

# Guía para crear usuarios dinámicamente en Django

## Paso 1: Configurar la app

Crea un nuevo proyecto de Django o utiliza uno existente.

Crea una nueva aplicación (app) dentro del proyecto si aún no tienes una.

## Paso 2: Definir el modelo del usuario

En el archivo `models.py` de tu app, define el modelo del usuario si no lo has hecho ya. Puedes extender el modelo `User` predeterminado de Django o crear uno personalizado. Por ejemplo:

```
from django.db import models
from django.contrib.auth.models import AbstractUser

class CustomUser(AbstractUser):
    # Agrega campos adicionales si es necesario
    age = models.PositiveIntegerField(blank=True, null=True)
    # Otros campos personalizados
```

## Paso 3: Migraciones

Realiza las migraciones necesarias para crear la tabla de usuarios personalizada en la base de datos. Ejecuta los siguientes comandos en la terminal:

```
python manage.py makemigrations
python manage.py migrate
```

## Paso 4: Crear el formulario de registro

Crea un formulario de registro personalizado en `forms.py` de tu app. Este formulario debe incluir los campos necesarios para el nuevo usuario (por ejemplo, nombre de usuario, correo electrónico, contraseña, etc.).

Puedes usar el formulario de creación de usuario incorporado de Django o personalizarlo según tus necesidades.

```
from django import forms
from django.contrib.auth.forms import UserCreationForm
from .models import CustomUser

class CustomUserCreationForm(UserCreationForm):
    class Meta:
        model = CustomUser
        fields = ['username', 'email', 'password1', 'password2', 'age']
    # Agrega los campos adicionales aquí
```

### Paso 5: Vista para el registro de usuarios

En views.py de tu app, crea una vista para procesar el registro de usuarios. Utiliza el formulario personalizado que creaste en el paso anterior para validar y crear el nuevo usuario.

```
from django.shortcuts import render, redirect
from .forms import CustomUserCreationForm

def register_user(request):
    if request.method == 'POST':
        form = CustomUserCreationForm(request.POST)
        if form.is_valid():
            form.save()
            return redirect('registro_exitoso') # Redirige a una página
de registro exitoso o donde desees
        else:
            form = CustomUserCreationForm()

    return render(request, 'registro.html', {'form': form})
```

### Paso 6: Plantilla de registro

Crea una plantilla de registro (registro.html) en el directorio de plantillas de tu app. Utiliza el formulario creado en el paso anterior para mostrar los campos necesarios para el registro.

```
<!DOCTYPE html>
<html>
<head>
    <title>Registro de usuario</title>
</head>
<body>
    <h1>Registro de usuario</h1>
    <form method="post">
        {% csrf_token %}
        {{ form.as_p }}
        <button type="submit">Registrarse</button>
    </form>
</body>
</html>
```

### Paso 7: Configurar URL

En urls.py de tu app, configura la URL para la vista de registro de usuarios.

```
from django.urls import path
from . import views

urlpatterns = [
    path('registro/', views.register_user, name='registro'),
    # Otras URL de la app
]
```

### Paso 8: Ejecutar la aplicación

Ejecuta el servidor de desarrollo para probar tu aplicación.

```
python manage.py runserver
```

Ahora deberías tener una aplicación Django donde los usuarios pueden registrarse de forma dinámica sin utilizar la página de administración. Recuerda que esta es solo una guía básica, y dependiendo de tus necesidades, puedes personalizar y expandir la funcionalidad del registro de usuarios en tu aplicación.