



# Manual del Sistema

## *Mis Gastos*

Pontificia Universidad Católica del Ecuador

Carrera: Tec Desarrollo De Software

Asignatura: Programación II

Profesor: Kevin Viteri

Elaborado por:

**Ariel Rosero y Ahinoa Andino**

Quito, 22 de junio de 2025

# Índice

<b>1. Introducción</b>	<b>2</b>
<b>2. Requerimientos previos</b>	<b>2</b>
<b>3. Instalación y configuración</b>	<b>2</b>
3.1. 1. Clonar el repositorio o descargar el código . . . . .	2
3.2. 2. Crear y activar entorno virtual . . . . .	2
3.3. 3. Instalar Django . . . . .	2
3.4. 4. Estructura de carpetas del proyecto . . . . .	3
<b>4. Estructura de archivos y código fuente</b>	<b>3</b>
4.1. 1. models.py . . . . .	3
4.2. 2. views.py . . . . .	3
4.3. 3. urls.py (de la app gastos) . . . . .	3
4.4. 4. urls.py (del proyecto principal) . . . . .	4
4.5. 5. settings.py (fragmento para archivos estáticos) . . . . .	4
<b>5. Archivos de plantilla (HTML)</b>	<b>4</b>
5.1. landing.html . . . . .	4
5.2. formulario.html . . . . .	4
5.3. listado.html . . . . .	5
<b>6. CSS (style.css)</b>	<b>5</b>
<b>7. JavaScript (scripts.js)</b>	<b>5</b>
<b>8. Ejecución del proyecto</b>	<b>6</b>
8.1. 1. Migrar la base de datos . . . . .	6
8.2. 2. Ejecutar el servidor . . . . .	6
<b>9. Evidencias de funcionamiento</b>	<b>6</b>
<b>10. Conclusión</b>	<b>7</b>

# 1 Introducción

El presente manual tiene como propósito guiar al usuario en la instalación, configuración, uso y comprobación del sistema **Mis Gastos**. Su objetivo principal es que cualquier persona que siga estos pasos logre implementar el sistema correctamente y pueda verificar su funcionamiento mediante evidencias visuales.

## 2 Requerimientos previos

Antes de comenzar, asegúrate de tener instalado en tu sistema lo siguiente:

- **Python 3.9 o superior**
- **Pip** (gestor de paquetes de Python)
- **Git** (opcional, para clonar el repositorio)
- **Un editor de código** (recomendado: VS Code, PyCharm, Sublime Text)
- **Navegador web** (Chrome, Firefox, Edge, etc.)

**Nota:** Se recomienda el uso de un entorno virtual para evitar conflictos de dependencias.

## 3 Instalación y configuración

### 3.1 1. Clonar el repositorio o descargar el código

Puedes clonar el repositorio si está en GitHub, o simplemente crear una carpeta y copiar los archivos proporcionados en este manual.

```
1 git clone https://github.com/tu_usuario/tu_repositorio.git
2 cd tu_repositorio
```

### 3.2 2. Crear y activar entorno virtual

```
1 python -m venv .venv
2 # Activar en Windows:
3 .venv\Scripts\activate
4 # Activar en Mac/Linux:
5 source .venv/bin/activate
```

### 3.3 3. Instalar Django

```
1 pip install django
```

### 3.4 4. Estructura de carpetas del proyecto

- mis-gastos/ (carpeta principal del proyecto)
- gastos/ (app principal)
- gastos/static/gastos/ (CSS, JS, imágenes)
- gastos/templates/gastos/ (archivos HTML)

Coloca las imágenes, archivos CSS y JS en las rutas señaladas.

## 4 Estructura de archivos y código fuente

### 4.1 1. models.py

```
1 from django.db import models
2
3 class Gasto(models.Model):
4     descripcion = models.CharField(max_length=120)
5     monto = models.DecimalField(max_digits=10, decimal_places=2)
6     fecha = models.DateField(auto_now_add=True)
```

### 4.2 2. views.py

```
1 from django.shortcuts import render, redirect
2 from .models import Gasto
3 from django.db.models import Sum
4
5 def landing(request):
6     gastos_count = Gasto.objects.count()
7     total_gastado = Gasto.objects.aggregate(total=Sum('monto'))['
8         total'] or 0
9     return render(request, 'gastos/landing.html', {
10         'gastos_count': gastos_count,
11         'total_gastado': f"{total_gastado:.2f}"
12     })
```

### 4.3 3. urls.py (de la app gastos)

```
1 from django.urls import path
2 from . import views
3
4 urlpatterns = [
5     path('', views.landing, name='landing'),
6     path('agregar/', views.formulario_gasto, name='formulario_gasto'),
7     path('historial/', views.listado_gastos, name='listado_gastos'),
8 ]
```

## 4.4 4. urls.py (del proyecto principal)

```
1 from django.contrib import admin
2 from django.urls import path, include
3
4 urlpatterns = [
5     path('admin/', admin.site.urls),
6     path('', include('gastos.urls')),
7 ]
```

## 4.5 5. settings.py (fragmento para archivos estáticos)

```
1 STATIC_URL = '/static/'
```

# 5 Archivos de plantilla (HTML)

## 5.1 landing.html

```
1 <!-- Fragmento representativo de landing.html -->
2 <div class="hero-content">
3     <h1>Controla tus gastos<br>y alcanza tus metas</h1>
4     <p>Lleva un seguimiento claro y gil de todos tus movimientos.<
5     <a href="{% url 'formulario_gasto' %}" class="btn-principal">
6     Empieza ahora</a>
7 </div>
```

## 5.2 formulario.html

```
1 <!-- Fragmento representativo de formulario.html -->
2 <form id="gastoForm" action="" method="post" class="form-gasto">
3     {% csrf_token %}
4     <div class="campo-form">
5         <label for="descripcion">Descripci n</label>
6         <input type="text" name="descripcion" id="descripcion"
7         required>
8     </div>
9     <div class="campo-form">
10        <label for="monto">Monto gastado</label>
11        <input type="number" step="0.01" name="monto" id="monto"
12        required>
13    </div>
14    <div id="form-message"></div>
15    <button type="submit" class="btn-principal">Agregar Gasto</
16    button>
17 </form>
```

## 5.3 listado.html

```
1 <!-- Fragmento representativo de listado.html -->
2 <ul class="gastos-list">
3     {% for gasto in gastos %}
4         <li class="gasto-item">
5             <span class="descripcion">{{ gasto.descripcion }}</span>
6             <span class="monto">${{ gasto.monto }}</span>
7         </li>
8     {% empty %}
9         <li class="gasto-vacio">No hay gastos registrados a n.</li>
10    {% endfor %}
11 </ul>
```

## 6 CSS (style.css)

```
1 /* Fragmento representativo de style.css */
2 body {
3     background: #f4f7fb;
4     font-family: 'Inter', Arial, sans-serif;
5     color: #253245;
6 }
7
8 .btn-principal {
9     background: linear-gradient(90deg, #1abc9c 0%, #19857b 100%);
10    color: #fff;
11    border-radius: 28px;
12    font-size: 1.25rem;
13    font-weight: 700;
14    border: none;
15 }
```

## 7 JavaScript (scripts.js)

```
1 document.addEventListener('DOMContentLoaded', function () {
2     // Validación y feedback en el formulario de gasto
3     const form = document.getElementById('gastoForm');
4     const msg = document.getElementById('form-message');
5     if (form && msg) {
6         form.addEventListener('submit', function (e) {
7             const descripcion = form.descripcion.value.trim();
8             const monto = parseFloat(form.monto.value);
9             if (!descripcion || isNaN(monto) || monto <= 0) {
10                 e.preventDefault();
11                 msg.textContent = "Por favor escribe una descripción válida y un monto mayor a 0.";
12                 msg.style.color = "#d32f2f";
13                 return false;
14             }
15         });
16     }
17 });
```

```
16     }  
17 });
```

## 8 Ejecución del proyecto

### 8.1 1. Migrar la base de datos

Antes de utilizar la aplicación, es necesario preparar la base de datos para que Django cree las tablas necesarias según los modelos definidos en el proyecto. Para ello, debes ejecutar los siguientes comandos en la terminal:

```
1 python manage.py makemigrations  
2 python manage.py migrate
```

¿Qué hacen estos comandos?

- **makemigrations**: Prepara archivos de migración que describen los cambios necesarios en la base de datos a partir de los modelos de Python.
- **migrate**: Aplica esos cambios, creando o actualizando las tablas en la base de datos real.

Es un paso fundamental para que la aplicación pueda guardar y mostrar información correctamente.

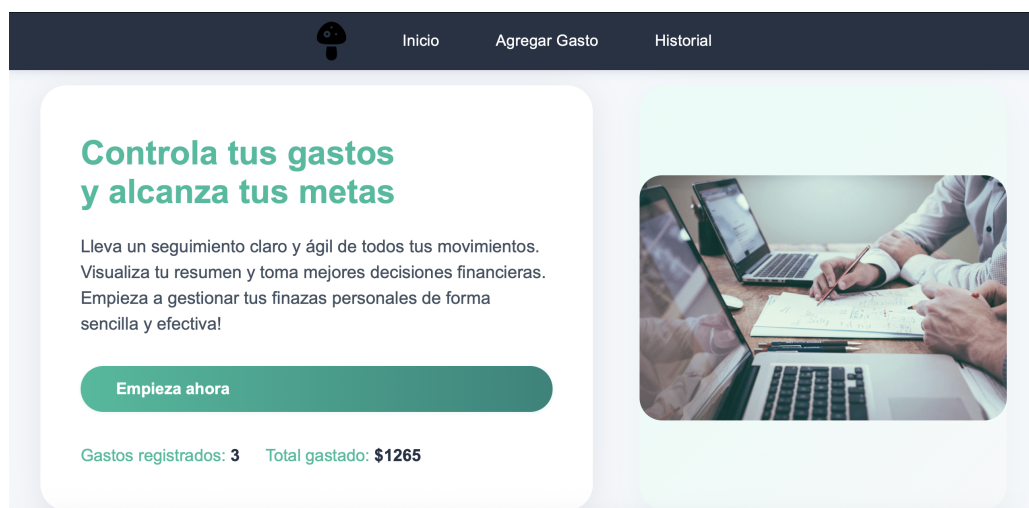
### 8.2 2. Ejecutar el servidor

```
1 python manage.py runserver
```

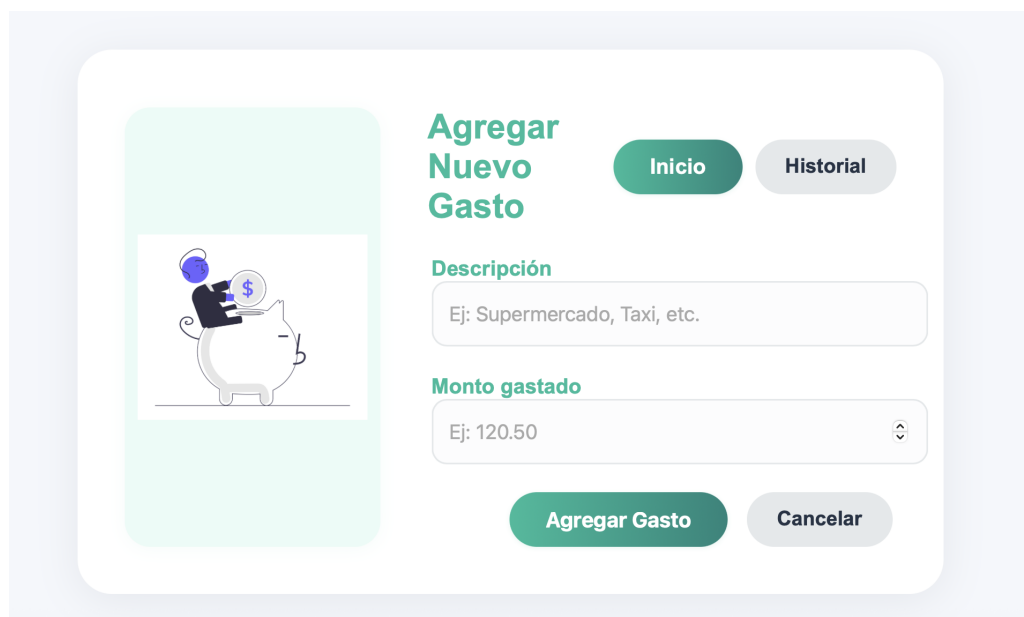
Abre tu navegador y entra a: <http://127.0.0.1:8000/>

## 9 Evidencias de funcionamiento

Página principal o landing



## Formulario para agregar gastos



Formulario para agregar gastos. El formulario tiene un título "Agregar Nuevo Gasto" en verde. A la izquierda hay un icono de un cerdito ahorrador con un signo de dólar. A la derecha hay dos botones: "Inicio" (verde) y "Historial" (gris). Debajo del título hay dos campos de texto: "Descripción" con el ejemplo "Ej: Supermercado, Taxi, etc." y "Monto gastado" con el ejemplo "Ej: 120.50". Al final hay dos botones: "Agregar Gasto" (verde) y "Cancelar" (gris).

## Historial de gastos



Historial de Gastos. El formulario tiene un icono de una persona con una lupa. A la izquierda del título "Historial de Gastos" hay un botón "Inicio" (verde) y a la derecha un botón "Agregar Gasto" (gris). Debajo del título hay un botón "Total gastado: \$1265" (gris). Debajo de eso hay un título "Mis gastos" en verde. A continuación hay una lista de tres gastos: "Pago de matricula" (\$1115.00), "compra de casa" (\$120.00) y "comida rapida" (\$30.00). Cada ítem de la lista está en un recuadro gris con el monto a la derecha en un recuadro verde.

## 10 Conclusión

La finalidad de este manual es servir como una guía clara para la correcta instalación, configuración y uso de la aplicación “Mis Gastos”. Cualquier usuario podrá implementarlo y verificar su funcionamiento siguiendo los pasos aquí detallados.

**Fin del Manual**