

프로젝트 리뷰 보고서

작성자: DeepSeek API
작성일자: 2025-01-22

1. 리뷰 데이터 요약

PR ID	제목	평균 등급	작성일자
6	Test PR Review 1	B	2025-01-19
12	Revert "[feat/#57] 보고서 제작에 DEEPSEEK API 연동 구현"	B	2025-01-20
20	Revert "[feat/#57] 보고서 제작에 DEEPSEEK API 연동 구현"	B	2025-01-20
26	Revert "[feat/#57] 보고서 제작에 DEEPSEEK API 연동 구현"	A	2025-01-21
27	Revert "[feat/#57] 보고서 제작에 DEEPSEEK API 연동 구현"	A	2025-01-21

2. 분석 내용

****2-1. 리뷰 결과 통계****

- **분석된 PR 수**: 5
- **Clean 모드**: 1개의 리뷰
- **Optimize 모드**: 0개의 리뷰
- **Study 모드**: 0개의 리뷰
- **newbie 모드**: 0개의 리뷰
- **basic 모드**: 0개의 리뷰

****2-2. 주요 취약점 및 개선 우선순위****

****취약한 유형 통계 및 개선 방향**:**

- **취약점 유형 문제점**:
 - 코드 가독성: 변수명이 직관적이지 않거나, 코드 구조가 복잡하여 이해하기 어려움.
 - 중복 코드: 동일한 로직이 여러 곳에서 반복되어 유지보수가 어려움.
 - 예외 처리 부재: 예외 상황에 대한 처리가 미흡하여 프로그램의 안정성이 떨어짐.

- ****개선 방향****:

- ****코드 가독성****: 변수명을 직관적으로 변경하고, 함수를 적절히 분리하여 코드 구조를 단순화.

- ****중복 코드****: 공통 로직을 함수로 추출하여 재사용성을 높임.

- ****예외 처리****: 예외 상황을 고려하여 적절한 예외 처리를 추가.

- ****안좋은 예시와 좋은 예시****:

- ****안좋은 예시****:

```
```python
def calc(a, b):
 return a + b
```
```

- 변수명 `a`, `b`가 무엇을 의미하는지 명확하지 않음.

- ****좋은 예시****:

```
```python
def calculate_sum(num1, num2):
 return num1 + num2
```
```

- 변수명이 직관적이며, 함수명도 명확함.

****2-3. 개인화된 피드백 및 권장사항****

****사용자 맞춤 개선 방향****:

- ****가장 낮은 점수를 받은 평가 기준****: 코드 가독성

- ****개선 방향****:

- 변수명과 함수명을 직관적으로 변경하여 코드의 의도를 명확히 표현.

- 코드 블록을 적절히 분리하고, 주석을 추가하여 코드의 흐름을 쉽게 이해할 수 있도록 함.

- 코드 리뷰를 통해 동료 개발자와의 협업을 통해 가독성을 높이는 방법을 학습.

****2-4. 종합 결론****

- **총평:**

- 프로젝트의 전체적 성향 및 평균 등급: 프로젝트는 기본적인 기능 구현에 초점이 맞춰져 있으나, 코드의 가독성과 유지보수성에서 개선 여지가 큼니다. 특히, 코드 가독성과 중복 코드 부분에서 개선이 필요합니다.

- **강점:**

1. 기본적인 기능 구현이 잘 되어 있습니다.
2. 코드의 논리적 구조가 명확합니다.
3. 프로젝트의 목표를 달성하기 위한 핵심 로직이 잘 구현되어 있습니다.

- **약점:**

1. 코드 가독성이 떨어져 이해하기 어렵습니다.
2. 중복 코드가 많아 유지보수가 어렵습니다.
3. 예외 처리가 미흡하여 프로그램의 안정성이 떨어집니다.

- **향후 권장 사항:**

- ****Clean 모드****를 사용하여 코드의 가독성을 높이는 연습을 권장합니다.
- 코드 리뷰를 통해 동료 개발자와의 협업을 통해 코드 품질을 개선하는 방법을 학습하세요.
- 예외 처리와 같은 안정성 관련 부분에 더 많은 관심을 기울여 프로그램의 신뢰성을 높이세요.

****첨부 자료****

- **추천 학습 자료:**

- [Clean Code by Robert C. Martin](<https://www.amazon.com/Clean-Code-Handbook-Software-Craftsmanship/dp/0132350882>)

- [Refactoring: Improving the Design of Existing Code by Martin Fowler](<https://www.amazon.com/Refactoring-Improving-Design-Existing-Code/dp/0201485672>)

- **관련 예시 코드:**

```
```python
```

```
좋은 예시: 가독성이 높은 코드
```

```
def calculate_total_price(items):
```

```
total_price = 0
for item in items:
 total_price += item.price
return total_price
'''
```

- 이 코드는 변수명과 함수명이 직관적이며, 코드의 의도가 명확하게 드러납니다.