Informe de Pruebas Unitarias del Proyecto SIGEM

Fecha: 26 de mayo de 2025  
Desarrolladores, QA: Matías Larenas, Benjamín González, Constanza Pavez  
Proyecto: SIGEM

## Introducción

Este documento presenta el informe de pruebas unitarias realizadas al sistema SIGEM (Sistema de Gestión de Equipos y Materiales) desarrollado con Django y MySQL. El objetivo principal de estas pruebas es validar el correcto funcionamiento de los modelos de datos implementados en la aplicación, asegurando que cada componente del sistema se comporte conforme a los requerimientos establecidos.

## Objetivo

Como parte del proceso de aseguramiento de calidad del sistema SIGEM, se optó por implementar pruebas unitarias debido a su efectividad para validar el funcionamiento correcto de cada componente de forma aislada. Esta técnica permite detectar errores de manera temprana, asegurar la estabilidad del sistema y facilitar futuras modificaciones del código sin comprometer la funcionalidad existente.

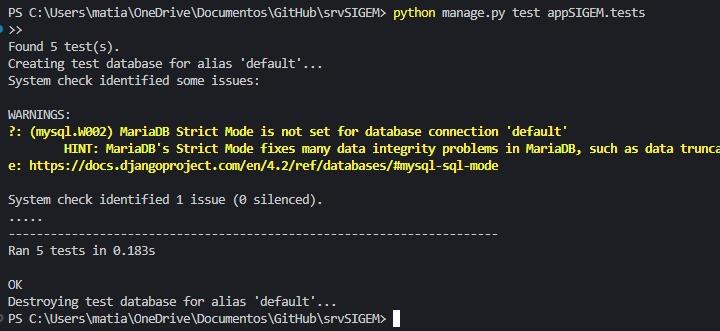
## Herramientas y Justificación.

La herramienta utilizada para la implementación de estas pruebas fue el módulo de pruebas **unittest** integrado en Django, ya que se ajusta de forma nativa al entorno del proyecto, simplifica la ejecución de los tests y ofrece una estructura clara para evaluar los resultados. Su integración con el ORM de Django también permite simular datos y escenarios realistas dentro de un entorno controlado de pruebas.

La decisión de aplicar esta técnica y herramienta responde a criterios de buenas prácticas profesionales y académicas, como el enfoque preventivo de errores, la automatización de validaciones y la mejora continua del software, competencias fundamentales que se espera dominen los estudiantes en su etapa final de formación.

## Módulos Evaluados y sin errores.

| Modelo | Estado | Descripción |
| --- | --- | --- |
| User | OK | Se creo un usuario personalizado correctamente utilizando el create\_user |
| Categoría | OK | Se validó la creación y representación de la categoría |
| TipoMaterial | OK | Se validó la creación y representación del tipo de material |
| Marca | OK | Se verificó la creación de un material, y el aumento automático del stock. |
| Material | OK | Se validó la creación de un material, y el aumento automático del stock |

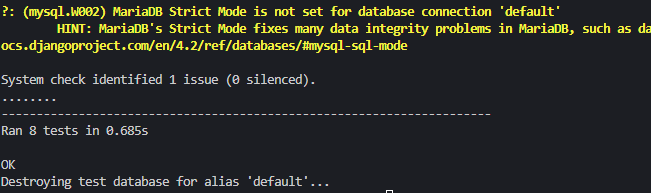


26 de Mayo del 2025

**Fueron creados nuevos módulos como Carrito, ItemCarrito, Solicitud y ItemSolicitud los cuales fueron evaluados en conjunto con los módulos anteriores y superan los tests sin errores.**

## 

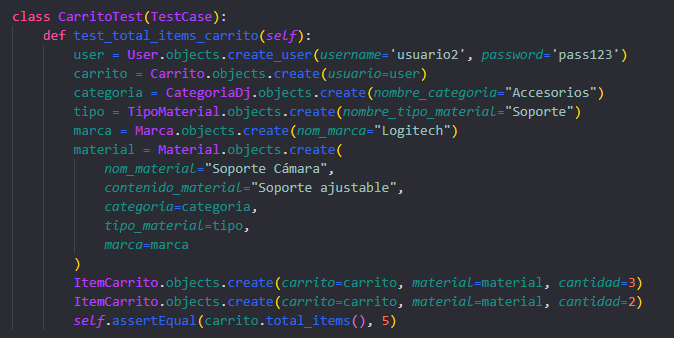
| Modelo | Estado | Descripción |
| --- | --- | --- |
| Carrito | OK | Se creó un carrito y se validó el cálculo correcto del total de ítems agregados. |
| ItemCarrito | OK | Se verificó la suma correcta de cantidades de materiales dentro del carrito. |
| Solicitud | OK | Se creó una solicitud validando la generación única de numero\_solicitud según el año. |
| ItemSolicitud | OK | Se creó un ítem de solicitud y se validó su representación en cadena (\_\_str\_\_). |

****

30 de mayo del 2025

**No se emite la prueba de la clase ItemCarrito por si sola, sino que se incluyó en CarritoTest.**

**Porque la funcionalidad principal de ItemCarrito(cantidad de materiales asociadas al carrito) se evalúa indirectamente al probar el método que lleva el total de ítems de Carrito. Se probó la integración de Carrito a ItemCarrito por funcionalidad.**

****

## Conclusión

Se ejecutaron 8 pruebas unitarias correspondientes a los modelos principales del sistema SIGEM, todas con resultados exitosos. Esto demuestra que las estructuras de datos, relaciones y comportamientos esperados funcionan correctamente. Como estudiante de Ingeniería en Informática, estas pruebas validan la calidad del sistema y refuerzan la importancia de integrar testing desde las primeras etapas del desarrollo.

**Pruebas ejecutadas desde:**

* appSIGEM/tests/test\_models.py

**Pruebas ejecutadas en el Terminal con:**

* **python** manage.py test appSIGEM.tests.test\_models