

SISTEMA EXPERTO “CopaMaestra”

Stiven Alejandro Carvajal

Victor Manuel Garcia

Natalia Velez Orjuela

Universidad Autónoma de Manizales

Facultad de Ingeniería de Sistemas

Paradigmas de Lenguajes de Programación

2025

Resumen

Proyecto Sistema Experto "CopaMaestra"

El estudio de los lenguajes de programación a nivel conceptual permite entender mejor las características comunes entre ellos. Este conocimiento ayuda a los programadores a utilizar los lenguajes de forma más efectiva, reconociendo sus fortalezas y limitaciones.

Este documento presenta el desarrollo de un sistema experto diseñado para el curso de Paradigmas de Lenguajes de Programación. El sistema fue implementado utilizando **Python para la interfaz gráfica y lógica de control**, y **Prolog como motor de inferencia lógica**. El objetivo principal es recomendar cócteles personalizados a los usuarios a partir de su perfil demográfico y académico, basándose en una base de datos de calificaciones históricas.

Este sistema experto funciona de la siguiente manera:

El sistema solicita al usuario que ingrese los siguientes datos:

- Edad
- Género
- Estrato socioeconómico
- Carrera universitaria

Con esta información, el sistema realiza una inferencia lógica sobre los cócteles que usuarios similares han calificado positivamente. El motor de reglas en **Prolog** identifica patrones de

comportamiento y preferencias, permitiendo recomendar 2 o 3 cócteles que mejor se adapten al perfil ingresado.

Componentes del sistema

- **Base de hechos** en Prolog: contiene datos de usuarios y calificaciones previas.
- **Reglas de inferencia**: definen perfiles favorables a partir de patrones comunes.
- **Interfaz en Python (Tkinter)**: donde el usuario ingresa sus datos y visualiza los resultados.
- **Ventana de cocteles**: los cocteles que se adaptan al perfil del usuario
- **Ventana de detalles del cóctel**: muestra imagen (PNG), instrucciones y una lista de ingredientes.

Tabla de Contenidos

1. Introducción.....	7
1.1. Descripción del sistema experto.....	7
1.1.1 Tecnologías utilizadas.....	8
2. Planteamiento del Problema.....	9
2.1 Problema.....	9
2.2. Resolución del problema.....	10
3. Justificación del proyecto.....	10
3.1 Relevancia del proyecto.....	10
4. Análisis del Problema.....	11
4.1 Casos de uso.....	11
4.1.1 Caso de uso: Ingresar datos del usuario.....	11
4.1.2 Caso de uso: Cargar datos históricos de calificaciones.....	11
4.1.3 Caso de uso: Establecer comunicación entre Python y Prolog.....	11
4.1.4 Caso de uso: Inicializar motor lógico en Prolog.....	11
4.1.5 Caso de uso: Procesar coincidencias del perfil.....	12
4.1.6 Caso de uso: Calcular recomendación.....	12
4.1.7 Caso de uso: Generar recomendaciones de cócteles.....	12
4.1.8 Caso de uso: Visualizar cócteles.....	12
4.1.9 Caso de uso: Mostrar detalles del cóctel.....	12
4.1.10 Caso de uso: Regresar a pantalla principal.....	13
4.1.11 Caso de uso: Salir de la aplicación.....	13
4.2 Diagrama de casos de uso.....	13
4.3 Diagrama de clases.....	15
5. Diseño de la solución propuesta.....	15
5.1 Descripción general del sistema.....	15
5.1.1 Resumen del funcionamiento.....	15
5.1.2 Base de conocimientos (Hechos y reglas).....	17
5.1.3 Ejemplos de hechos y reglas en el sis.....	18
5.1.4 Motor de inferencia.....	20
5.1.5 Interfaz de usuario.....	21
5.1.6 Sistema de recomendaciones.....	23
6. Pruebas e interacciones.....	24
6.1 Ejemplos del uso del sistema.....	24
7. Manual de usuario.....	29
Utilización.....	29
8. Referencias.....	35
9. Conclusiones.....	37

Lista de tablas

Lista de figuras

- Figura 1.** Diagrama de casos de uso del sistema experto – pág. 15
- Figura 2.** Diagrama de clases del sistema experto – pág. 16
- Figura 3.** Ejemplo de regla llamada `recolectar_puntajes` – pág. 20
- Figura 4.** CopaMaestra - Ventana principal – pág. 26
- Figura 5.** CopaMaestra - Ventana que muestra los 3 cócteles recomendados – pág. 27
- Figura 6.** CopaMaestra - Ventana de detalles del cóctel (1) – pág. 28
- Figura 7.** CopaMaestra - Ventana de detalles del cóctel (2) – pág. 29
- Figura 8.** Clonación del repositorio desde GitHub – pág. 30
- Figura 9.** Instalación de librería Pandas – pág. 31
- Figura 10.** Instalación de librería Tkinter – pág. 31
- Figura 11.** Instalación de librería PySwip – pág. 31
- Figura 12.** Estructura del sistema experto “CopaMaestra” – pág. 32
- Figura 13.** Ejecución del proyecto en terminal – pág. 32
- Figura 14.** CopaMaestra - Ventana principal con formulario – pág. 33
- Figura 15.** Carga de datos históricos – pág. 34
- Figura 16.** Confirmación de carga de datos – pág. 34
- Figura 17.** Recomendación de cóctel – pág. 34
- Figura 18.** Ventana de opciones de cócteles – pág. 35
- Figura 19.** Ventana final con detalles del cóctel – pág. 35

1. Introducción

En la actualidad, la personalización se ha convertido en un factor importante para mejorar la experiencia del usuario, especialmente en sectores como el entretenimiento, la gastronomía y el comercio electrónico. Teniendo en cuenta esto, los sistemas expertos han empezado a tener más relevancias pues permiten la automatización de decisiones basadas en conocimiento previo y reglas lógicas.

Este proyecto propone el desarrollo de un sistema experto de Recomendación de Cócteles, el cual utiliza técnicas de inferencia lógica y análisis de datos para ofrecer sugerencias personalizadas de cócteles según el perfil demográfico y carrera universitaria.

Mediante la integración de tecnologías como Prolog y Python, el sistema analiza un conjunto de registros de calificaciones de cócteles, identifica las coincidencias relevantes y calcula una puntuación para cada bebida. Como resultado, se dan al usuario las tres opciones más adecuadas según su perfil, junto con sus ingredientes e instrucciones de preparación.

Este sistema demuestra la utilidad de la programación lógica en la cotidianidad y el potencial de los sistemas expertos para ofrecer soluciones eficientes y personalizadas.

1.1. Descripción del sistema experto

El sistema experto CopaMaestra, es una solución que ofrece recomendaciones personalizadas de bebidas con base en características individuales de cada usuario. Este sistema cuenta con técnicas de inferencia lógica y análisis de datos para dar recomendaciones de tres cócteles que mejor se ajustan al perfil del usuario.

- Ingreso de datos: El usuario completa un formulario con información como edad, estrato socioeconómico, carrera y género.
- Carga de datos históricos: Se utilizan registros previos que contienen calificaciones de cócteles dadas por usuarios con perfiles variados.
- Procesamiento en Prolog: Se ejecutan reglas de inferencia en Prolog para identificar coincidencias entre el usuario actual y usuarios históricos.
- Recomendaciones: Las recomendaciones se generan a partir de una **base de datos con 6000 calificaciones históricas** de usuarios. El sistema utiliza la información ingresada por el usuario (edad, estrato, carrera y género) para **buscar coincidencias con perfiles similares** y aplicar un sistema de **ponderación por coincidencia y calificación**. Cada coincidencia aporta un puntaje al cóctel según el tipo de match y la calificación dada en el pasado. Finalmente, se suman todos los puntajes y se recomienda el **top 3 de cócteles con mayor puntuación acumulada**.

1.1.1 Tecnologías utilizadas

- **Prolog:** Motor de inferencia lógica, define y evalúa las reglas, además de permitir el procesamiento y análisis eficiente de datos.
- **Python + Tkinter:** Desarrolla la interfaz gráfica amigable y funcional.

- **PySwip:** Facilita la integración entre Python y Prolog para intercambiar datos en tiempo real.
- **-Pandas:** Se utiliza para la carga de la base de datos de conocimiento.

2. Planteamiento del Problema.

2.1 Problema

En el contexto actual de los sistemas expertos, es común encontrar aplicaciones enfocadas en áreas como el comercio electrónico, el entretenimiento o la educación. Sin embargo, pocas propuestas exploran la personalización de experiencias gastronómicas o de coctelería a partir del perfil del usuario.

En este sentido, surge la necesidad de desarrollar un sistema capaz de recomendar cócteles personalizados, considerando características demográficas y académicas de los usuarios. El objetivo es ir más allá de las recomendaciones genéricas, para ofrecer sugerencias que se ajusten mejor a los gustos y patrones observados en personas con perfiles similares.

Por ello, se propone el desarrollo de **CopaMaestra**, un sistema experto que, mediante la integración de Python y Prolog, sea capaz de analizar datos, identificar patrones de preferencia y sugerir cócteles que se adapten al perfil del usuario ingresado. De esta manera, se busca mejorar la experiencia de recomendación a través de un enfoque basado en usuarios que compartieron sus experiencias con esos cócteles.

2.2. Resolución del problema

Para responder a la necesidad de recomendaciones personalizadas en el ámbito de la coctelería, se desarrolló CopaMaestra, un sistema experto que combina la lógica de Prolog con Python. A partir del perfil del usuario (edad, género, estrato y carrera) y un historial de calificaciones, el sistema identifica coincidencias relevantes y asigna puntajes a cada cóctel considerando las valoraciones previas de otros usuarios. Además, emplea un modelo de afinidad que pondera las opiniones de usuarios con características similares, permitiendo generar recomendaciones ajustadas a los gustos individuales con mayor precisión.

Gracias a esto, CopaMaestra ofrece al usuario tres cócteles recomendados que se ajustan mejor al perfil que ingresó, mejorando la experiencia con sugerencias personalizadas, precisas y basadas en inferencias lógicas.

3. Justificación del proyecto

3.1 Relevancia del proyecto

Este sistema representa una aplicación de los sistemas expertos en el área del entretenimiento y la gastronomía, demostrando de qué manera se pueden adaptar a preferencias humanas subjetivas. Su relevancia radica en:

- **Personalización de experiencias:** Ofrece recomendaciones precisas al basarse en perfiles reales y criterios específicos.

- **Uso de inferencia lógica:** Implementa técnicas que permiten tomar decisiones explicables.
- **Integración:** Combina conocimientos de programación, lógica, estadística y diseño de interfaces, lo que lo convierte en un proyecto completo.
- **Aplicabilidad real:** Puede implementarse en bares, apps móviles de bebidas, asistentes virtuales o sistemas de recomendación.

4. Análisis del Problema

4.1 Casos de uso

4.1.1 Caso de uso: Ingresar datos del usuario

Actor: Usuario

Descripción: El usuario proporciona su edad, género, estrato y carrera para personalizar las recomendaciones.

4.1.2 Caso de uso: Cargar datos históricos de calificaciones

Actor: CopaMaestra

Descripción: Al iniciar el sistema se cargan automáticamente los registros históricos de calificaciones de cócteles.

4.1.3 Caso de uso: Establecer comunicación entre Python y Prolog

Actor: PySwip

Descripción: El sistema usa PySwip para enviar consultas a Prolog y recibir los resultados para procesar las recomendaciones.

4.1.4 Caso de uso: Inicializar motor lógico en Prolog

Actor: CopaMaestra

Descripción: El usuario puede ver los tres cócteles sugeridos según su perfil en una ventana de los cuales puede elegir.

4.1.5 Caso de uso: Procesar coincidencias del perfil

Actor: Prolog

Descripción: El sistema experto compara el perfil del usuario con los perfiles de los registros históricos para encontrar similitudes.

4.1.6 Caso de uso: Calcular recomendación

Actor: CopaMaestra

Descripción: Se activa el motor lógico en Prolog mediante PySwip para ejecutar las reglas de inferencia.

4.1.7 Caso de uso: Generar recomendaciones de cócteles

Actor: CopaMaestra

Descripción: El sistema selecciona los tres cócteles que se adaptan más al perfil del usuario.

4.1.8 Caso de uso: Visualizar cócteles

Actor: CopaMaestra

Descripción: El usuario puede ver los tres cócteles sugeridos según su perfil en una ventana de los cuales puede elegir.

4.1.9 Caso de uso: Mostrar detalles del cóctel

Actor: CopaMaestra

Descripción: Con el uso de un redireccionamiento para cada cóctel el usuario puede ver en otra ventana la imagen del cóctel, los ingredientes y las instrucciones de preparación.

4.1.10 Caso de uso: Regresar a pantalla principal

Actor: CopaMaestra

Descripción: Desde la vista de detalles el usuario puede volver a la ventana principal de ingreso de datos.

4.1.11 Caso de uso: Salir de la aplicación

Actor: CopaMaestra

Descripción: El usuario puede cerrar la ventana o finalizar el uso del sistema.

4.2 Diagrama de casos de uso

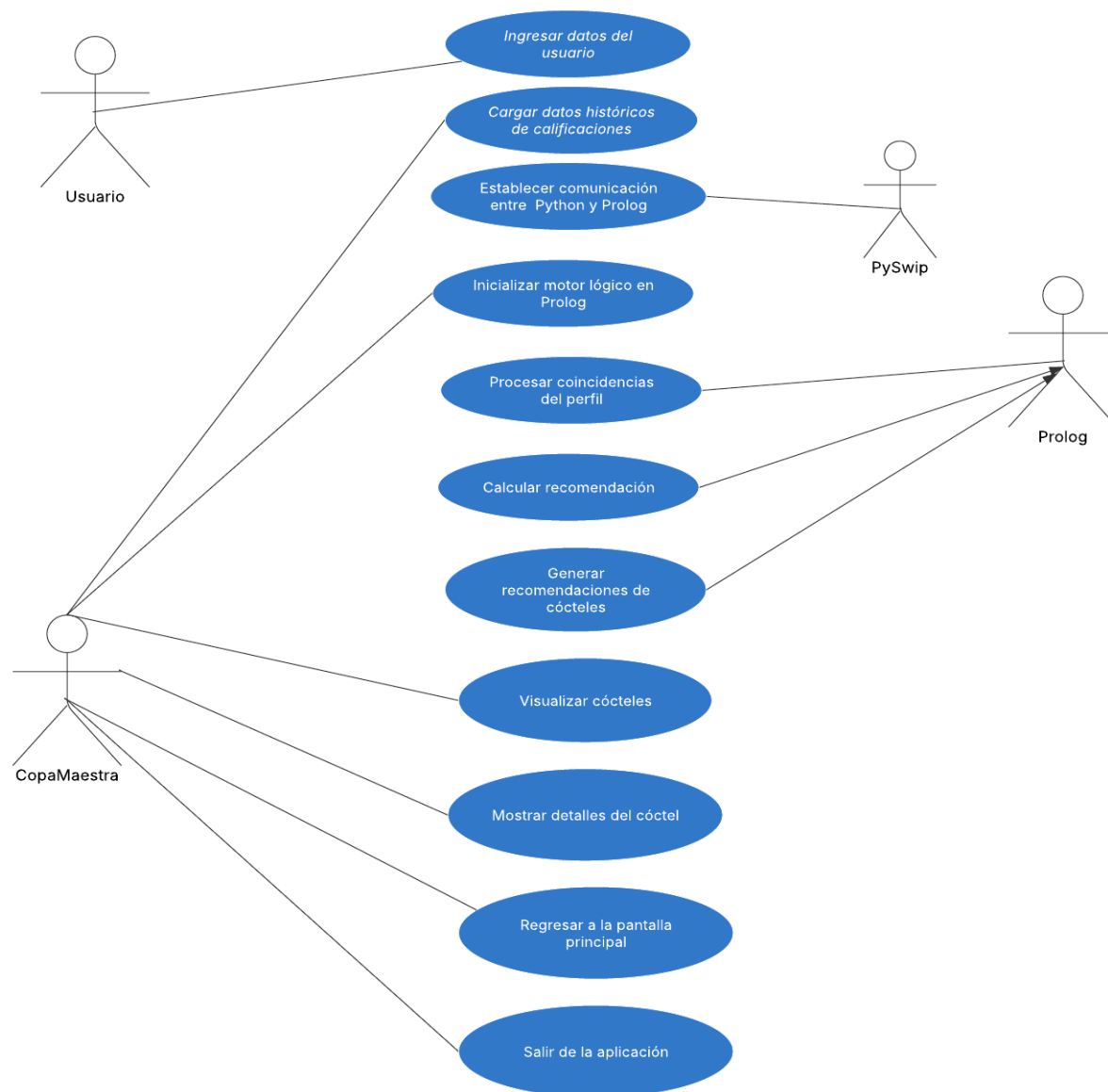


Figura 1. Diagrama de casos de uso del sistema experto

4.3 Diagrama de clases

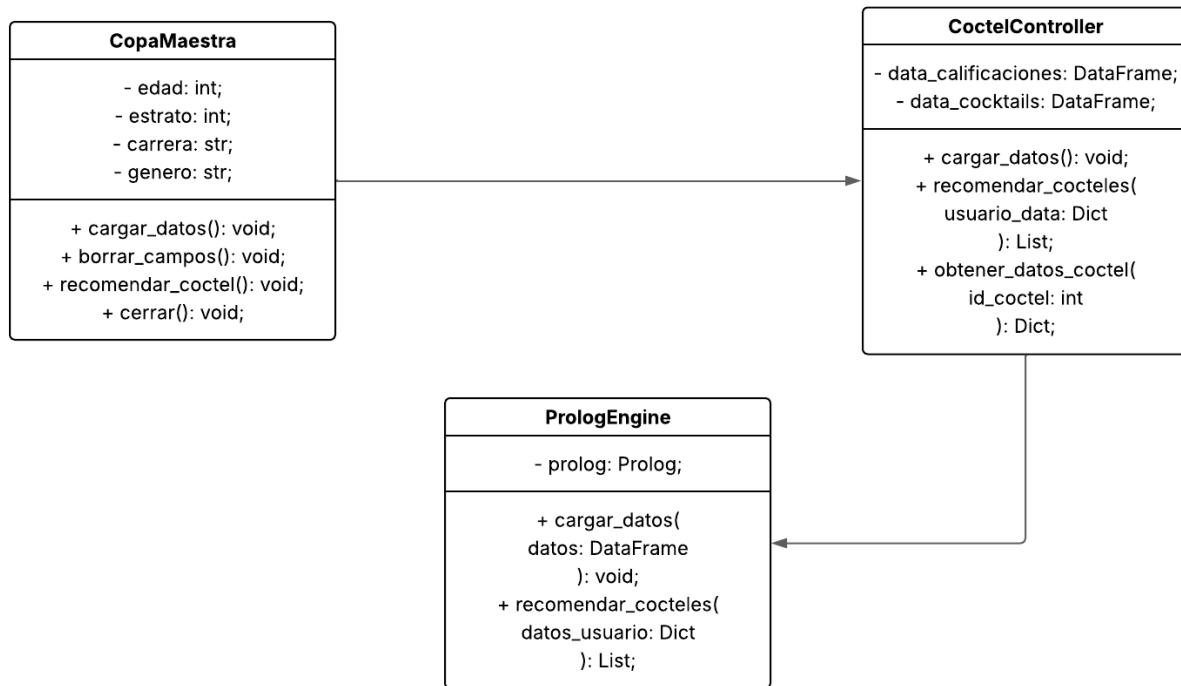


Figura 2. Diagrama de clases de uso del sistema experto

5. Diseño de la solución propuesta

5.1 Descripción general del sistema

5.1.1 Resumen del funcionamiento

Este sistema experto recomienda cócteles personalizados con base en las características de un usuario, tales como edad, estrato, carrera y género, utilizando la experiencia de usuarios anteriores almacenada en una base de datos de 6000 registros históricos.

El usuario ingresa su información personal a través de un formulario en una interfaz gráfica. Se cargan automáticamente los datos históricos de calificaciones de cócteles.

El sistema utiliza Prolog para encontrar coincidencias entre el nuevo usuario y los datos existentes.

Cada coincidencia encontrada otorga una puntuación al cóctel correspondiente, dependiendo de su tipo (exacta, por rango, o sin carrera) y la calificación asociada.

Finalmente, se seleccionan y recomiendan los 3 cócteles con mayor puntaje acumulado.

Tecnologías utilizadas

Prolog – Motor de inferencia lógica.

Python + Tkinter – Desarrollo de la interfaz gráfica.

Pandas – Procesamiento y análisis de los datos.

PySwip – Librería para la integración entre Python y Prolog.

Lógica del motor

El motor de inferencia clasifica las coincidencias en tres niveles según su precisión:

Match exacto: todos los atributos coinciden.

Match por rango: se relajan edad y estrato.

Match sin carrera: se relajan edad, estrato y carrera.

Cada tipo de coincidencia aplica una función de pesos distinta, en la cual las calificaciones altas suman puntos y las calificaciones bajas los restan. El sistema suma todos los pesos obtenidos por cóctel, y con base en eso determina cuáles son los más recomendables.

Básicamente lo que hacemos es que se cargan los datos → Se insertan como hechos en Prolog → Se lanza la inferencia → Se calcula el puntaje → Se selecciona el Top 3 de cócteles recomendados.

5.1.2 Base de conocimientos (Hechos y reglas)

El sistema experto se apoya en **dos bases de datos principales** que conforman su base de conocimiento:

a) base_calificaciones_cocteles_6000.csv

Contiene **6000 registros reales simulados** de calificaciones que usuarios han otorgado a distintos cócteles. Cada registro tiene los siguientes campos:

- Edad: edad del usuario.
- Estrato: nivel socioeconómico del usuario.
- Carrera: área de estudio o profesión.
- Genero: género del usuario.
- Coctel_ID: identificador único del cóctel calificado.
- Calificacion: valor del 1 al 5 que representa qué tanto le gustó el cóctel al usuario.

Esta base es cargada en Prolog como hechos y es utilizada por el motor de inferencia para encontrar patrones de preferencia con base en coincidencias entre usuarios.

b) cocktails_con_sabores.xlsx

Contiene información detallada de más de 100 cócteles, incluyendo sus atributos principales:

- id: identificador único del cóctel.
- Drink: nombre del cóctel.
- DrinkThumb: enlace a la imagen del cóctel.
- Inuactions: instrucciones de preparación.
- Ingredient1 a Ingredient6: ingredientes utilizados.
- Measure1 a Measure6: medidas de cada ingrediente.
- Flavor: sabor predominante del cóctel (ej: dulce, amargo, fresco, ácido).

Esta base se utiliza en la interfaz gráfica para mostrar los detalles visuales y descriptivos de cada cóctel recomendado, permitiendo que el usuario conozca su preparación, ingredientes, medidas y sabor antes de elegir.

5.1.3 Ejemplos de hechos y reglas en el sis

Contiene hechos como: calificación(22, 2, 'diseño', 'femenino', 17222, 4).

Las reglas definen coincidencias exactas o parciales, aplicando pesos distintos y considerando la calificación para cada coincidencia.

A continuación vamos a explicar una de las reglas más importantes de nuestro proyecto la cual permite recolectar los puntajes de los cócteles para así determinar finalmente los 3 mejores y lograr la inferencia.

```
recolectar_puntajes(Edad, Estrato, Carrera, Genero, Lista) :-
    setof(
        Puntaje-Id,
        (coctel_posible(Id), puntaje_total(Edad, Estrato, Carrera, Genero, Id,
Puntaje)),
        ListaOrdenada
    ),
    reverse(ListaOrdenada, ListaDesc),
    extraer_top(ListaDesc, Lista).
```

Figura 3. Ejemplo de regla llamada recolectar_puntajes

Dada la información del usuario (Edad, Estrato, Carrera, Genero), esta regla construye una lista con los identificadores de los 3 cócteles mejor puntuados por usuarios similares.

setof reúne todas las instancias de Puntaje-Id que satisfacen (coctel_posible(Id), puntaje_total(..., Id, Puntaje)), y las ordena en forma ascendente.

Para cada Id que existe en la base de datos (coctel_posible(Id)), Prolog calcula el Puntaje total de ese cóctel para el perfil dado, y construye un término Puntaje-Id.

El resultado se unifica en ListaOrdenada, que queda ordenada de menor a mayor según el Puntaje y aplicamos reverse(ListaOrdenada, ListaDesc) para obtener la lista de mayor a menor. Por último, extrae los primeros 1, 2 o 3 elementos (según la longitud disponible), devolviendo sólo los Id de los cócteles con mayor puntuación. Los cuales serán la recomendación que se le mostrará al usuario en la interfaz.

5.1.4 Motor de inferencia

El motor de inferencia desarrollado en Prolog es el núcleo del sistema experto. Su función es analizar las características de un usuario nuevo y, a partir de una base de conocimientos compuesta por 6000 registros históricos de calificaciones de cócteles, determinar cuáles son los más apropiados para recomendar.

El proceso de inferencia se basa en **reglas lógicas** que clasifican la similitud entre usuarios en tres niveles de coincidencia:

1. **Match exacto:** Coinciden todos los atributos (edad, estrato, carrera y género).
2. **Match por rango:** Se ajustan los atributos de edad (± 5 años) y estrato (± 2 niveles), pero la carrera y el género siguen coincidiendo.
3. **Match sin carrera:** Se ajustan edad y estrato como en el caso anterior, pero la carrera ya no se considera, manteniendo solo el género como filtro.

Cada tipo de coincidencia tiene una **función de ponderación distinta** que transforma la calificación (de 1 a 5) en un **peso numérico**, de acuerdo a su nivel de confiabilidad. Por ejemplo:

- Una calificación de 5 con match exacto suma +1 punto.
- Una calificación de 2 con match por rango resta -0.2 puntos.
- Una calificación de 3 (neutral) no aporta ni resta.

Todas las coincidencias encontradas se agrupan por ID de cóctel, y se calcula un **puntaje total acumulado** para cada uno. Finalmente, se seleccionan los tres cócteles con mayor puntuación como resultado de la inferencia.

Esta lógica permite que el sistema sea robusto y flexible: **recompensa fuertemente las coincidencias exactas** y penaliza levemente las coincidencias débiles, lo que mejora la precisión de las recomendaciones sin dejar de ofrecer opciones incluso cuando no hay coincidencias perfectas.

5.1.5 Interfaz de usuario

El sistema experto cuenta con una interfaz gráfica desarrollada en Python con Tkinter, diseñada para guiar de forma intuitiva al usuario a través del proceso de recomendación. Esta interfaz está compuesta por tres ventanas principales:

1. Ventana Principal

Es la pantalla inicial del sistema. Aquí el usuario ingresa su información personal:

Edad

Estrato socioeconómico

Carrera universitaria o profesión

Género

Luego, el usuario puede pulsar el botón “Recomendar Cóctel”, que activa el motor de inferencia y muestra una nueva ventana con los resultados.

2. Ventana de Recomendaciones

Una vez calculada la inferencia en Prolog, esta ventana presenta al usuario los 3 cócteles recomendados, cada uno con un botón individual. El usuario puede hacer clic en cualquiera para ver más detalles.

Además, la ventana permite cerrar fácilmente la sesión de recomendaciones.

3. Ventana de Detalle del Cóctel

Al hacer clic sobre un cóctel recomendado, se abre una tercera ventana que muestra información específica del cóctel seleccionado:

Nombre del cóctel

Imagen cargada desde una URL real

Instrucciones de preparación

Lista de ingredientes con sus respectivas medidas

Sabor característico del cóctel

Esta vista le permite al usuario explorar a profundidad la opción sugerida antes de tomar una decisión. También incluye un botón para cerrar y regresar a la ventana anterior.

5.1.6 Sistema de recomendaciones

El sistema experto está diseñado para ofrecer recomendaciones personalizadas de cócteles a partir de la información suministrada por el usuario. Tras obtener las recomendaciones, se abre automáticamente una segunda ventana donde el usuario puede visualizar los nombres de los cócteles sugeridos. Cada uno se presenta como un botón interactivo.

Al hacer clic en cualquiera de los cócteles recomendados, se despliega una ventana, que proporciona información ampliada sobre la bebida seleccionada, incluyendo:

- Nombre del cóctel
- Imagen ilustrativa
- Instrucciones de preparación
- Ingredientes acompañados de sus medidas
- Sabor predominante de la bebida

5.1.7 Sistema de explicaciones

El sistema justifica sus recomendaciones a través de un **modelo de ponderación basado en reglas declarativas**, implementado en Prolog. Cada recomendación se construye a partir de la experiencia de usuarios previos, evaluando las calificaciones que dieron a los cócteles según el nivel de coincidencia con el usuario actual.

Se utilizan tres tipos de coincidencias:

- **Coincidencia Exacta** (Match 1): edad, estrato, carrera y género idénticos.
- **Coincidencia Parcial** (Match 2): edad y estrato dentro de un rango, misma carrera y género.
- **Coincidencia General** (Match 3): solo se consideran edad y estrato similares.

Cada coincidencia aporta un **peso a un cóctel específico**, ajustado según la calificación registrada por usuarios anteriores. Las calificaciones más altas generan pesos positivos; las bajas, negativos.

El sistema **suma todos los pesos asociados a cada cóctel**, considerando todos los matches posibles. Finalmente, ordena los cócteles por puntaje total y selecciona los **tres con mejor rendimiento**, como recomendación final.

6. Pruebas e interacciones

6.1 Ejemplos del uso del sistema

Inicialmente corremos el programa desde la terminal con el comando python [main.py](#)

Después el abre la primera ventana



COPA MAESTRA

Edad: 20

Estrato: 4

Carrera: sistemas

Género: masculino

Cargar Datos

Recomendar Cóctel

Borrar Campos

Cerrar

Figura 4. Copa Maestra ventana 1

Le damos en cargar datos y luego en recomendar coctel y abre la siguiente ventana}



Figura 5. Copa Maestra ventana que muestra los 3 cocteles

Seguidamente cuando le damos en un cóctel. Muestra toda la informacion como lo hemos escrito a lo largo del documento

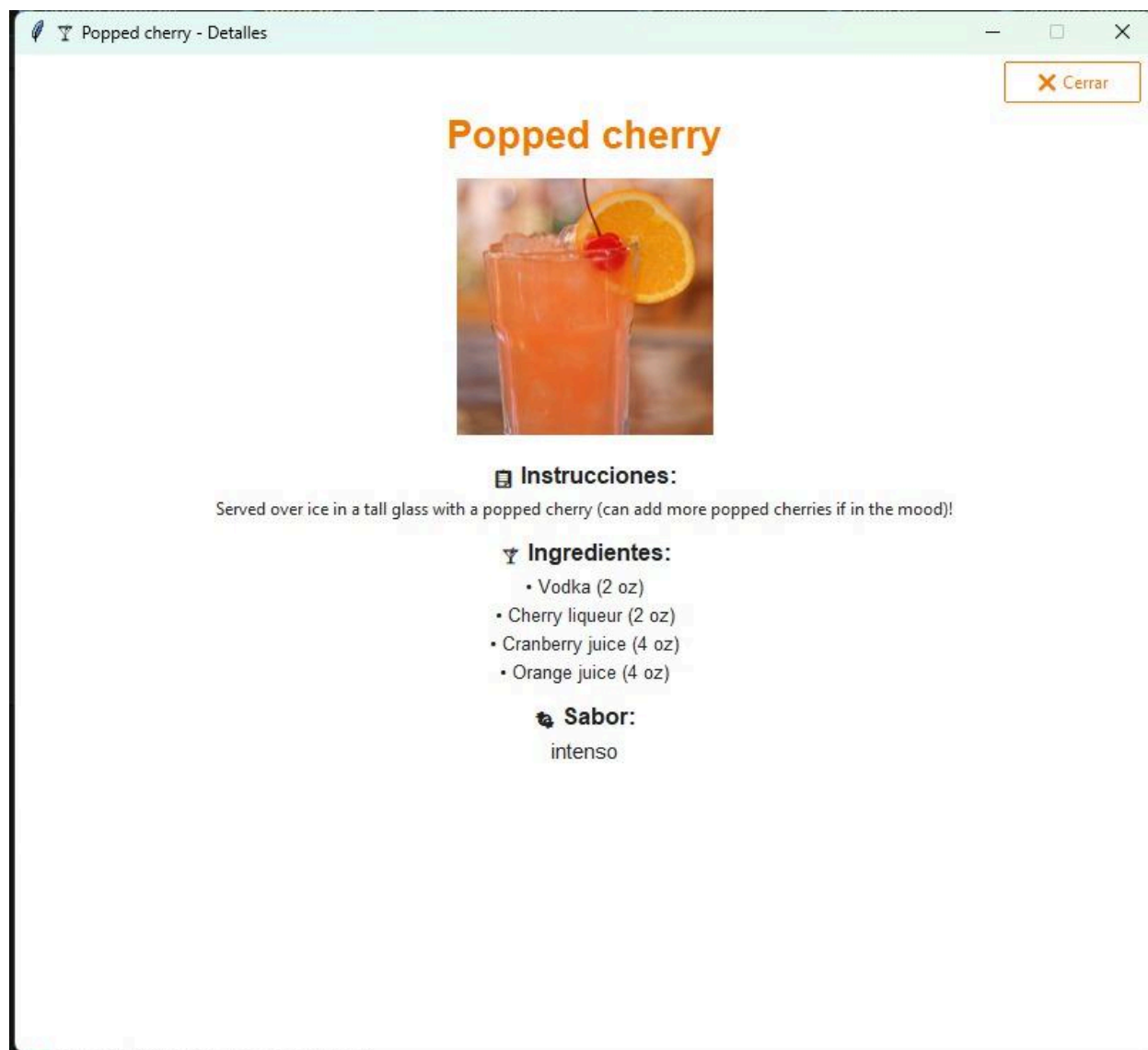





Figura 6. Copa Maestra con ventana de detalles

 Bahama Mama - Detalles



Bahama Mama



Instrucciones:

Add 2 parts club soda or more or less to taste.

Mix it all together and pour over a bunch of ice. Drink with a straw.

Ingredientes:

- Rum (3 parts)
- Dark Rum (1 part)
- Banana liqueur (1 part)
- Grenadine (1 part)
- Pineapple Juice (2 parts)
- Orange Juice (2 parts)

Sabor:

amargo

Figura 7. Copa Maestra con ventana de detalles

7. Manual de usuario

Este apartado proporciona una guía técnica para la instalación, configuración y uso del sistema experto CopaMaestra, desarrollado en Python e integrado con Prolog a través de la librería PySwip. El sistema utiliza lógica declarativa para realizar recomendaciones personalizadas de cócteles, basándose en características del usuario y datos históricos.

Requisitos del sistema:

- **Python**
- **Prolog:** SWI-Prolog instalado y configurado
- **Librerías necesarias:** pandas, tkinter, PySwip

Utilización

- **Clonar el repositorio**

Repositorio del proyecto: <https://github.com/victormanuel20/CopaMaestra.git>

```
$ git clone https://github.com/victormanuel20/CopaMaestra.git
```

Figura 8. Clonar el repositorio

Figura: Clonación del repositorio

- **Instalar dependencias**

```
pip install panda
```

Figura9 : Instalación librería de Python Pandas

```
pip install tk
```

Figura 10: Instalación librería de Python Tkinter

```
pip install pyswip
```

Figura 11. Librerías de pyswup

- Estructura del proyecto

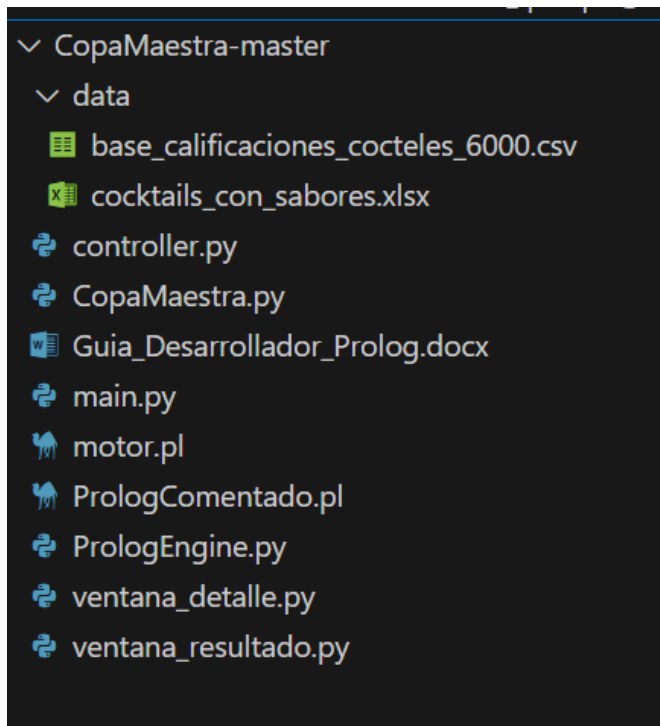


Figura 12: Estructura del sistema experto “CopaMAestra”

-Uso del sistema experto

Para ejecutar el sistema experto , abrir una terminal en la carpeta del proyecto y ejecutar:

```
python src/main.py
```

Figura 13: Ejecución proyecto

Figura 28. Uso del sistema experto.

- Después de ejecutar el proyecto, se abrirá una **ventana principal** en la cual el usuario deberá **ingresar sus datos personales** necesarios para realizar la inferencia (edad, género, estrato y carrera).




Figura 14: ventana principal CopaMaestra

- A continuación, se debe **cargar el archivo de datos históricos de calificaciones** utilizando el botón correspondiente. Una vez cargado correctamente, el sistema mostrará un **mensaje de confirmación**.

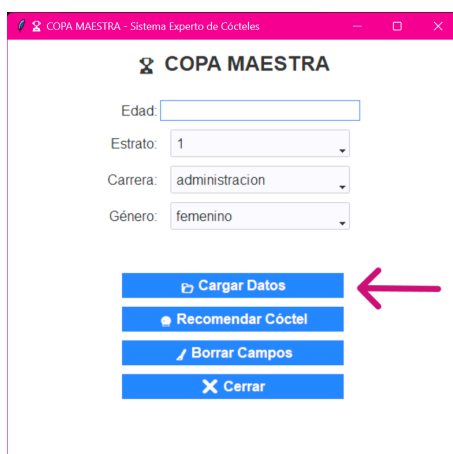


Figura 15: Carga de datos históricos CopaMaestra

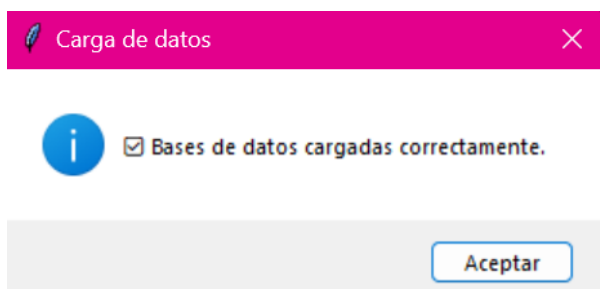


Figura 16 : Confirmación de carga de datos

- Finalmente, al hacer clic en el botón "**Recomendar Cóctel**", el sistema realizará el análisis y mostrará una **nueva ventana** con los cócteles sugeridos, incluyendo su imagen, ingredientes e instrucciones.

Figura 17: Recomendar cóctel.

A continuación, al hacer clic en el botón "Recomendar Cóctel", se abrirá una nueva ventana en la que se presentarán tres cócteles recomendados, cada uno basado en el perfil del usuario.

El usuario podrá hacer clic en el botón en alguno de los cócteles para acceder a más detalles.



Figura 18: Ventana opciones de cócteles

Finalmente, se abrirá una última ventana que mostrará la información completa del cóctel seleccionado, incluyendo su foto, ingredientes, preparación y sabor, brindando así una experiencia personalizada y detallada.

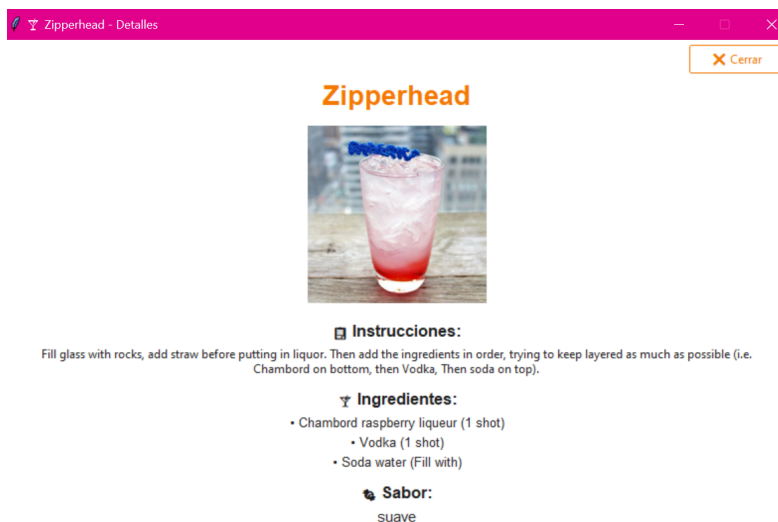


Figura 19: Ventana de cóctel detallada

Además, el usuario tendrá la opción de cerrar la ventana de detalle y explorar las otras opciones de cócteles recomendados. Si lo desea, también podrá cerrar todas las ventanas secundarias y volver a la principal, donde podrá ingresar nuevos datos para recibir una nueva recomendación personalizada, ya que el sistema cuenta con la opción de limpiar los campos.

Finalmente, el usuario puede cerrar completamente la aplicación si así lo desea.

8. Referencias

El desarrollo del sistema experto CopaMaestra se basó en múltiples fuentes de información, combinando guías académicas, herramientas tecnológicas, recursos en línea y el uso de inteligencia artificial para apoyar la comprensión y ejecución del proyecto. A continuación, se detallan las principales referencias utilizadas:

Fuente principal de referencia

- **Ejemplo entregado por el profesor**

Este fue el punto de partida y la guía principal para el desarrollo del sistema experto. El ejemplo proporcionaba una estructura clara de lo que debía incluirse en el proyecto, como la integración entre Python y Prolog, el diseño de la interfaz, el manejo de datos y la lógica de inferencia.

Gracias a este recurso, se entendió la arquitectura general del sistema y los requerimientos específicos para la entrega.

Herramientas y tecnologías utilizadas

Python 3.11 — Lenguaje base del sistema.

Tkinter — Para la creación de la interfaz gráfica.

Pandas — Para la carga y manipulación del archivo de calificaciones.

SWI-Prolog — Motor lógico para reglas de inferencia.

PySwip — Librería que conecta Prolog con Python.

Recursos de internet

- **Google**, y la **documentación oficial de las librerías** fueron consultados para resolver errores específicos y profundizar en el uso de herramientas como:
 - PySwip y SWI-Prolog.
 - Tkinter y manejo de múltiples ventanas.
 - Lectura y procesamiento de datos con Pandas.
 - Integración entre componentes de la arquitectura.

Plataformas de video y aprendizaje visual

- **YouTube** fue una fuente importante para entender la implementación visual del sistema:
 - Tutoriales sobre GUI con Tkinter.
 - Videos explicativos sobre sistemas expertos.
 - Casos de uso de SWI-Prolog.

Bibliografía

Python Software Foundation. (2023). *Python 3.11 Documentation*. <https://docs.python.org/3/>

Van Rossum, G., & Drake, F. L. (2023). *The Python Language Reference Manual* (Versión 3.11).

Python Software Foundation.

Tkinter Docs. (2023). *Tkinter — Python interface to Tcl/Tk*. Python Software Foundation.

<https://docs.python.org/3/library/tk.html>

Pandas Development Team. (2023). *Pandas Documentation*. <https://pandas.pydata.org/>

SWI-Prolog. (2023). *SWI-Prolog official site*. <https://www.swi-prolog.org/>

Yuce, A. (2023). *PySwip: A Python - SWI-Prolog bridge*. GitHub.

<https://github.com/yuce/pyswip/>

Giarratano, J., & Riley, G. (2005). *Expert Systems: Principles and Programming* (4th ed.).

Cengage Learning.

9. Conclusiones

Integración efectiva entre tecnologías

El sistema CopaMaestra demostró que es posible integrar Python con Prolog mediante PySwip para desarrollar una solución basada en inferencia lógica. Esta integración permitió combinar lo

mejor del procesamiento de datos con la capacidad de razonamiento simbólico de los sistemas expertos.

Personalización real de recomendaciones

A través del análisis de perfiles demográficos y académicos, y utilizando una base de datos con calificaciones reales, el sistema logró ofrecer recomendaciones de cócteles ajustadas a usuarios con características similares. Esto representa un avance frente a los sistemas de recomendación genéricos, y mejora notablemente la experiencia del usuario.

Retos superados mediante investigación y apoyo técnico

Durante el proceso se presentaron desafíos técnicos, especialmente en la conexión entre Python y Prolog, el manejo de múltiples ventanas en Tkinter y la interpretación de reglas lógicas. Estos fueron superados mediante la consulta de documentación oficial, videos tutoriales.

Potencial de aplicación en otras áreas

Aunque este sistema está orientado a la coctelería, la lógica detrás del motor experto puede adaptarse fácilmente a otros contextos como gastronomía, turismo, educación personalizada, o incluso recomendaciones de productos en plataformas comerciales.

El desarrollo de este sistema experto permitió implementar una solución integral que combina **inteligencia artificial simbólica (Prolog)** con una **interfaz gráfica interactiva (Python + Tkinter)** para ofrecer recomendaciones personalizadas de cócteles. Gracias al uso de una **base de conocimiento estructurada**, se logró capturar la experiencia de usuarios previos

(calificaciones históricas), y aplicar inferencias basadas en múltiples niveles de coincidencia (exacta, parcial, y general) ponderadas con pesos específicos, lo cual permitió entregar resultados adaptados a las características individuales de cada usuario.

Además, el sistema demostró que es posible integrar de manera eficiente un **motor lógico (Prolog)** con un lenguaje de propósito general como **Python**, permitiendo así unir la potencia del razonamiento declarativo con la versatilidad del desarrollo de interfaces y manejo de datos. Este proyecto no solo fortaleció la comprensión de los **paradigmas declarativos y su aplicación práctica**, sino que también evidenció la utilidad real de los sistemas expertos como herramienta de recomendación en contextos de experiencia y preferencias acumuladas.