

## **Ingeniería de Software II**

### **Trabajo Práctico N°2: Arquitectura - Caso de estudio chatGPT**

**Docentes:**

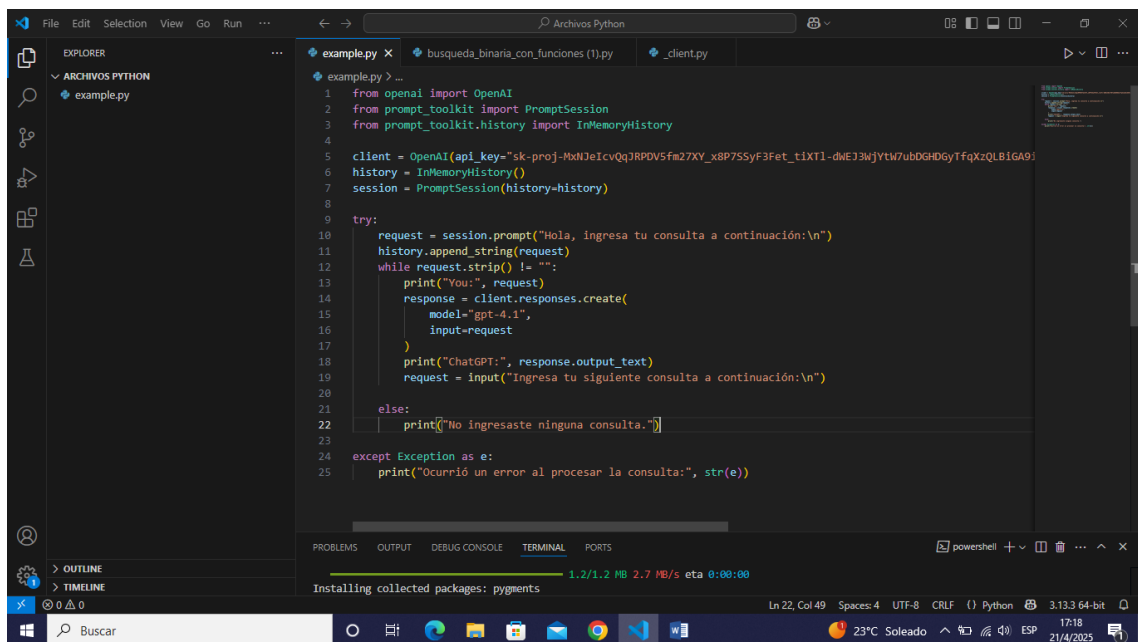
**Dr. Pedro E. Colla**

**Lic. Lucía Blanc**

**Alumno:**

**Mustafá Baissetto**

1 2 3 -

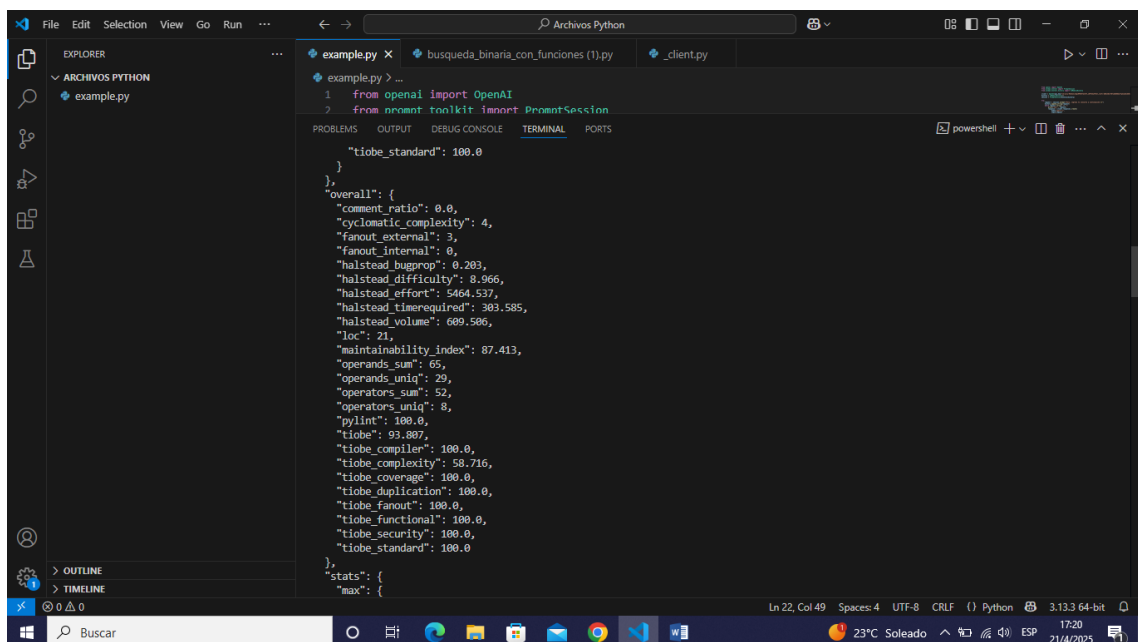


```

1 from openai import OpenAI
2 from prompt_toolkit import PromptSession
3 from prompt_toolkit.history import InMemoryHistory
4
5 client = OpenAI(api_key="sk-proj-MbNJeIcvQqJRPOV5fm27XY_x8P7SSyF3Fet_t1XT1-dmE33WjYtW7ubDGDGyTfqXzQLB1GA9:
6 history = InMemoryHistory()
7 session = PromptSession(history=history)
8
9 try:
10     request = session.prompt("Hola, ingresa tu consulta a continuación:\n")
11     history.append_string(request)
12     while request.strip() != "":
13         print("You:", request)
14         response = client.responses.create(
15             model="gpt-4.1",
16             input=request
17         )
18         print("ChatGPT:", response.output_text)
19         request = input("Ingresa tu siguiente consulta a continuación:\n")
20     else:
21         print("No ingresaste ninguna consulta.")
22 except Exception as e:
23     print("Ocurrió un error al procesar la consulta:", str(e))

```

4 - a)



```

"tiobe_standard": 100.0
},
"overall": {
  "comment_ratio": 0.0,
  "cyclomatic_complexity": 4,
  "fanout_external": 3,
  "fanout_internal": 0,
  "halstead_bugprop": 0.203,
  "halstead_difficulty": 8.966,
  "halstead_effort": 5464.537,
  "halstead_timerequired": 303.585,
  "halstead_volume": 609.506,
  "loc": 21,
  "maintainability_index": 87.413,
  "operands_sum": 65,
  "operands_uniq": 29,
  "operators_sum": 52,
  "operators_uniq": 8,
  "pylint": 100.0,
  "tiobe": 93.887,
  "tiobe_compiler": 100.0,
  "tiobe_complexity": 58.716,
  "tiobe_coverage": 100.0,
  "tiobe_duplication": 100.0,
  "tiobe_fanout": 100.0,
  "tiobe_functional": 100.0,
  "tiobe_security": 100.0,
  "tiobe_standard": 100.0
},
"stats": {
  "max": {

```

El comment ratio está en un valor de 0, ya que en efecto, mi programa carece de comentarios en su código. La manera de aumentar este valor para que sea más afín al idóneo (33%), es agregando comentarios al mismo.

b) Según lo investigado, el halstead effort es la métrica del esfuerzo mental requerido para desarrollar un programa, mientras que el halstead

time required es la métrica que refleja el tiempo que se necesitara para escribir el mismo o mantenerlo.

Se obtuvieron un valor de:

- "halstead\_effort": 5464.537,
- "halstead\_timerequired": 303.585,

c) El valor de halstead bug prop es de 0.203, lo cual concuerda con la cantidad de errores que tuve que solucionar desde que lo ejecute por primera vez.

d) Ya que el valor es menor a 10, no es necesario tomar medidas para reducir este aún más.

e) Los resultados obtenidos de ejecutar Multimetrix sobre este archivo son los siguientes:

PS C:\Users\Usuario\Desktop\Archivos Python> multimetrix example.py

```
{
  "files": {
    "C:\\Users\\Usuario\\Desktop\\Archivos Python\\example.py": {
      "comment_ratio": 0.0,
      "cyclomatic_complexity": 4,
      "fanout_external": 3,
      "fanout_internal": 0,
      "halstead_bugprop": 0.203,
      "halstead_difficulty": 8.966,
      "halstead_effort": 5464.537,
      "halstead_timerequired": 303.585,
      "halstead_volume": 609.506,
      "lang": [
        "Python"
      ],
      "loc": 21,
      "maintainability_index": 87.413,
      "operands_sum": 65,
      "operands_uniq": 29,
      "operators_sum": 52,
      "operators_uniq": 8,
```

```
"pylint": 100.0,  
"tiobe": 93.807,  
"tiobe_compiler": 100.0,  
"tiobe_complexity": 58.716,  
"tiobe_coverage": 100.0,  
"tiobe_duplication": 100.0,  
"tiobe_fanout": 100.0,  
"tiobe_functional": 100.0,  
"tiobe_security": 100.0,  
"tiobe_standard": 100.0  
}  
,  
"overall": {  
  "comment_ratio": 0.0,  
  "cyclomatic_complexity": 4,  
  "fanout_external": 3,  
  "fanout_internal": 0,  
  "halstead_bugprop": 0.203,  
  "halstead_difficulty": 8.966,  
  "halstead_effort": 5464.537,  
  "halstead_timerequired": 303.585,  
  "halstead_volume": 609.506,  
  "loc": 21,  
  "maintainability_index": 87.413,  
  "operands_sum": 65,  
  "operands_uniq": 29,  
  "operators_sum": 52,  
  "operators_uniq": 8,  
  "pylint": 100.0,  
  "tiobe": 93.807,
```

```
"tiobe_compiler": 100.0,  
"tiobe_complexity": 58.716,  
"tiobe_coverage": 100.0,  
"tiobe_duplication": 100.0,  
"tiobe_fanout": 100.0,  
"tiobe_functional": 100.0,  
"tiobe_security": 100.0,  
"tiobe_standard": 100.0  
},  
"stats": {  
  "max": {  
    "comment_ratio": 0.0,  
    "cyclomatic_complexity": 4,  
    "fanout_external": 3,  
    "fanout_internal": 0,  
    "halstead_bugprop": 0.203,  
    "halstead_difficulty": 8.966,  
    "halstead_effort": 5464.537,  
    "halstead_timerequired": 303.585,  
    "halstead_volume": 609.506,  
    "loc": 21,  
    "maintainability_index": 87.413,  
    "operands_sum": 65,  
    "operands_uniq": 29,  
    "operators_sum": 52,  
    "operators_uniq": 8,  
    "pylint": 100.0,  
    "tiobe": 93.807,  
    "tiobe_compiler": 100.0,  
    "tiobe_complexity": 58.716,
```

```
"tiobe_coverage": 100.0,  
"tiobe_duplication": 100.0,  
"tiobe_fanout": 100.0,  
"tiobe_functional": 100.0,  
"tiobe_security": 100.0,  
"tiobe_standard": 100.0  
,  
"mean": {  
  "comment_ratio": 0.0,  
  "cyclomatic_complexity": 4,  
  "fanout_external": 3,  
  "fanout_internal": 0,  
  "halstead_bugprop": 0.203,  
  "halstead_difficulty": 8.966,  
  "halstead_effort": 5464.537,  
  "halstead_timerequired": 303.585,  
  "halstead_volume": 609.506,  
  "loc": 21,  
  "maintainability_index": 87.413,  
  "operands_sum": 65,  
  "operands_uniq": 29,  
  "operators_sum": 52,  
  "operators_uniq": 8,  
  "pylint": 100.0,  
  "tiobe": 93.807,  
  "tiobe_compiler": 100.0,  
  "tiobe_complexity": 58.716,  
  "tiobe_coverage": 100.0,  
  "tiobe_duplication": 100.0,  
  "tiobe_fanout": 100.0,
```

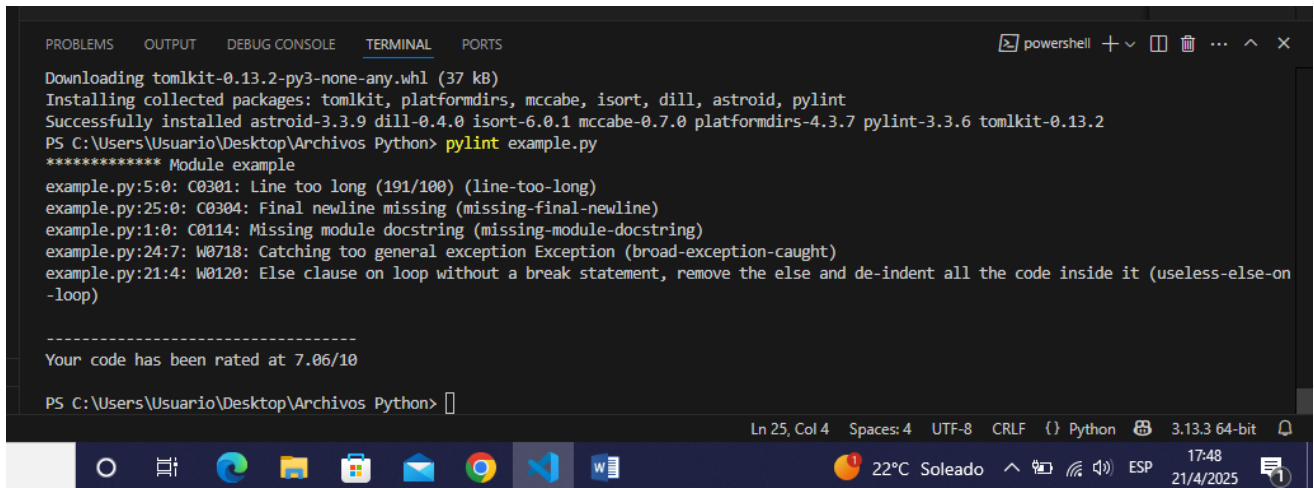
```
"tiobe_functional": 100.0,  
"tiobe_security": 100.0,  
"tiobe_standard": 100.0  
,  
"median": {  
  "comment_ratio": 0.0,  
  "cyclomatic_complexity": 4,  
  "fanout_external": 3,  
  "fanout_internal": 0,  
  "halstead_bugprop": 0.203,  
  "halstead_difficulty": 8.966,  
  "halstead_effort": 5464.537,  
  "halstead_timerequired": 303.585,  
  "halstead_volume": 609.506,  
  "loc": 21,  
  "maintainability_index": 87.413,  
  "operands_sum": 65,  
  "operands_uniq": 29,  
  "operators_sum": 52,  
  "operators_uniq": 8,  
  "pylint": 100.0,  
  "tiobe": 93.807,  
  "tiobe_compiler": 100.0,  
  "tiobe_complexity": 58.716,  
  "tiobe_coverage": 100.0,  
  "tiobe_duplication": 100.0,  
  "tiobe_fanout": 100.0,  
  "tiobe_functional": 100.0,  
  "tiobe_security": 100.0,  
  "tiobe_standard": 100.0
```

```
},  
"min": {  
  "comment_ratio": 0.0,  
  "cyclomatic_complexity": 4,  
  "fanout_external": 3,  
  "fanout_internal": 0,  
  "halstead_bugprop": 0.203,  
  "halstead_difficulty": 8.966,  
  "halstead_effort": 5464.537,  
  "halstead_timerequired": 303.585,  
  "halstead_volume": 609.506,  
  "loc": 21,  
  "maintainability_index": 87.413,  
  "operands_sum": 65,  
  "operands_uniq": 29,  
  "operators_sum": 52,  
  "operators_uniq": 8,  
  "pylint": 100.0,  
  "tiobe": 93.807,  
  "tiobe_compiler": 100.0,  
  "tiobe_complexity": 58.716,  
  "tiobe_coverage": 100.0,  
  "tiobe_duplication": 100.0,  
  "tiobe_fanout": 100.0,  
  "tiobe_functional": 100.0,  
  "tiobe_security": 100.0,  
  "tiobe_standard": 100.0  
}  
}  
}
```



PS C:\Users\Usuario\Desktop\Archivos Python>

5 - b) Primera ejecución del Pylint:

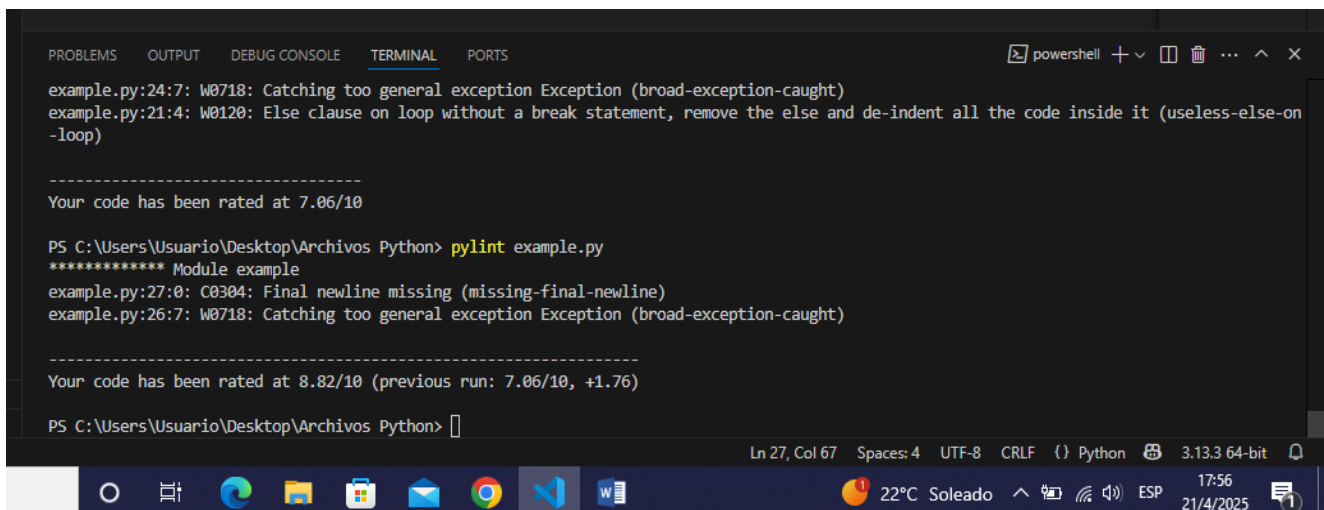


```
Download tomkit-0.13.2-py3-none-any.whl (37 kB)
Installing collected packages: tomkit, platformdirs, mccabe, isort, dill, astroid, pylint
Successfully installed astroid-3.3.9 dill-0.4.0 isort-6.0.1 mccabe-0.7.0 platformdirs-4.3.7 pylint-3.3.6 tomkit-0.13.2
PS C:\Users\Usuario\Desktop\Archivos Python> pylint example.py
***** Module example
example.py:5:0: C0301: Line too long (191/100) (line-too-long)
example.py:25:0: C0304: Final newline missing (missing-final-newline)
example.py:1:0: C0114: Missing module docstring (missing-module-docstring)
example.py:24:7: W0718: Catching too general exception Exception (broad-exception-caught)
example.py:21:4: W0120: Else clause on loop without a break statement, remove the else and de-indent all the code inside it (useless-else-on-loop)

-----
Your code has been rated at 7.06/10

PS C:\Users\Usuario\Desktop\Archivos Python>
```

Segunda ejecución del Pylint:



```
example.py:24:7: W0718: Catching too general exception Exception (broad-exception-caught)
example.py:21:4: W0120: Else clause on loop without a break statement, remove the else and de-indent all the code inside it (useless-else-on-loop)

-----
Your code has been rated at 7.06/10

PS C:\Users\Usuario\Desktop\Archivos Python> pylint example.py
***** Module example
example.py:27:0: C0304: Final newline missing (missing-final-newline)
example.py:26:7: W0718: Catching too general exception Exception (broad-exception-caught)

-----
Your code has been rated at 8.82/10 (previous run: 7.06/10, +1.76)

PS C:\Users\Usuario\Desktop\Archivos Python>
```

c - Los comentarios restantes no fueron corregidos debido a la naturaleza del código.

6 - Las mejoras que sugirió chatGPT son las siguientes:

- Capturar excepciones específicas en lugar de Exception.
- Permitir salida con palabras clave como "salir" o "exit".
- Guardar el historial de conversaciones en un archivo de texto.
- Validar que la entrada no esté vacía antes de enviarla.
- Separar la lógica en funciones reutilizables.

- Verificar que el cliente OpenAI esté correctamente configurado (opcional).
- Agregar separación visual o numeración entre consultas.