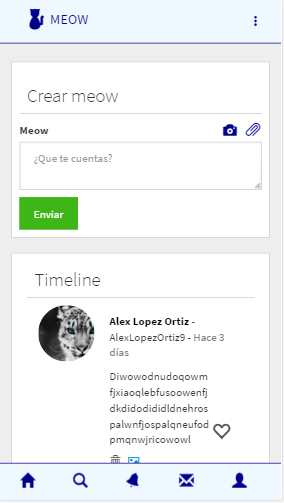
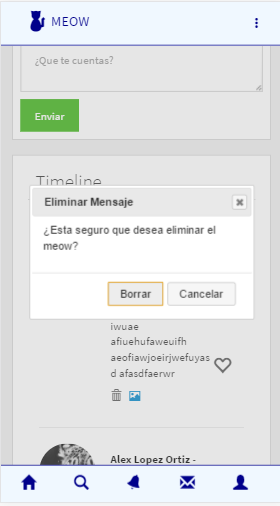


Cuando iniciamos sesión con nuestro usuario nos carga la página principal en la que encontramos un timeline con las publicaciones de la gente a la que sigo y las nuestras propias, las cuales puedo borrar si quisiera realizando una petición ajax y se muestra la función de jQueryUI en la que nos sale el popup preguntándonos si queremos borrar dicha publicación o no.

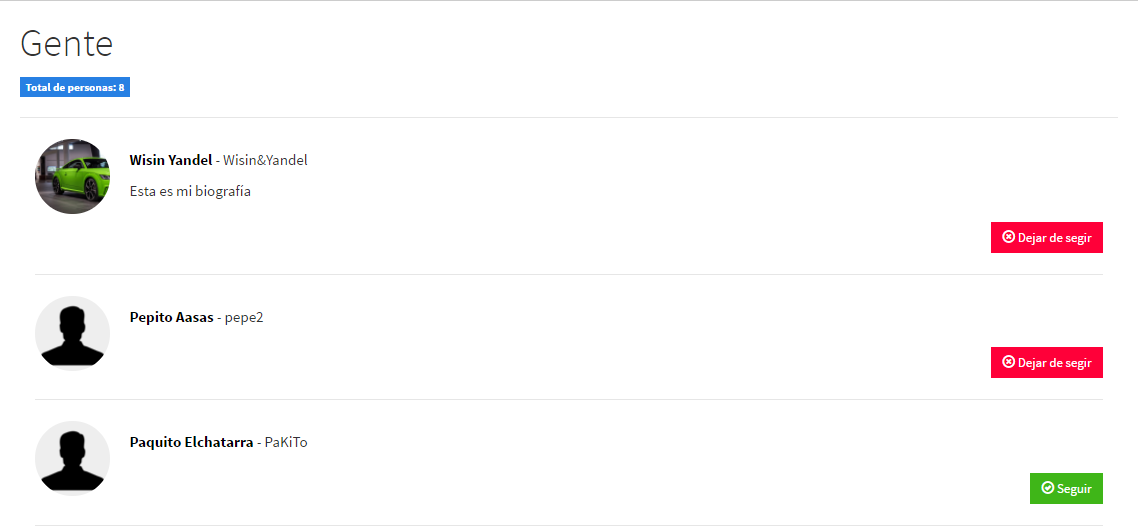


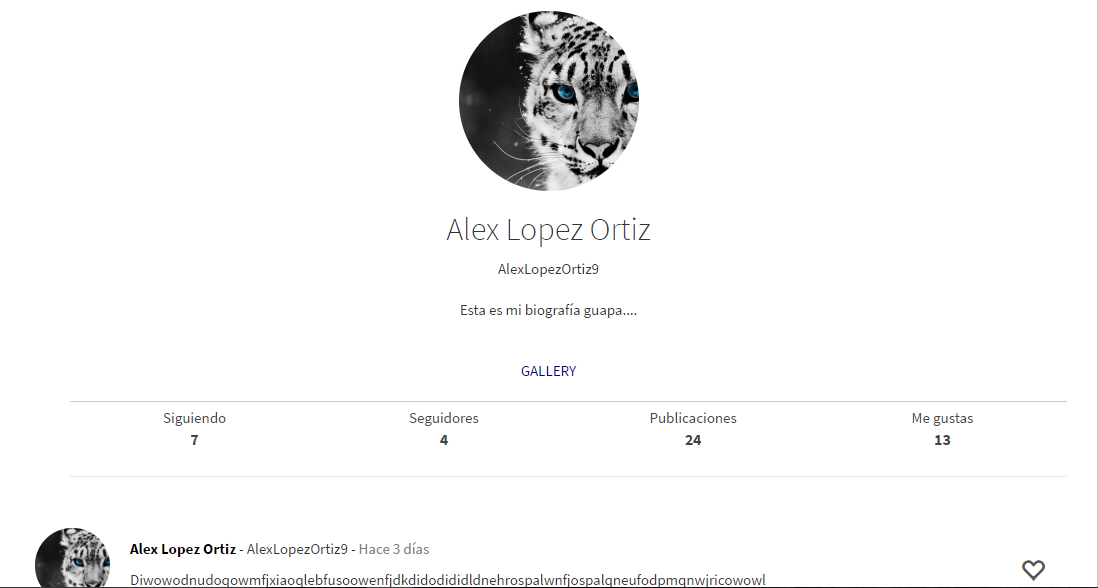


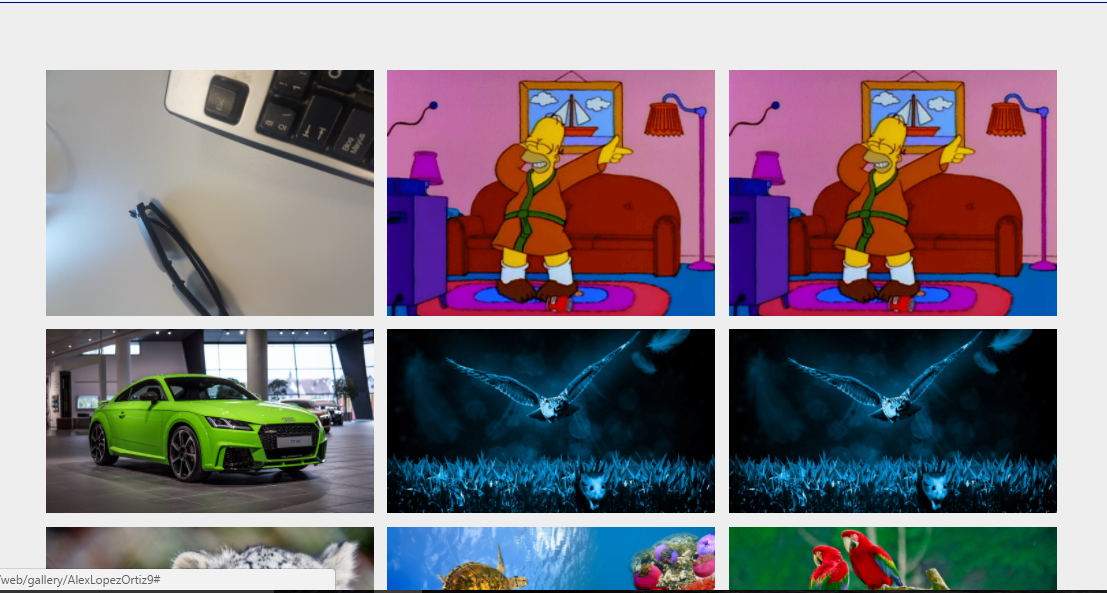
Versión móvil:

También podemos buscar gente, podemos seguir  y dejar de seguir a las personas que queramos, hay un buscador para buscar a personas registradas y ver su perfil sin necesidad de seguirlos aunque tenemos la opción de seguir o dejar de seguir, nos sale información de si nos sigue a nosotros o no y también podemos ver sus publicaciones y podemos marcarlas como me gusta o quitarlas de me gusta y también podemos ver la galería de imágenes que muestra todas las imágenes subidas del usuario pudiendo ver el meow y la fecha en que lo publicó, tenemos una tabla con las estadísticas del perfil que visitamos en la cual podemos ver el número de publicaciones, seguidores, personas que sigue y me gusta que tiene.

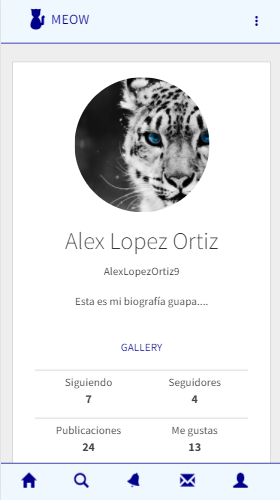
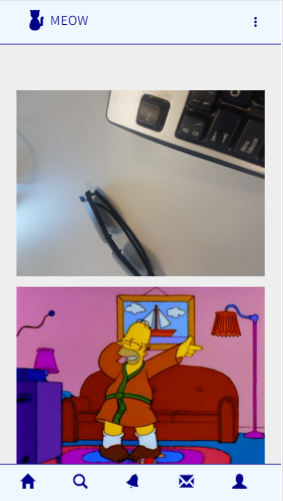




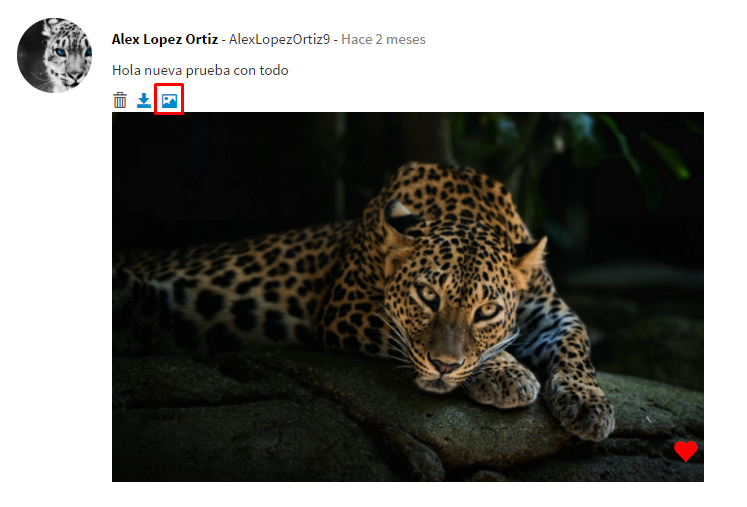


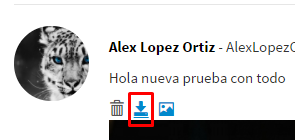


Versión móvil:

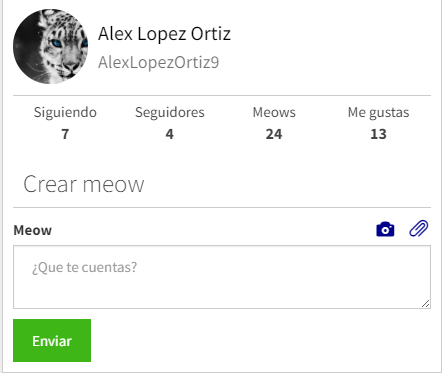
Desde el timeline de la pagina principal también podemos darle a me gusta a las publicaciones y automáticamente nos sale en nuestra lista de me gusta, podemos mostrar la imagen que se adjunta y también podemos visualizar y descargar el archivo que se adjunta en la publicación.

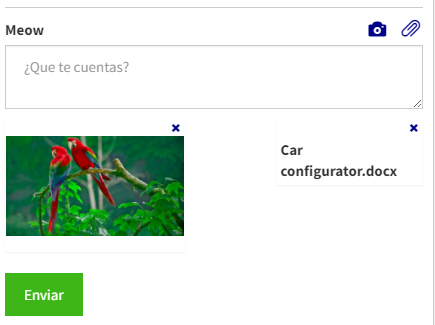


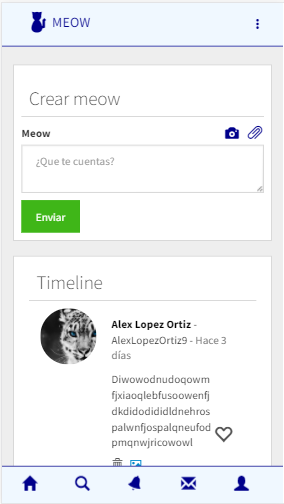
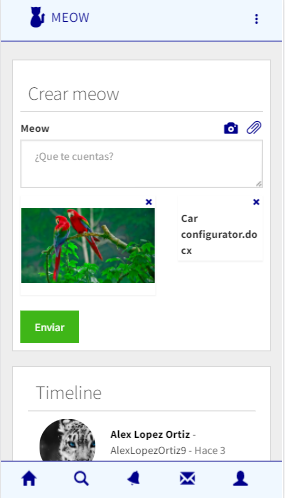
 Con este botón podremos visualizar y descargar el archivo.

Podemos crear nuestras publicaciones en las que podemos subir también fotos y archivos y automáticamente sale en el timeline de todos los usuarios que me siguen y también en el mío.

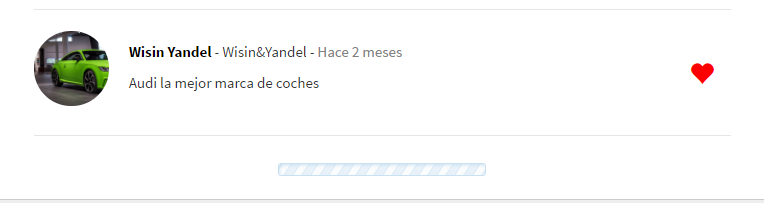
En la página principal tenemos nuestra tarjeta personal con acceso directo a nuestro perfil y se muestran todos nuestros datos (nombre, nick, biografía, foto perfil, publicaciones, estadísticas). También podemos acceder a la sección de mi perfil que sería un link igual al perfil de usuario.



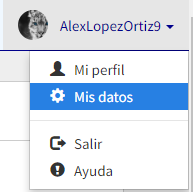


Cuando hacemos scroll se ve que se hace una paginación infinita, al cargar una serie de paginas de esta paginación infinita nos sale el botón de ver más publicaciones lo que nos permite que sea mas amigable la navegación con nuestro timeline.



En la sección de mis datos podemos editar nuestros datos e imagen de perfil.





Versión móvil:



Podemos cerrar sesión y entrar sin problemas.

Tenemos un sistema de notificaciones que se activan cuando nos dan me gusta en nuestras publicaciones o nos siguen, esto se realiza con peticiones ajax.

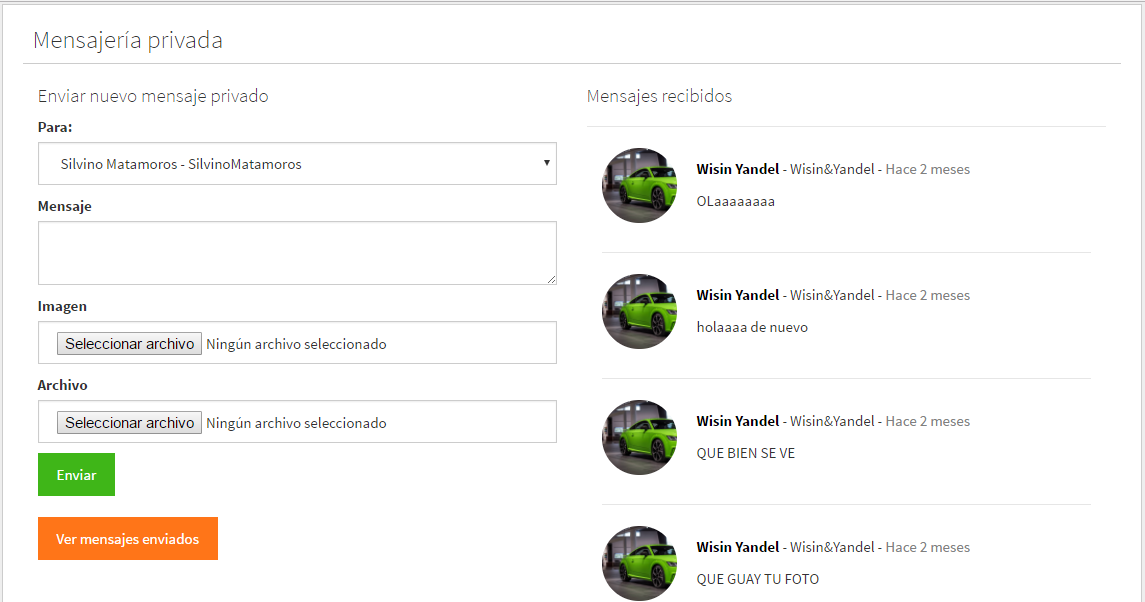


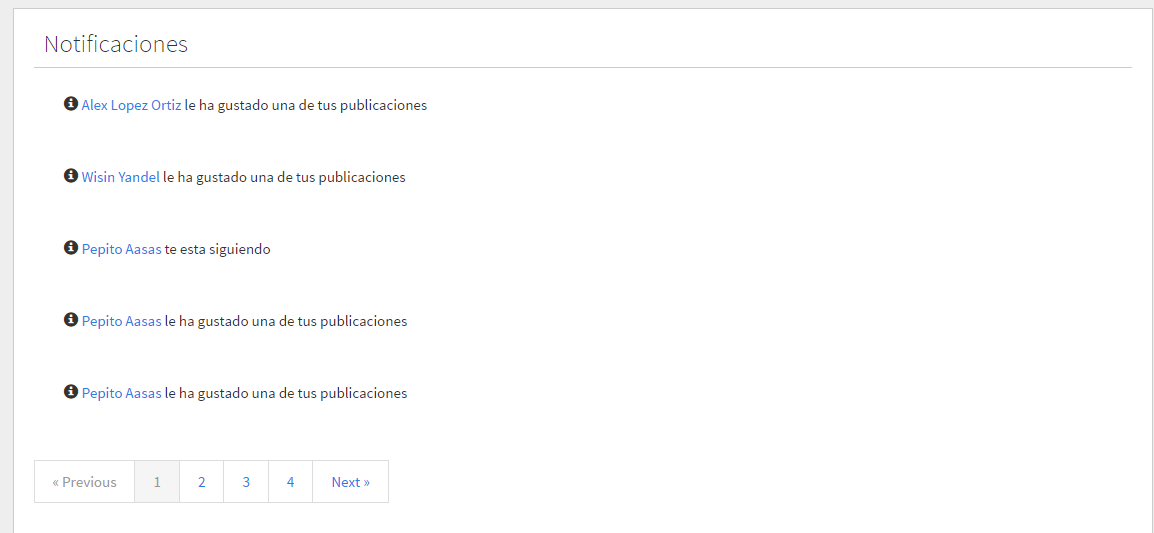
Versión móvil:



Hay una mensajería privada que podemos enviar mensajes privados y recibirlos de la gente.

Dentro de la mensajería tenemos una lista con los mensajes recibidos y otra con los mensajes enviados.





**2. MANUAL TÉCNICO**

**2.1. Modelo de Datos**

La base de datos consta de 6 tablas:

- Users

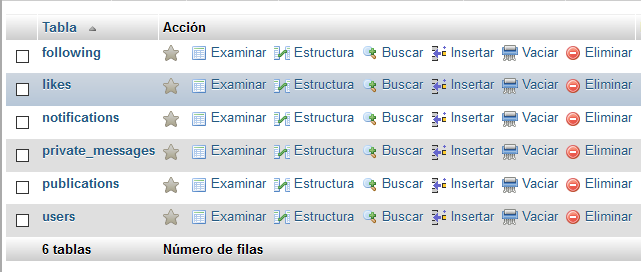
- Publications

- Likes

- Following

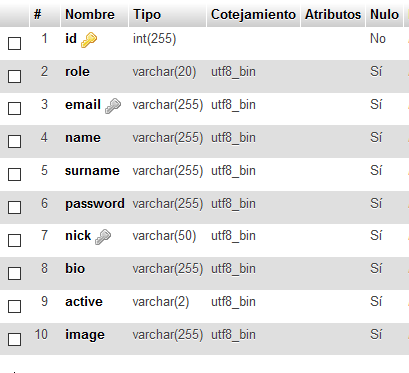
- Private\_messages

- Notifications



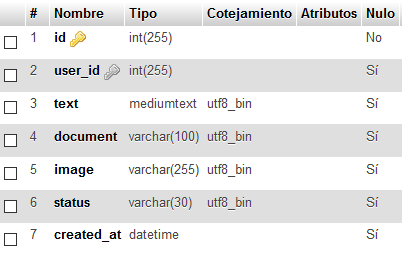
En la tabla Users nos encontramos con las siguientes columnas:

* Id: Identificación del usuario
* Role: Tipo de usuario
* Email: Email para el registro e inicio de sesión del usuario
* Name: Nombre elegido por el usuario
* Surname: Apellido del usuario
* Password: Contraseña elegida por el usuario
* Nick: Nick elegido por el usuario para su cuenta
* Bio: Biografía en la que el usuario pondrá los datos que quiera
* Active: Para ver si está activo o no
* Image: Imagen de perfil del usuario



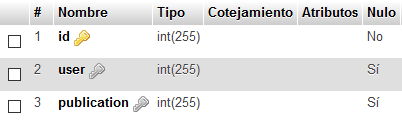
En la tabla Publications nos encontramos con las siguientes columnas:

* Id: Identificación de la publicación
* User\_id: Identificación del usuario que ha creado la publicación
* Text: Texto introducido en la publicación
* Document: Documento subido en la publicación
* Image: Imagen subida en la publicación
* Status:
* Created\_at: Para saber la hora de publicación



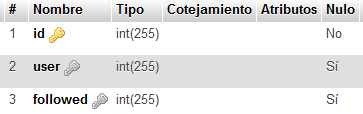
En la tabla Likes nos encontramos las siguientes columnas:

* Id: Identificador del like
* User: Id del usuario que le da like a la publicación
* Publication: Id de la publicación a la que le han dado like



En la tabla Following nos encontramos las siguientes columnas:

* Id: Identificador de following
* User: Identificador del usuario que sigue a alguien
* Followed: Id del usuario que han seguido



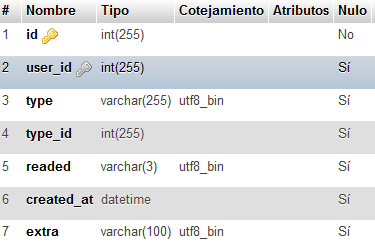
En la tabla Private\_messages nos encontramos con las siguientes columnas:

* Id: Identificador del mensaje
* Message: El texto que se envía en el mensaje
* Emitter: Id del usuario que envía el mensaje
* Receiver: Id del usuario que recibe el mensaje
* File: Archivo que enviamos en el mensaje
* Image: Imagen que enviamos en el mensaje
* Readed: Para marcar el mensaje como leído
* Created\_at: Para el momento del envío del mensaje



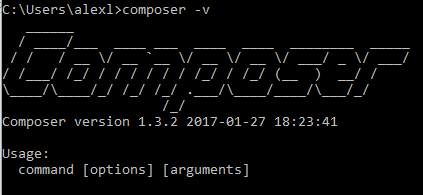
En la tabla Notifications nos encontramos con las siguientes columnas:

* Id: Id de la notificación
* User\_id: Id del usuario al que va dirigida la notificación
* Type: Para saber si es tipo follow o like
* Type\_id: Id del usuario que realiza la notificación
* Readed: Para marcar como leída la notificación
* Created\_at: Para saber el momento de la notificación
* Extra: Id de la publicación que gusta en caso de que sea notificación por publicación sino el valor por defecto es null



**Creación del Proyecto Symfony 3**

En primer lugar instalé el servidor de aplicaciones web Wamp64. Después instalé Composer que es un gestor de dependencias de php con el que podremos descargar paquetes y actualizarlos.



A continuación instalé el emulador de la consola de Linux Cygwin.



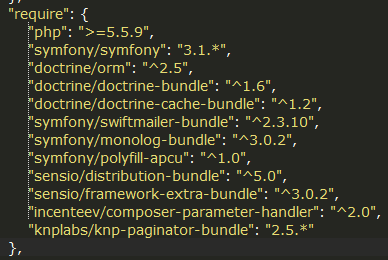
El IDE que utilicé fue el Netbeans.

Después de haber preparado el entorno de desarrollo, comencé con la instalación de Symfony3.

Utilizando la consola Cywing para crear el proyecto dentro de wamp64.



Utilizo un bundle de Symfony para tener las paginaciones en los listados de cosas que va a tener la web. En el archivo de composer.json ponemos:



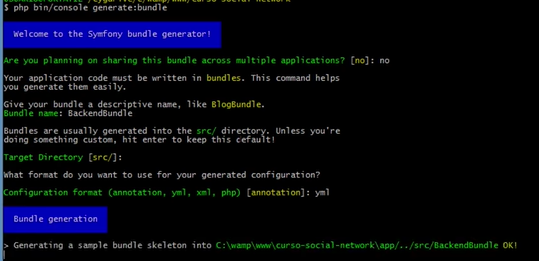
Y en la consola se pone dentro del directorio de nuestro proyecto:



Así instalo los nuevos paquetes, las actualizaciones y el nuevo paquete de la paginación.

La aplicación está dividida en dos Bundles. Uno será para guardar las entidades y modelos y trabajos con la base de datos y el otro para el resto de la aplicación.

El que se encarga del resto de la aplicación esta por defecto creado en el proyecto Symfony, pero el otro Bundle lo vamos a crear nosotros con la consola dentro de la carpeta de nuestro proyecto:



Ahora al crear el BackendBundle voy a pasar nuestras tablas de la base de datos a código doctrine.

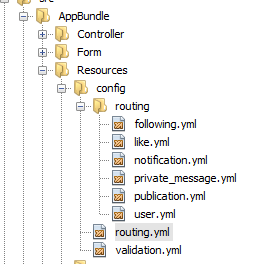
Importo la base de datos a yml:



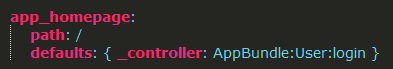
Ahora genero las entidades para poder utilizarlas en el código:



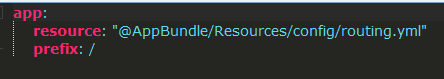
Creo los ficheros de rutas que voy a necesitar para cargar las diferentes rutas de los controladores.



Creo una ruta por defecto:

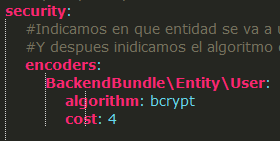


Con esto cargo directamente el fichero de rutas.



Empiezo con la maquetación de la página principal utilizando el motor de plantillas TWIG.

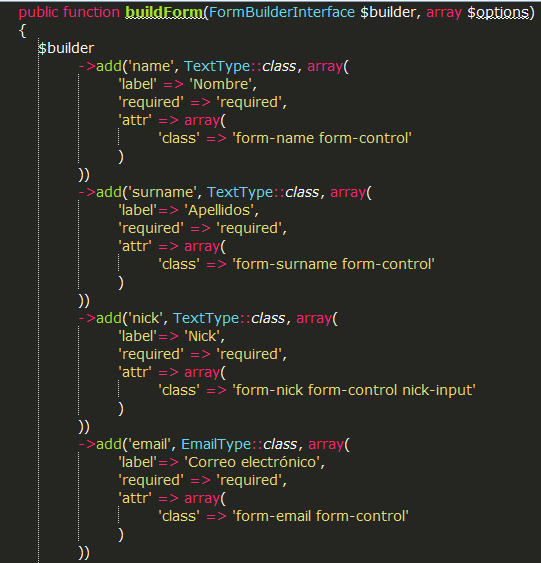
Antes de empezar con el login y el registro, indico en que entidad se va a utilizar este protocolo de cifrado y después indico el algoritmo de cifrado que es bcrypt y después, cuántas veces se vuelve a cifrar la contraseña ya cifrada

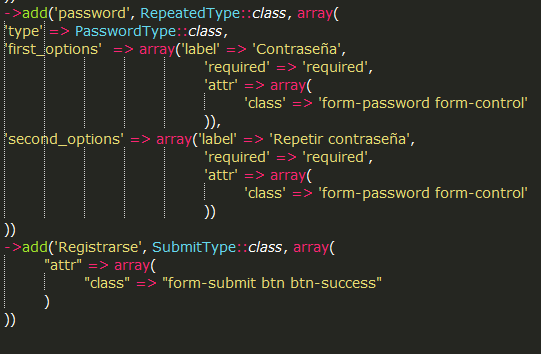


Esto nos va generar una clase con un formulario basada en la entidad especificada.

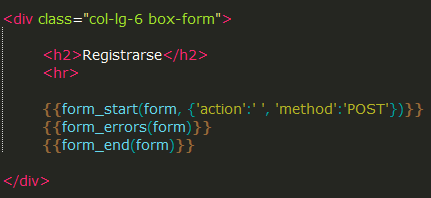


Cuando se crea el formulario realizo modificaciones para hacerlo útil:



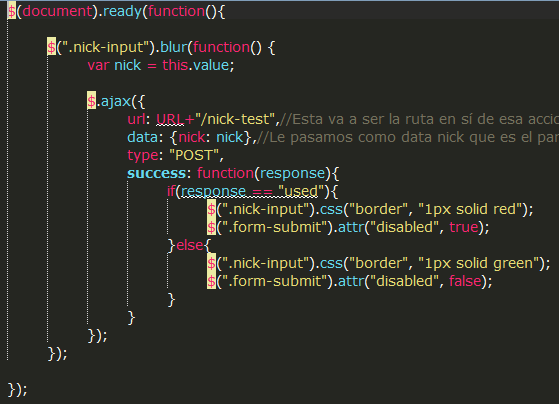


Lo muestro en la vista de ésta manera:

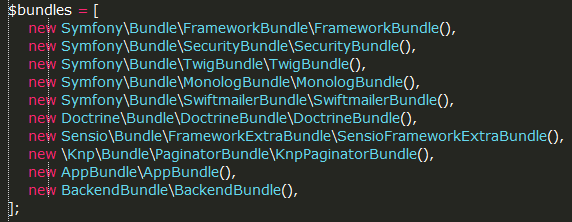


Utilizo las sesiones para hacer la función de los mensajes flush que es una pequeña alerta de bootstrap y que cuando volvemos a actualizar la página ese mensaje flush ya no esté. Así que en UserController creo la sesión.

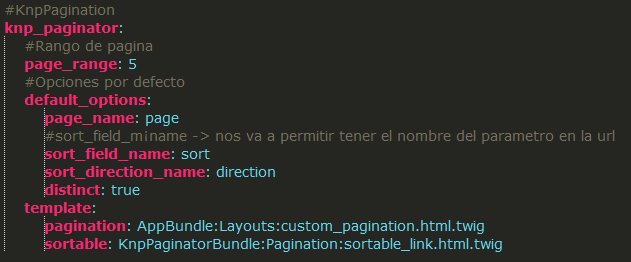
Ahora en el registro realizo la comprobación por Ajax del nickname.



Para realizar el sistema de paginación, instancio el knp\_paginator en AppKernel que es donde están todos los bundles que tiene la app.

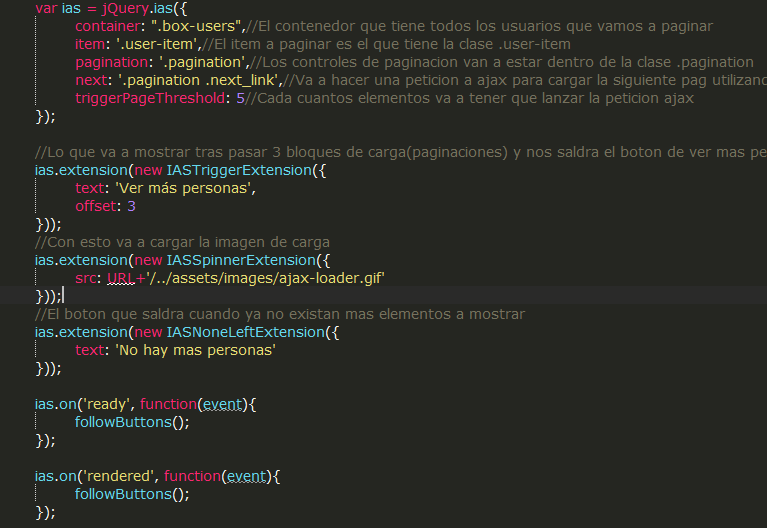


Configuro la paginación en el config.yml

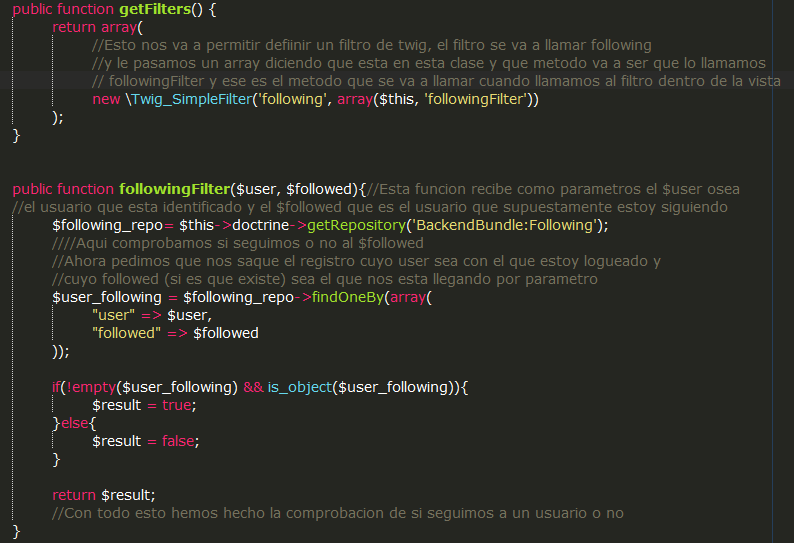


Después de la sección gente hago la parte del buscador en la que creo el formulario con Symfony y también creo el Action del buscador.

Para terminar con la sección de gente realizo el js de users en el que utilizamos el plugin de Ajax (ias).



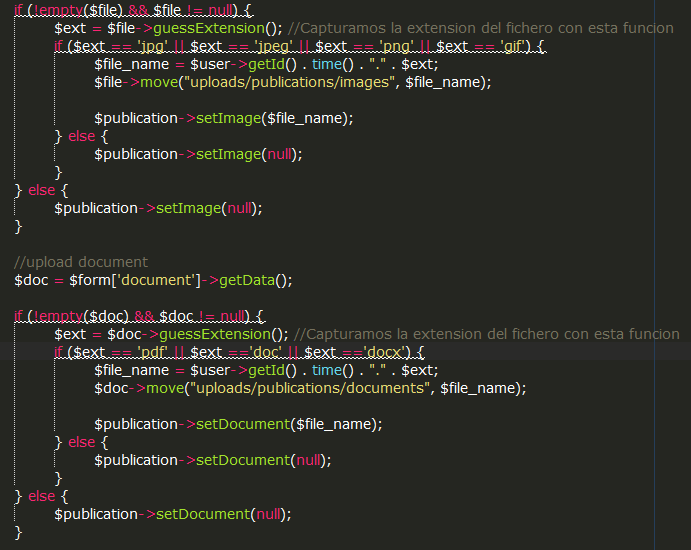
El sistema de Following en el cual podemos seguir y dejar de seguir a otros usuarios, realizando su Action como en las demás ocasiones y creamos una extensión de Twig.



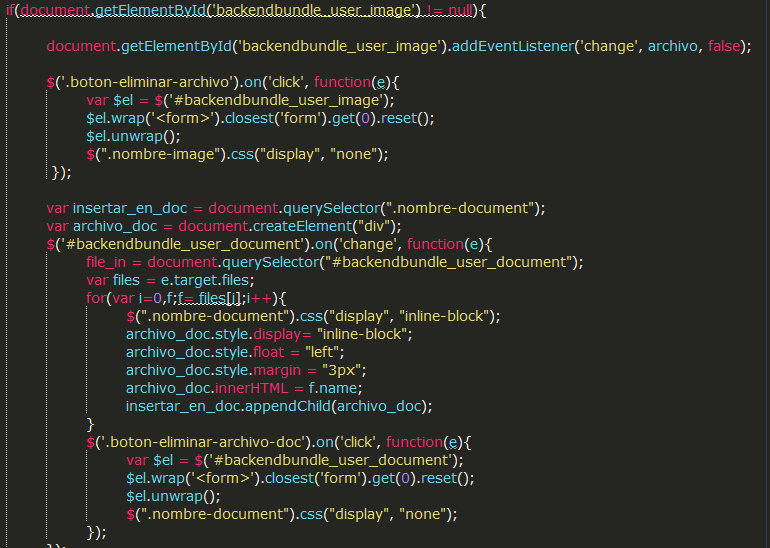
Luego su funcionalidad en Ajax y el cambio de botones con jquery.



En las publicaciones podemos subir archivos ya sean imágenes o documentos y tienen restricciones de que tipo se acepta y cual no.



Cuando seleccionamos la imagen a subir o el documento muestro una previsualización de los elegidos y lo realizo mediante jquery con una función que realiza la lectura del documento y luego la muestro.

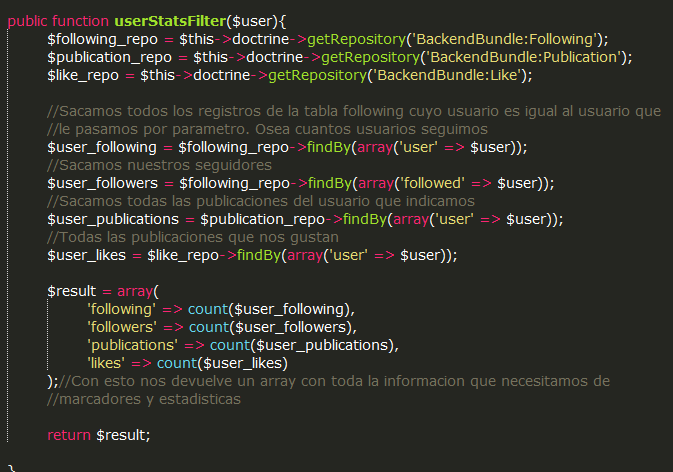


En el timeline cuando la publicación que hay tiene una imagen realizamos una función con JQuery en la que habilitamos dicha foto. Y si la publicación es nuestra saldrá el icono de la basura con el cual borramos dicha modificación con Ajax abriendo primero una ventana de JQuery UI.

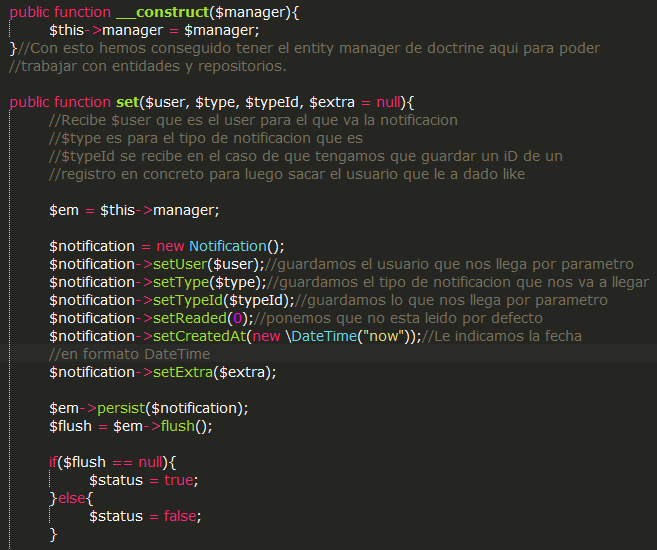


El sistema de me gustas también lo hago con Ajax y realizo el cambio de botón.



En el home presentamos unas estadísticas con nuestros datos de seguimientos publicaciones y me gustas. Para esto se necesita otra extensión de twig que se llama userStatsFilter.

Para la sección de notificaciones realizo un servicio que puedo utilizar en todos los controladores de la app.



Hago lo mismo con el sistema de notificaciones por Ajax que realizo la comprobación de que si dichas notificaciones están leídas o no y muestro el label con las notificaciones.

