**Họ và Tên: Nguyễn Kim Đăng Khoa  
MSSV: DE180292**

**Lớp: SE18D12**

**Lab1: Use AI prompt:**

1. ***Bạn là một chuyên viên kiểm thử phần mềm có kinh nghiệm hơn 10 năm trong lĩnh vực Testing/IT và có khả năng phân tích, hướng dẫn như một mentor. Dựa vào phần Seven Testing Principles trong chương Fundamental Test Principles của sách Foundations of Software Testing (4th Edition) . Hãy giúp một sinh viên năm 3 phân tích và giải thích câu hỏi sau: Giải thích 7 nguyên tắc cơ bản của kiểm thử phần mềm theo ISTQB. Cho ví dụ minh họa cho nguyên tắc 'Testing shows presence of defects'.***

7 nguyên tắc kiểm thử phần mềm (ISTQB):

1.Testing shows presence of defects: Kiểm thử chỉ phát hiện lỗi, không chứng minh phần mềm không lỗi.

Ví dụ: Tìm thấy 10 lỗi trong hàm tính toán không đảm bảo hàm đó đã hoàn hảo.

2.Exhaustive testing is impossible.

3.Early testing saves time/cost.

4.Defect clustering (80/20 rule).

5.Pesticide paradox.

6.Testing is context-dependent.

7.Absence-of-errors fallacy.

1. **Bạn là một chuyên viên kiểm thử phần mềm có kinh nghiệm hơn 10 năm trong lĩnh vực Testing/IT *và có khả năng phân tích, hướng dẫn như một mentor*. Dựa vào phần *Software development life cycle models trong chương Testing Throughout the Software Lifecycle của sách Foundations of Software Testing (4th Edition)*. Hãy giúp một sinh viên năm 3 phân tích và giải thích câu hỏi sau: Phân biệt giữa Verification và Validation trong kiểm thử phần mềm. Mỗi loại được áp dụng ở test level nào?**

Verification (Xác minh): Đánh giá sản phẩm trong quá trình phát triển ("Are we building the product right?").

Áp dụng ở: Unit Test, Integration Test.

Validation (Xác thực): Đánh giá sản phẩm cuối cùng ("Are we building the right product?").

Áp dụng ở: System Test, Acceptance Test.

1. ***Bạn là một chuyên viên kiểm thử phần mềm có kinh nghiệm hơn 10 năm trong lĩnh vực Testing/IT và có khả năng phân tích, hướng dẫn như một mentor. Dựa vào phần Test Levels trong chương Testing Throughout the Software Lifecycle của sách Foundations of Software Testing (4th Edition) . Hãy giúp một sinh viên năm 3 phân tích và giải thích câu hỏi sau: Mô tả 4 mức độ kiểm thử (Test Levels) trong V-model. Tại sao Component Test lại quan trọng nhất với developer?***

4 Test Levels trong V-model:

1. Component Test (Unit Test): Kiểm thử từng module code.

Quan trọng với developer vì phát hiện lỗi sớm, giảm chi phí sửa.

2. Integration Test.

3. System Test.

4. Acceptance Test.

1. ***Bạn là một chuyên viên kiểm thử phần mềm có kinh nghiệm hơn 10 năm trong lĩnh vực Testing/IT và có khả năng phân tích, hướng dẫn như một mentor. Dựa vào phần Test Levels trong chương Testing Throughout the Software Lifecycle của sách Foundations of Software Testing (4th Edition) . Hãy giúp một sinh viên năm 3 phân tích và giải thích câu hỏi sau: Khi nào nên sử dụng System Test thay vì Acceptance Test? Liệt kê 3 tiêu chí để xác định test coverage ở System Test.***

Dùng System Test khi cần đánh giá toàn hệ thống trước khi giao cho khách hàng.

3 tiêu chí test coverage:

1. Requirement coverage (100% yêu cầu được kiểm tra).

2. Functional coverage (tất cả chức năng chính/phụ).

3. Risk-based coverage (ưu tiên module rủi ro cao).

1. ***Bạn là một chuyên viên kiểm thử phần mềm có kinh nghiệm hơn 10 năm trong lĩnh vực Testing/IT và có khả năng phân tích, hướng dẫn như một mentor. Dựa vào hai chương Fundamental Test Principles và Testing Throughout the Software Lifecycle của sách Foundations of Software Testing (4th Edition). Hãy giúp một sinh viên năm 3 phân tích và giải thích câu hỏi sau: Theo ISTQB, 'Early Testing' nên được áp dụng thế nào trong các test level? Cho ví dụ cách áp dụng nguyên tắc này ở Integration Test***

Áp dụng Early Testing bằng cách bắt đầu kiểm thử ngay từ giai đoạn yêu cầu (requirement analysis).

Ví dụ ở Integration Test:

Thiết kế test cases ngay khi xác định giao diện giữa các module.

Phát hiện lỗi giao tiếp API trước khi triển khai toàn hệ thống.