Bài thi:  
SV: Trần Thị Mai Hiên

Câu 2:  
Trong một dự án phát triển phần mềm sử dụng Git, người quản lý kỹ thuật có thể sử dụng nhiều nhánh để tạo điều kiện cho quá trình phát triển. Dưới đây là một số nhánh phổ biến và mục đích của chúng:

* **Nhánh Main (Chính):** Nhánh này đại diện cho phiên bản ổn định của phần mềm. Chỉ nên chứa mã nguồn đã được kiểm tra kỹ lưỡng và được chấp nhận để phát hành.
* **Nhánh Development (Phát triển):** Nhánh này là nơi diễn ra công việc phát triển liên tục. Các nhà phát triển đẩy các nhánh tính năng của họ vào đây để tích hợp và kiểm tra.
* **Các Nhánh Tính Năng:** Mỗi tính năng hoặc mô-đun mới được phát triển thường có một nhánh tính năng riêng. Các nhánh này cho phép các nhà phát triển làm việc trên các tính năng cụ thể mà không ảnh hưởng đến nhánh phát triển chính. Khi tính năng hoàn thành và kiểm tra, nó có thể được hợp nhất vào nhánh phát triển.
* **Nhánh Phát Hành:** Khi đến lúc chuẩn bị một bản phát hành mới, có thể tạo ra một nhánh phát hành từ nhánh phát triển. Nhánh này được sử dụng để kiểm tra cuối cùng và sửa lỗi trước khi bản phát hành được thực hiện. Khi bản phát hành sẵn sàng, nó có thể được hợp nhất vào nhánh chính.
* **Nhánh Sửa Lỗi Nhanh (Hotfix):** Trong trường hợp phát hiện ra lỗi hoặc vấn đề nghiêm trọng trong nhánh chính cần được chú ý ngay lập tức, có thể tạo ra nhánh sửa lỗi nhanh từ nhánh chính. Các nhánh này cho phép sửa lỗi nhanh chóng và được hợp nhất vào cả nhánh chính và nhánh phát triển.

Câu 3:  
**- What are types of testing strategies?**

1. **Black box testing** – Tests the functionality of the software without looking at the internal code structure.
2. **White box testing** – Tests the internal code structure and logic of the software.
3. **Unit testing**– Tests individual units or components of the software to ensure they are functioning as intended.
4. **Integration testing**– Tests the integration of different components of the software to ensure they work together as a system.
5. **Functional testing** – Tests the functional requirements of the software to ensure they are met.
6. **System testing**– Tests the complete software system to ensure it meets the specified requirements.
7. **Acceptance testing** – Tests the software to ensure it meets the customer’s or end-user’s expectations.
8. **Regression testing** – Tests the software after changes or modifications have been made to ensure the changes have not introduced new defects.
9. **Performance testing**– Tests the software to determine its performance characteristics such as speed, scalability, and stability.
10. **Security testing**– Tests the software to identify vulnerabilities and ensure it meets security requirements.

**-What is Unit Testing used for?**

Unit Testing được sử dụng để kiểm tra từng phần riêng lẻ của mã nguồn, như hàm hoặc phương thức, để đảm bảo tính đúng đắn và chức năng của từng phần một cách cô đọng. Trong kiểm thử đơn vị, các đoạn mã được kiểm thử độc lập từng phần mà không phụ thuộc vào các phần khác của hệ thống. Điều này giúp xác định lỗi và sửa chúng ở mức độ cụ thể nhất, tăng tính tin cậy và sự ổn định của mã nguồn.

**- What is Integration Testing used for?**

Kiểm thử tích hợp (Integration Testing) được sử dụng để kiểm tra các thành phần đã được hợp nhất lại với nhau trong hệ thống để đảm bảo tính đúng đắn của giao tiếp và tương tác giữa chúng. Mục tiêu của kiểm thử tích hợp là xác định các lỗi xuất phát từ việc tích hợp và giao tiếp giữa các thành phần của hệ thống. Bằng cách này, kiểm thử tích hợp đảm bảo rằng các thành phần hoạt động một cách hợp lý khi được kết hợp với nhau và không gây ra các vấn đề không mong muốn trong quá trình tích hợp.

**- If a tester has a new module to test, which steps does he/she should test from starting the module to releasing the module?**

Khi một người kiểm thử có một mô-đun mới để kiểm thử, các bước nên thực hiện từ khi bắt đầu kiểm thử mô-đun đó đến khi phát hành mô-đun là:

1. **Phân Tích Yêu Cầu Kiểm Thử:**
   * Hiểu rõ yêu cầu về chức năng và phi chức năng của mô-đun cần kiểm thử.
2. **Lập Kế Hoạch Kiểm Thử:**
   * Xác định phạm vi, kịch bản kiểm thử và tài nguyên cần thiết cho quá trình kiểm thử.
3. **Xây Dựng Môi Trường Kiểm Thử:**
   * Chuẩn bị môi trường kiểm thử phù hợp với mô-đun để thực hiện các ca kiểm thử.
4. **Thiết Kế Bài Kiểm Thử:**
   * Tạo các ca kiểm thử dựa trên kịch bản kiểm thử đã xác định.
5. **Thực Hiện Kiểm Thử:**
   * Thực hiện các ca kiểm thử trên mô-đun và ghi lại kết quả.
6. **Sửa Lỗi và Kiểm Tra Lại:**
   * Nếu phát hiện lỗi, ghi lại và báo cáo lỗi, sau đó sửa chúng và thực hiện kiểm tra lại để đảm bảo tính ổn định của mô-đun.
7. **Kiểm Thử Tích Hợp:**
   * Nếu mô-đun liên quan đến các thành phần khác, thực hiện kiểm thử tích hợp để đảm bảo tính tương thích và tương tác đúng đắn.
8. **Kiểm Thử Hệ Thống và Chấp Nhận:**
   * Thực hiện kiểm thử hệ thống và chấp nhận cuối cùng để đảm bảo rằng mô-đun hoạt động như mong đợi và đáp ứng được yêu cầu của người dùng.

Bằng cách này, người kiểm thử có thể đảm bảo rằng mô-đun mới đã được kiểm thử kỹ lưỡng và sẵn sàng để được triển khai và sử dụng trong sản phẩm.