



Sistemas Expertos

“Pc Master”

Ernesto Misael Silva Muñoz

Registro: 22310168

Grupo: 7F

Ingeniería Mecatrónica

Mtro. Mauricio Alejandro Cabrera
Arellano

1. Introducción

En la actualidad, la variedad de componentes de hardware y computadoras portátiles en el mercado es abrumadora. Para un usuario promedio, entender especificaciones técnicas como "Frecuencia de reloj", "Núcleos CUDA" o "VRAM" puede resultar confuso, llevando a decisiones de compra erróneas (comprar un equipo muy costoso para tareas básicas o uno insuficiente para tareas pesadas).

El presente proyecto, titulado "**PC Master**", consiste en el desarrollo de un **Sistema Experto** diseñado para emular el criterio de un asesor técnico. El software interactúa con el usuario para determinar sus necesidades y presupuesto, y utiliza un motor de inferencia para filtrar y recomendar las opciones óptimas desde una base de conocimientos.

2. Objetivo General

Desarrollar una aplicación de escritorio que funcione como un sistema experto de selección de hardware, capaz de recomendar computadoras específicas basándose en reglas lógicas de uso (Basico, Profesional, Gaming) y restricciones presupuestarias, ofreciendo una interfaz gráfica intuitiva y visualmente atractiva.

3. Arquitectura Técnica

El sistema fue construido utilizando las siguientes tecnologías:

- **Lenguaje de Programación:** Python 3.10+.
- **Interfaz Gráfica (Frontend):** Librería CustomTkinter. Se implementó un diseño "Pixel Perfect" basado en prototipos de Figma, utilizando lienzos (Canvas) y manejo de imágenes para lograr una estética moderna y limpia.
- **Gestión de Datos (Backend):** Base de datos relacional **SQLite**.
- **Manejo de Recursos:** Librería Pillow (PIL) para el procesamiento y renderizado dinámico de imágenes de los productos.

4. Base de Conocimientos y Reglas de Inferencia

El "cerebro" del sistema reside en su base de datos y en las reglas programadas en Python. La base de datos sistema_experto.db almacena información técnica detallada (Benchmark de CPU, Benchmark de GPU, RAM, Almacenamiento, etc.).

El **Motor de Inferencia** aplica las siguientes reglas heurísticas según la selección del usuario:

| Perfil de Usuario | Reglas Lógicas Aplicadas (IF-THEN) | Criterio de Ordenamiento |
|-------------------|---|---------------------------|
| Básico | IF Precio <= Presupuesto | Menor Precio (Economía) |
| Profesional | IF Precio <= Presupuesto AND RAM >= 16GB | Mejor CPU (Procesamiento) |
| Gamer | IF Precio <= Presupuesto AND GPU_Dedicada == True | Mejor GPU (Gráficos) |

5. Desarrollo e Implementación

El desarrollo se dividió en tres módulos principales:

1. **Módulo de Datos:** Se creó un script (llenar_bd.py) encargado de limpiar y poblar la base de datos con información actualizada de laptops de diversas marcas (Asus, Dell, HP, Apple, etc.), asegurando la integridad de los datos y la correcta asociación con sus imágenes .jpg.
2. **Módulo de Interfaz:** Se diseñó una ventana principal de tamaño fijo (1100x750) que integra una imagen de fondo diseñada en Figma. Se utilizaron técnicas avanzadas de superposición para colocar controles (Sliders, Listas) sobre el diseño sin generar bordes visuales indeseados.

3. **Módulo de Consulta:** Al presionar el botón "Analizar", el sistema conecta con la BD, ejecuta la sentencia SQL generada dinámicamente según las reglas del punto 4, y renderiza los resultados en tarjetas visuales con foto, precio y especificaciones.

6. Demostración de Funcionamiento

A continuación, se presentan las evidencias del sistema operando en diferentes escenarios:

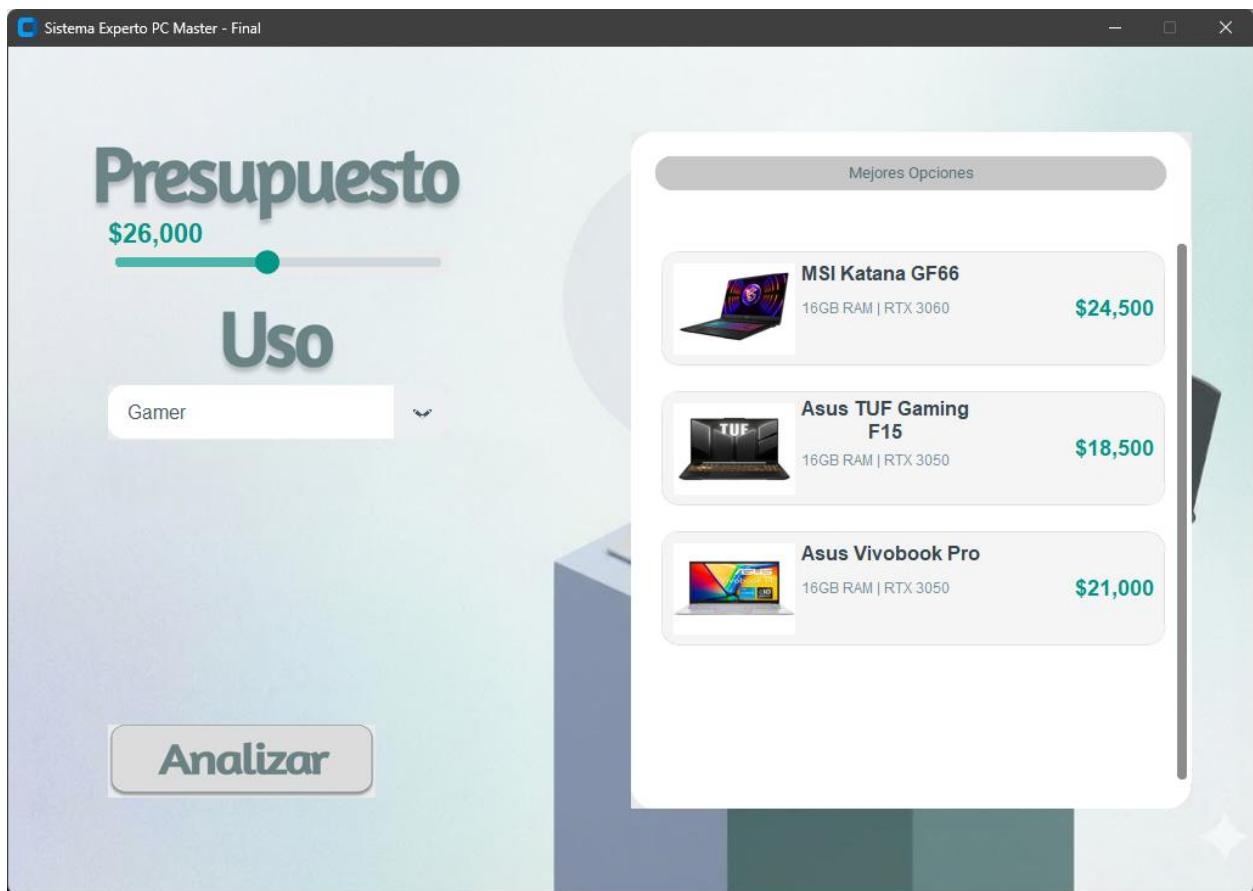
- **6.1. Pantalla de Inicio**

Se aprecia la interfaz limpia basada en el diseño de Figma, con controles de presupuesto y tipo de uso integrados visualmente.



- **6.2. Caso de Uso: Perfil Gamer**

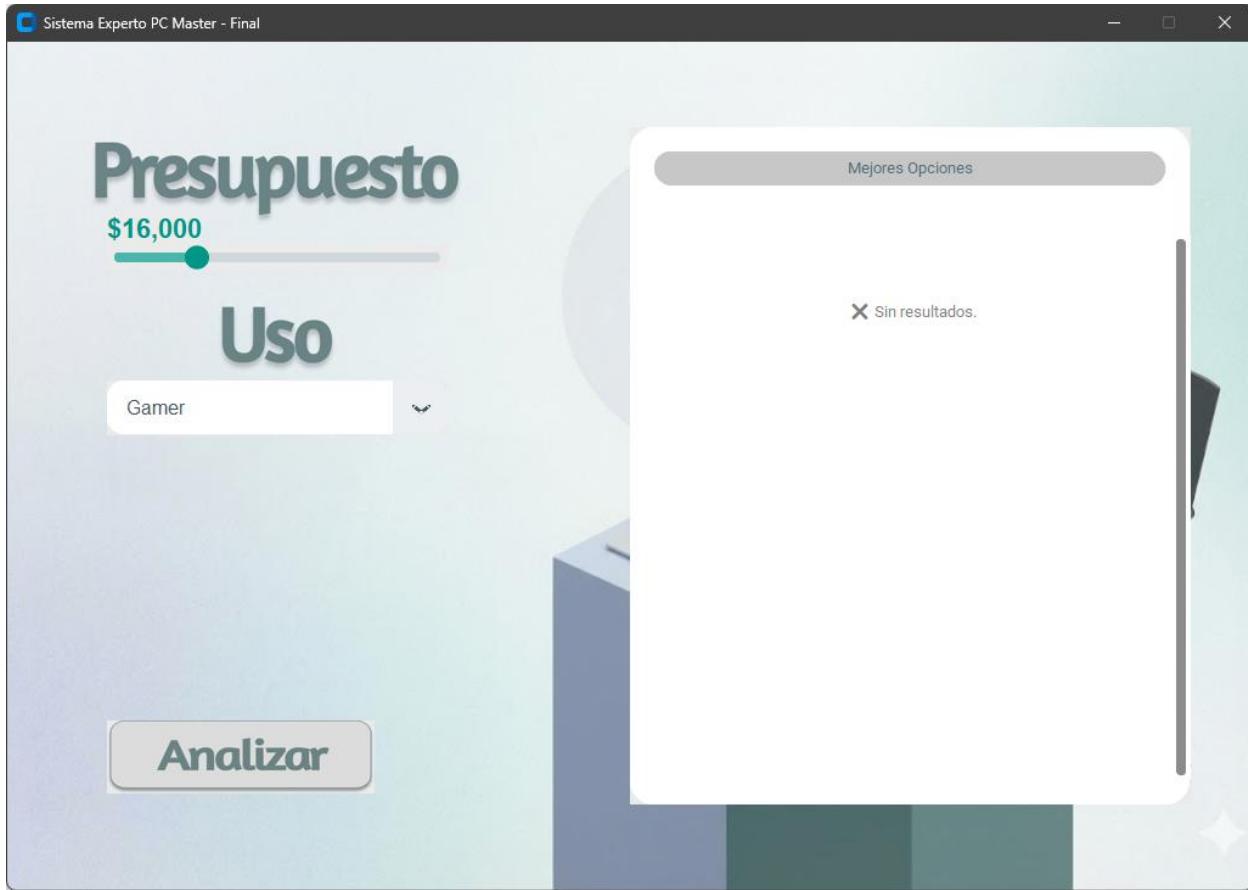
El usuario selecciona un presupuesto medio-alto y el perfil "Gamer". El sistema filtra equipos que contienen tarjetas gráficas dedicadas (NVIDIA RTX) y los muestra en tarjetas interactivas.



[

- **6.3. Manejo de Excepciones**

Si el presupuesto es insuficiente para las exigencias del perfil (ej. Gamer con muy bajo presupuesto), el sistema informa adecuadamente que no hay coincidencias.



7. Video Demostrativo

Para una visualización completa del flujo de trabajo y la respuesta en tiempo real de la aplicación, se adjunta el siguiente video explicativo:

Enlace a YouTube: https://youtu.be/_1EAhnDkXII

8. Conclusión

El proyecto "PC Master" cumple exitosamente con el objetivo de simplificar la selección de hardware. La implementación de una base de datos SQLite permite que el sistema sea escalable (fácil de agregar nuevos productos), mientras que el uso de reglas lógicas asegura que las recomendaciones sean técnicamente acertadas. Además, la integración de un diseño personalizado de Figma demuestra la capacidad de Python para crear interfaces de usuario profesionales y estéticamente agradables.