



# SHORCUT

**Manual de uso**

---

20/1/2015

- 1. ENTORNO Y ALCANCE DE LA WEB**
- 2. NORMAS Y PROCEDIMIENTOS PARA SU IMPLEMENTACIÓN**
- 3. DESCRIPCIÓN DE MODELOS**
- 4. DESCRIPCIÓN DE VISTAS**
  - 4.1. GESTIÓN DE USUARIO
  - 4.2. GESTIÓN DE SHORTCUTS
  - 4.3. GESTIÓN DE PERFIL
  - 4.4. GESTIÓN DE FOLLOWS
  - 4.5. BÚSQUEDA

## 1. ENTORNO Y ALCANCE DE LA WEB

ShortCut es una versión bastante prematura de lo que podría llegar a ser, ya que le faltan ciertas funcionalidades que sí tienen otras páginas de la misma índole, aunque en futuras versiones podrían implementarse. Con las funcionalidades que posee actualmente podría identificarse como una red social de microblogging propiamente dicha.

El diseño de la web se ha realizado mediante HTML5 y CSS.

## 2. NORMAS Y PROCEDIMIENTOS PARA SU IMPLEMENTACIÓN

La web ha sido realizada bajo el framework Django en su versión 1.7. Se puede utilizar en cualquier navegador disponible.

Para su puesta en marcha en un *localhost* con fin de hacer una prueba, hay que ejecutar el comando *python manage.py runserver* en el directorio del proyecto.

El proyecto posee ya una base de datos con varios usuarios que poseen sus propios shortcuts y sus propios seguidores y usuarios a los que siguen. Se ha dejado un usuario creado con nombre y contraseña **practica**. Este usuario no posee ningún shortcut, como tampoco ningún seguidor ni sigue a nadie, ya que se ha dejado listo para la prueba de todas las funcionalidades del sistema. Es superusuario. Se deja otro usuario (nombre y contraseña) **practica2** sin estos permisos.

## 3. DESCRIPCIÓN DE MODELOS

Se ha hecho uso de dos únicos modelos en este proyecto:

- **Shortcut:** Esta clase tiene tres atributos:
  - content: Un campo de cadena con un máximo de 140 caracteres.
  - creation\_date: Atributo que contiene la fecha en la que se creó el shortcut. Se almacena automáticamente.
  - user: Clave foránea a la clase User de Django.
- **UserProfile:** Esta clase tiene cuatro atributos:
  - user: Clave foránea a la clase User de Django.
  - follows: Un campo muchos-a-muchos con la simetría de la relación desactivada, ya que si un usuario sigue a otro, este último no tiene porqué seguir al primero.
  - avatar: Imagen que tendrá el usuario en su perfil.
  - bio: Cadena de caracteres con un máximo de 160 donde el usuario pondrá una breve descripción de sí mismo.

Posteriormente se utiliza una propiedad de Django para que al usar *User.profile*, se pueda acceder también a los campos de la clase *UserProfile*.

## 4. DESCRIPCIÓN DE VISTAS

### 4.1. GESTIÓN DE USUARIO

- **home:** Vista que controla qué mostrar al cliente desde la página principal ('/'). Si el usuario no está logeado, mostrará la plantilla *home.html*, que le mostrará al cliente las opciones de inicio y registro de sesión. Si lo está, le mostrará la plantilla *base.html*, en la que se encuentran el menú de navegación, información sobre el usuario y los ShortCuts de los usuarios a los que sigue el usuario con sesión iniciada.
- **newUser:** Vista que permite el registro de un nuevo usuario en la base de datos.
- **login\_view:** Permite el control del inicio de sesión de un usuario en el sistema.
- **logout\_view:** Si el usuario tiene sesión iniciada, se usará ésta vista para que cierre sesión en el sistema.
- **users:** Vista que muestra los perfiles de los usuarios que están registrados en el sistema. Si esta vista recibe el un nombre de usuario, devolverá una plantilla que muestra el perfil de dicho usuario, dándole la oportunidad de seguir si aún no lo sigue. En la plantilla, si el perfil es del usuario con sesión iniciada, le dará la opción de modificar los datos de su perfil.

### 4.2. GESTIÓN DE SHORTCUTS

- **submit:** Controla que el formulario que genera un nuevo ShortCut es correcto y lo relaciona con el usuario que está logeado (es el que lo ha creado).

### 4.3. GESTIÓN DE PERFIL

- **settings:** Es una vista que muestra un formulario para modificar los diferentes campos de un usuario, como pueden ser u e-mail, su nombre, su nombre de usuario, su bio, su avatar o su contraseña.
- **interactions:** Muestra los ShortCuts que contienen el nombre de usuario del usuario que tiene la sesión iniciada.

### 4.4. GESTIÓN DE FOLLOWS

- **follow:** Controla que el usuario que ha hecho click en "Follow" en la página, no sigue al usuario al que quiere seguir, y si no lo sigue, actualiza dicho campo en su perfil y lo devuelve a la página donde se muestran todos los perfiles.

### 4.5. BÚSQUEDA

- **search\_view:** Se utiliza para realizar una búsqueda de unas palabras en los ShortCuts y usuarios existentes. Si se realiza una búsqueda de "practica" se mostrará el usuario **practica** y todos los ShortCuts que contengan dicha palabra.