Automated Code Review and Enhancement

지원자 주효진

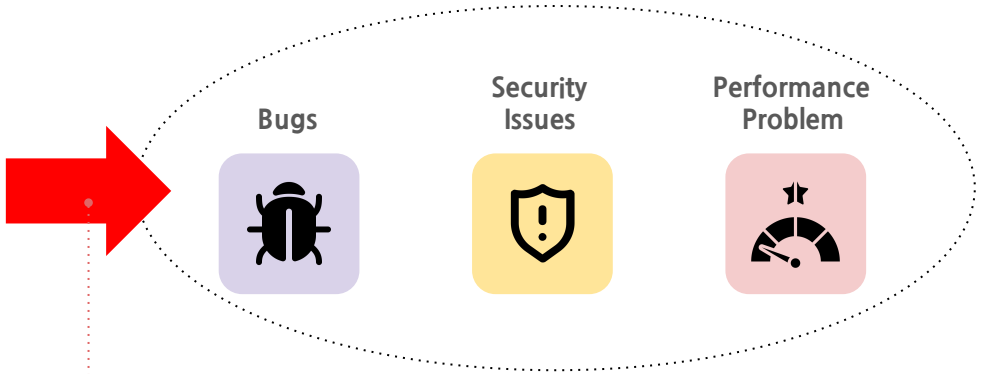
Hyojin.joo | hyojin.joo.ai@gmail.com

2025.08.06.수요일

Background

User Code

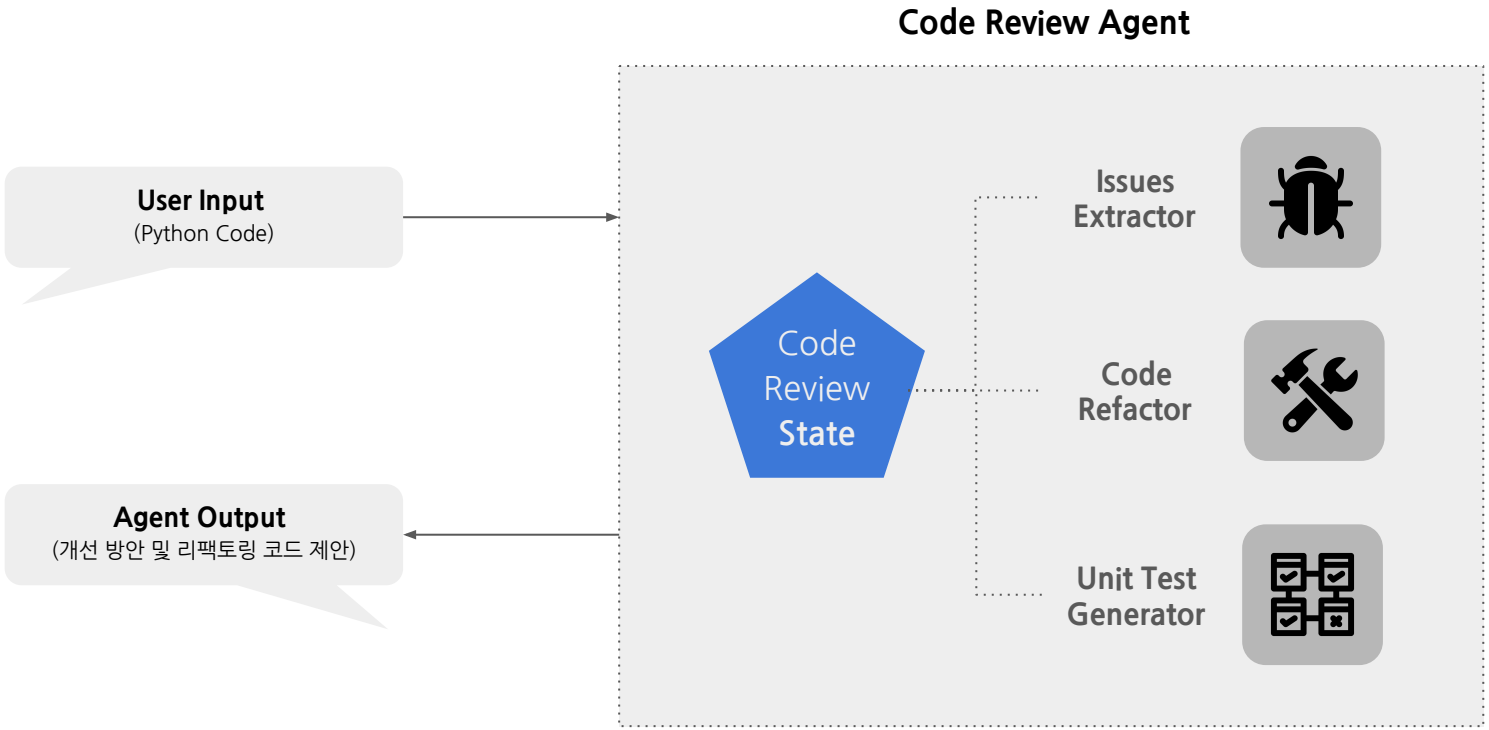
```
def process_user_input(user_data):
    query = f"SELECT * FROM users WHERE id = {user_data['id']}"

    buffer = [0] * 10
    for i in range(len(user_data['items'])):
        buffer[i] = user_data['items'][i]

    return query, buffer
```



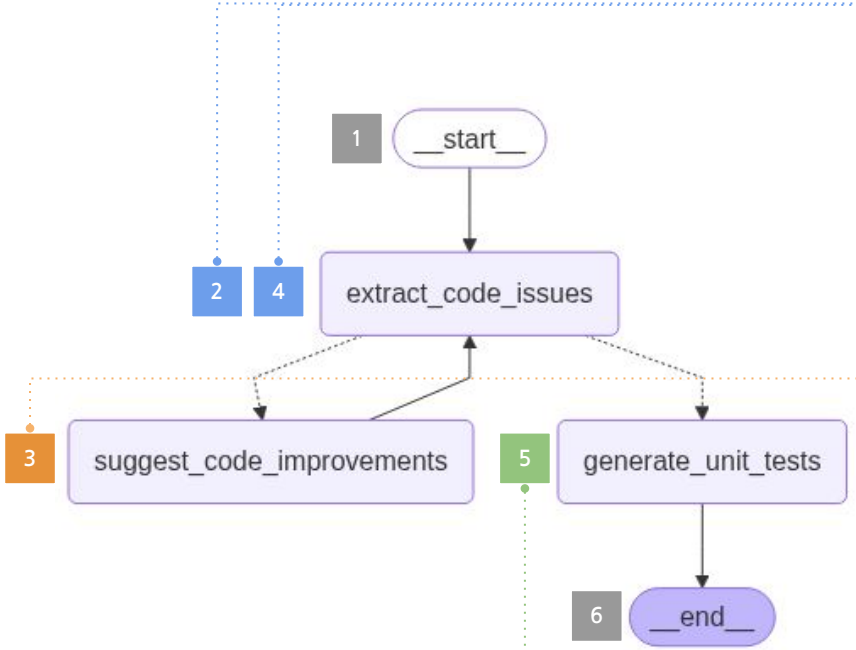
SQL injection vulnerability

Buffer overflow risk

Approach


Code Review Agent
User Input
(Python Code)
Agent Output
(개선 방안 및 리팩토링 코드 제안)
Code
Review
State
Issues
Extractor
Code
Refactor
Unit Test
Generator

Implementation: Agent



Issues Extractor

역할: Bugs, Security Issues, Performance Problem 추출

- **프롬프트 최적화** (Generated Knowledge Prompting) 적용
- **정적 분석 도구 4종 통합** (Pylint, Flake8, Bandit, MyPy 등)
- 데이터 추출 및 활용에 용이하도록 **JSON 형식**으로 결과 반환

Code Refactor

역할: Issues를 개선한 리팩토링 코드 생성

- **이전 단계의 실행 결과 (추출한 이슈 리스트) 전달**하여 코드 생성 시 활용
- 프롬프트에 꼼꼼한 가이드라인 & 전달되는 데이터의 상세 정보 전달
- 코드 생성 시 **주석**을 추가하여 변경한 부분 확인 가능

Unit Test Generator

역할: 코드의 Unit Test 실행을 위한 코드 생성

- LLM에게 **예시를 제공**하여 코드 생성에 도움 (One-Shot Prompting)
- 역할 부여 및 단계별 가이드라인 제공

Implementation: Issue Extractor

Static Analysis (PyLint, Bandit, Flake8, MyPy)

```
== PYLINT REPORT ==
********** Module tmpuorjhhy6
/tmp/tmpuorjhhy6.py:4:0: C0303: Trailing whitespace (trailing-whitespace)
/tmp/tmpuorjhhy6.py:8:0: C0303: Trailing whitespace (trailing-whitespace)
/tmp/tmpuorjhhy6.py:1:0: C0114: Missing module docstring (missing-module-docstring)
/tmp/tmpuorjhhy6.py:2:0: C0116: Missing function or method docstring
(missing-function-docstring)
-----------------------------------
Your code has been rated at 3.33/10

== BANDIT REPORT ==
Run started:2025-08-06 04:25:29.432937

Test results:
>> Issue: [B608:hardcoded_sql_expressions] Possible SQL injection vector through
string-based query construction.
   Severity: Medium   Confidence: Low
   CWE: CWE-89 (https://cwe.mitre.org/data/definitions/89.html)
   More Info:
https://bandit.readthedocs.io/en/1.8.6/plugins/b608_hardcoded_sql_expressions.html
   Location: /tmp/tmpuorjhhy6.py:3:12
1
2           def process_user_input(user_data):
3               query = f"SELECT * FROM users WHERE id = {user_data['id']}"
4
5               buffer = [0] * 10
…
```

Issues List

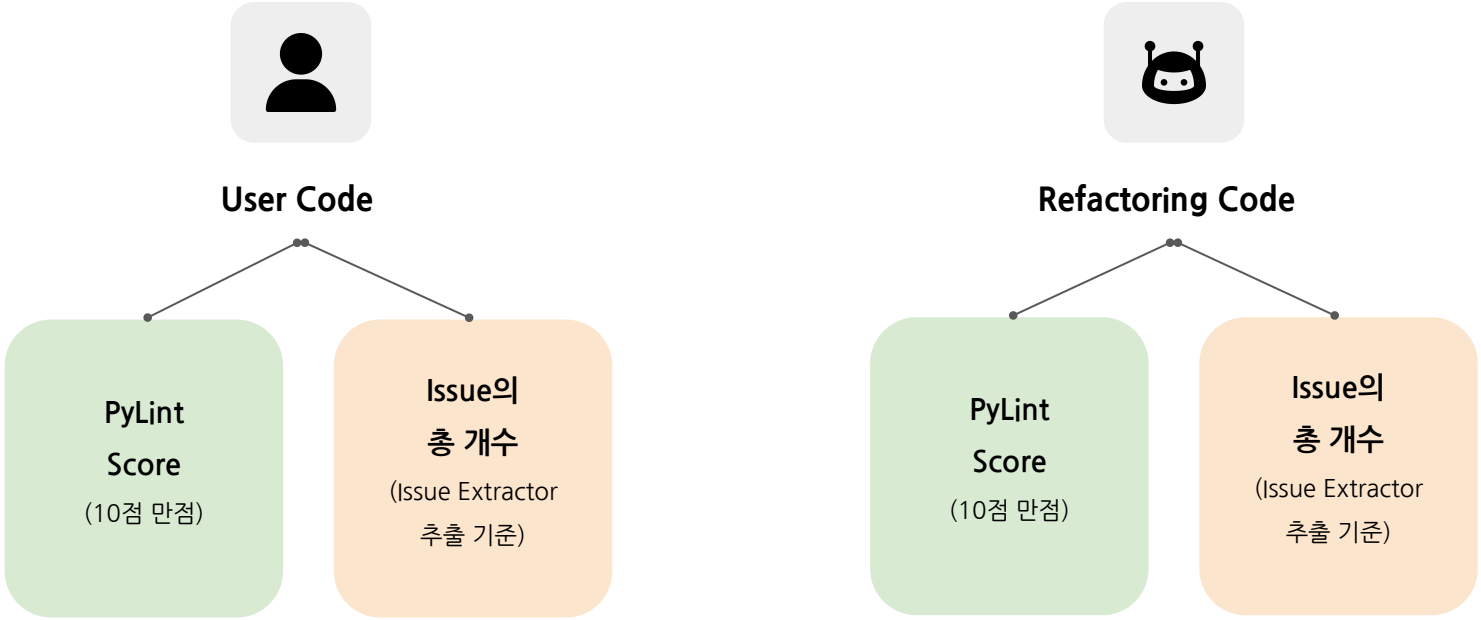
```
{
   "pylint_score": 3.33,
   "issues": [
      {
         "title": "Possible SQL injection",
         "description": "The query is constructed using string interpolation with user
input, which can lead to SQL injection vulnerabilities.",
         "issue_type": "Security Issue",
         "severity": "CRITICAL",
         "start_line": 3,
         "end_line": 3,
         "code_snippet": [
            "query = f\"SELECT * FROM users WHERE id = {user_data['id']}\""
         ]
      },
      …
   ]
}
```

Static Analysis: 정적 도구 분석 결과를 프롬프트에 추가

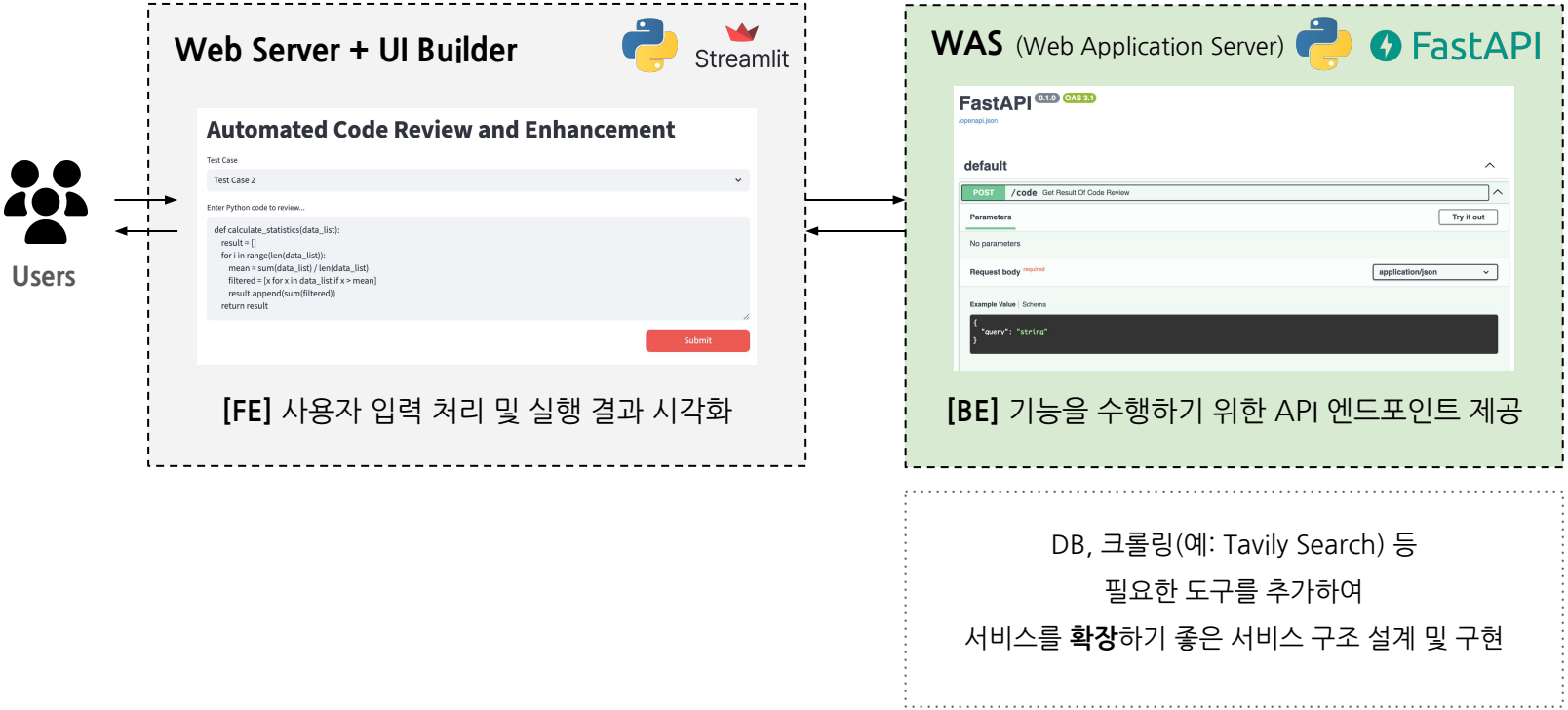
↓

AGENT(LLM): JSON 타입으로 정보 추출

Evaluation: Metrics

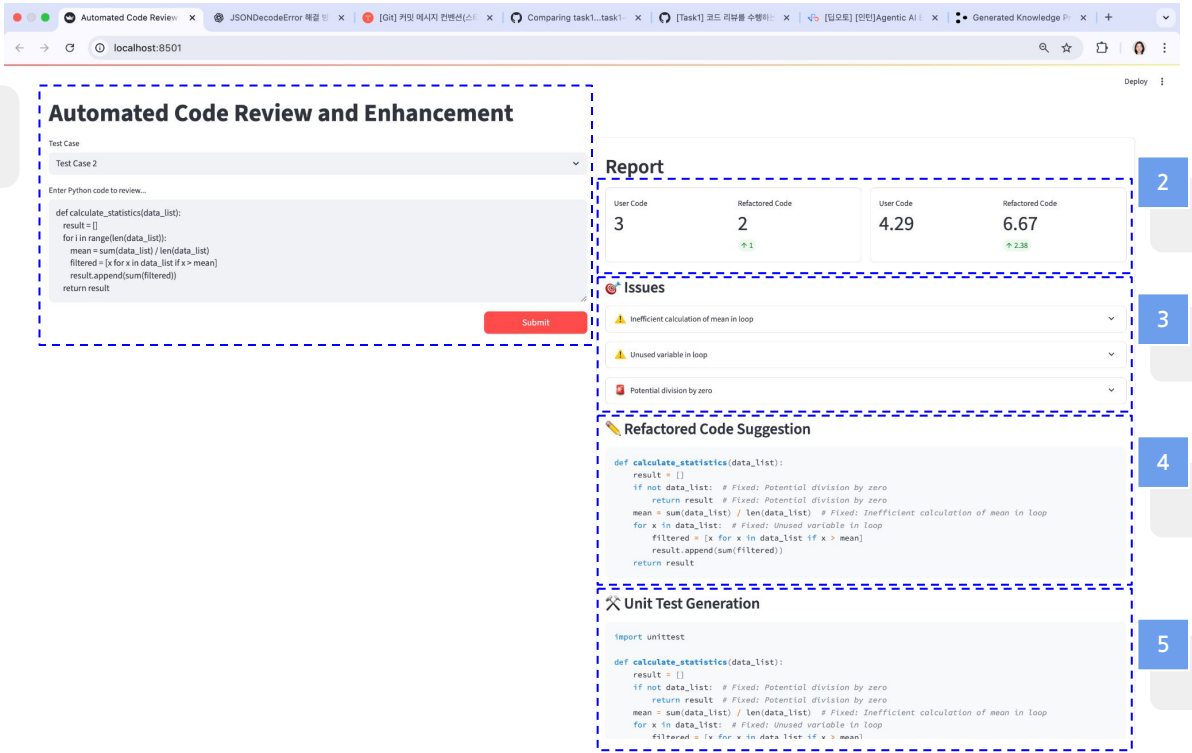


Implementation: Web Architecture



Result

1 사용자 입력부



2 평가 Metrics

3 추출한 이슈 리스트

4 리팩토링 코드

5 단위 테스트 코드