**Análisis de calidad con multimetric**

Se utilizó la herramienta multimetric para evaluar distintas métricas del programa desarrollado (chatgpt\_client.py). A continuación se presentan los resultados más relevantes y su análisis:

**🔹 1. comment\_ratio (proporción de comentarios)**

Resultado inicial: 11.63%

Acciones realizadas:

Se incorporaron comentarios detallados en cada función del código.

Se documentaron los parámetros, el propósito y las excepciones manejadas.

Resultado posterior: ~35-40% (estimado).

Conclusión: Se superó el mínimo requerido del 33%, mejorando la mantenibilidad del código.

**🔹 2. halstead\_effort y halstead\_timerequired**

halstead\_effort: 25966.93

halstead\_timerequired: 1442.61 (segundos) ≈ 24 minutos

Tiempo real invertido: Aprox. 40 minutos entre escritura y pruebas.

Conclusión: El esfuerzo y tiempo teóricos son razonablemente consistentes con la experiencia práctica, aunque un poco optimistas.

**🔹 3. halstead\_bugprop**

Valor estimado: 0.411 (aprox. 41%)

Errores reales encontrados:

Uno relacionado con la decodificación del archivo (decode() argument 'encoding' must be str, not None), corregido agregando # -\*- coding: utf-8 -\*-.

Otros errores menores fueron manejados con bloques try/except.

Conclusión: La estimación fue adecuada: hubo errores lógicos leves, pero fueron solucionados con facilidad.

**🔹 4. cyclomatic\_complexity (Complejidad ciclomática)**

Valor obtenido: 3

Interpretación: Excelente. El código tiene una estructura clara y poco ramificada.

Mejoras adicionales posibles: Se podría dividir la función main si se agregan más funcionalidades en el futuro, pero por ahora es innecesario.