

MANUAL DE INSTALACIÓN Y CONFIGURACIÓN DE AUDITOR TURPIAL

Esta librería denominada `turpial_auditor` es una extensión del *framework* Django que proporciona y monitorea las trazas de las instancias de los modelos auditables.

REQUISITOS

- Python 2.7
- Django 1.10
- jsonfield 2.2

INSTALACIÓN

Agregue `"turpial_auditor.auditor_core"` a su configuración `INSTALLED_APPS` de la siguiente manera:

```
INSTALLED_APPS = (  
    ...  
    'turpial_auditor.auditor_core',  
)
```

Incluya el URLconf del auditor en su proyecto `urls.py` de la siguiente manera:

```
url(r'^auditor/', include('turpial_auditor.urls')),
```

Ejecutar `"python manage.py makemigrations"` para crear el archivo de migración para el modelo de auditor.

Ejecutar `"python manage.py migrate"` para aplicar la migración y crear el modelo de Auditor.

USO Y EJEMPLOS

EL MODELO

Esta biblioteca proporciona un modelo que registra la acción (CREATED, UPDATED, DELETED), el valor anterior de la instancia, el nuevo valor, la fecha, la hora y el autor (la persona que ejecuta la acción), que se llama **AuditableAction**.

Nota: Los valores, antiguos y nuevos, de la instancia se representan como un JSON que excluye la representación de muchos campos.

HACER AUDITABLE UN MODELO EN DJANGO

Importar **AuditableMixin** de turpial_auditor. Una herencia en tu modelo, asegúrate de que sea el primero, el orden importa:

```
from turpial_auditor.auditor_core.models import AuditableMixin

class CarsModel(AuditableMixin, ..., models.Model, ...):
    ...
```

Si desea auditar un modelo muchos a muchos (*many to many*), deberá definir el suyo utilizando la propiedad **through**:

```
from turpial_auditor.auditor_core.models import AuditableMixin

class Person(models.Model):
    ...

class Group(models.Model):
    ...
    members = models.ManyToManyField(Person, through='Membership')
    ...

class Membership(AuditableMixin, models.Model):
    person = models.ForeignKey(Person, on_delete=models.CASCADE)
    group = models.ForeignKey(Group, on_delete=models.CASCADE)
    ...
```

Ahora su modelo puede ser auditado, sin embargo, no guarda el usuario registrado en el Modelo **AuditableAction**. Puedes lograr esto agregando a tus vistas otra *mixin* llamado **AuditableMixinView**:

```
from turpial_auditor.auditor_core.views import AuditableMixinView

class CarsView(AuditableMixinView, ...):
    ...
```