Hi. I’m Steve.

@schen_nyc

Let's Talk About Django

the D is silent

Rails vs Django

No 1. There’s no Django Magic.

Rails - Routes

```
Rails.application.routes.draw do
  resources :users
  resources :posts do
    resources :comments
  end
end
```

Django - Routes

```
from django.conf.urls import include, url
from . import views

urlpatterns = [
    url(r'^$', views.post_list, name='post_list'),
    url(r'^post/(?P<pk>[0-9]+)/$', views.post_detail, name='post_detail'),
]
```

Rails - Controller

```ruby
class PostsController < ApplicationController
  def index
    @posts = Post.all
  end

  def new
    @post = Post.new
  end

  def create
    @post = Post.create(post_params)
    current_user.add_post(@post)
    redirect_to posts_path
  end

  def show
    @post = Post.find(params[:id])
    @comment = Comment.new
    @comment.post_id = @post.id
  end
```

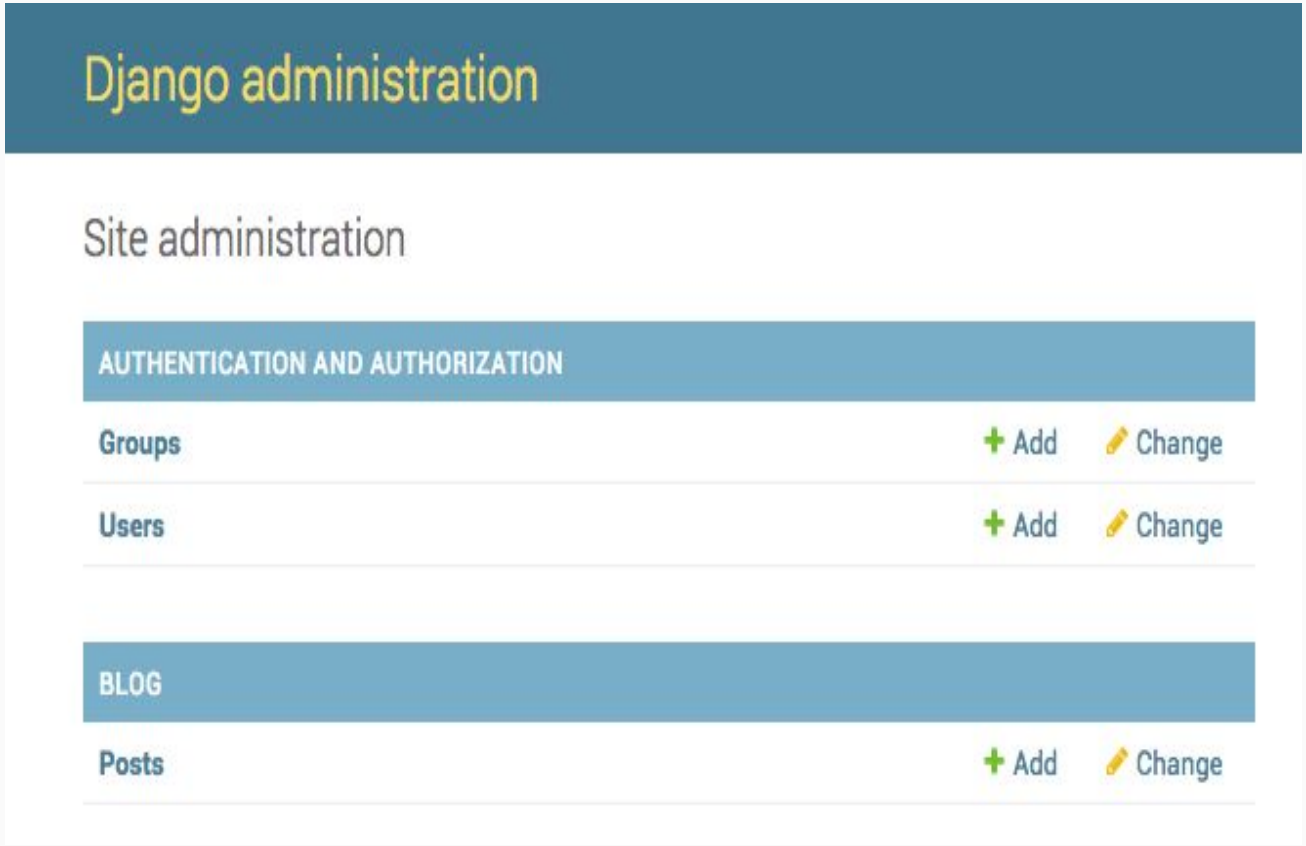
Django - Views

```python
from django.shortcuts import render, get_object_or_404, redirect
from .models import Name
from rest_framework import viewsets
from website.serializers import NameSerializer

# Create your views here.
def home(request):
    names = Name.objects.all()
    return render(request, 'website/home.html', {'names': names})

def company_detail(request,pk):
    name = get_object_or_404(Name,pk=pk)
    return render(request, 'website/company_detail.html', {'name': name})

def delete(request,pk):
    delete_name = get_object_or_404(Name,pk=pk).delete();
    return redirect('/')
```

Rails vs Django

No 2. Except when it comes to creating Users

Django - Admin interface



The Assignment