

## Django Snippets

Aplicación en la cual usuarios registrados pueden crear [Snippets de código](#) en diferentes lenguajes de programación. Estos pueden ser públicos (cualquier usuario o persona no registrada lo puede ver en el listado de snippets de la app) o privados (solo los puede ver el usuario que lo creó en su listado de snippets).

Ejemplo de app donde se cargan snippets: <https://www.codepile.net/pile/WE8kLbR4>

Se provee un [repositorio](#) con un proyecto Django ya configurado, con una app y templates necesarios. El proyecto además cuenta con vistas y urls hechas para que se pueda navegar por el sitio y comprender cómo debe funcionar. Estas vistas y URLs deben ser reemplazadas por vistas y urls reales en cada caso.

Se debe descargar el repositorio y subirlo a uno propio en [GitHub](#) o realizar un fork del mismo.

## Tecnologías a utilizar (definidas en el requirements.txt):

### Backend

- Python 3
- Django 2.2.4
- SQLite
- Pygments

### Frontend

- Bootstrap 4 <https://getbootstrap.com>

### Funcionalidades

- Login y Logout (la creación de usuarios se puede hacer desde el admin de Django).
- [CRUD](#) de Lenguajes de programación (utilizar el admin de Django, no tiene interfaz en la web).
- [CRUD](#) de [Snippets de código](#).
- En la pantalla de detalle del snippet, mostrar el texto del mismo formateado en el lenguaje correspondiente. Para ello se puede utilizar la librería [Pygments](#) para darle formato desde backend.

## Modelos

### Language

- name: CharField, 50 caracteres máximo, obligatorio.
- slug: CharField, 50 caracteres máximo, obligatorio. Se utilizará para identificar el lenguaje en la url de snippets del mismo.

### Snippet

- user: ForeignKey al usuario, obligatorio.
- created: DateTimeField (se autocompleta cuando se crea).
- updated: DateTimeField (se actualiza cuando se modifica).
- name: CharField, máximo de caracteres 255, obligatorio.
- description: TextField, opcional.
- snippet: TextField, obligatorio.
- language: ForeignKey al lenguaje, obligatorio.
- public: BooleanField, default False.

## Vistas

- Login.
- Pantalla de inicio: Lista todos los snippets públicos con un link para ver el detalle
- Detalle de snippet.
- Carga de snippets.
- Pantalla del usuario: Lista todos los snippets públicos del usuario, si es el mismo usuario logueado también muestra los snippets privados.
- Pantalla de edición del snippet: Se puede acceder desde la pantalla del listado desde un botón (solo aparece en los snippets del usuario) o desde la pantalla del usuario (sólo cuando el usuario es el mismo que el logueado)
- Eliminación del snippet: Botón que aparece en el listado de los snippets y en la pantalla de detalle del usuario (cuando el snippet es del usuario logueado).
- Listado de snippet por lenguajes (solo muestra los snippets públicos), desde cualquier listado de snippets o detalle de los mismos debe tener un enlace en el lenguaje que lleve a una vista que liste todos los snippets del lenguaje.
- **Opcional:** al crear un snippet nuevo (solo en la creación) enviar un email a una cuenta de correo electrónico personal con subject=name del snippet y body=description del snippet. **Este envío debe realizarse a través de Celery.**

## Notas

- Para los formularios de las vistas utilizar crispy-forms.
- Utilizar vistas basadas en clases.
- Realizar el deploy de la aplicación en Heroku.