**CHƯƠNG 1: TỔNG QUAN VỀ LÝ THUYẾT**

1. **Soflware Testing Overview:**
2. **Software Testing là gì?**

* Kiểm thử phần mềm là một quá trình thực thi một chương trình hoặc ứng dụng với mục đích tìm ra lỗi phần mềm. Và được sử dụng để xác định tính đúng đắn, đầy đủ và chất lượng của phần mềm máy tính đã phát triển.
* Kiểm thử phần mềm sẽ được test theo 2 hướng:
  + Positive
  + Negative

1. **Tại sao Software Testing lại cần thiết?**

* Kiểm thử phần mềm thực sự được yêu cầu để chỉ ra các khiếm khuyết và lỗi đã được thực hiện trong các giai đoạn phát triển.
* Điều cần thiết là vì nó đảm bảo rằng khách hàng thấy tổ chức đáng tin cậy và sự hài lòng của họ trong ứng dụng được duy trì

1. **7 nguyên tắc của Testing:**

* Testing shows presence of defects
* Exhaustive testing is impossible
* Early testing
* Defect clustering
* Pesticide paradox
* Testing is context dependent
* Absence – of – errors fallacy

1. **STLC là gì?**

* Vòng đời kiểm thử phần mềm (STLC) là quá trình kiểm thử được thực hiện theo cách có hệ thống và có kế hoạch. Trong quy trình STLC, các hoạt động khác nhau được thực hiện để nâng cao chất lượng của sản phẩm. Mỗi bước đều có Tiêu chí đầu vào riêng và có thể phân phối.
  + Requirement Analysis
  + Test Planning
  + Test Case Development
  + Environment Setup
  + Test Execution
  + Test Cycle Closure

1. **Bug, Fault & Failure là gì?**

* Bug: Sự hiện diện của lỗi tại thời điểm thực thi phần mềm
* Fault: Trạng thái của phần mềm do lỗi
* Failure: Sự sai lệch của phần mềm so với kết quả mong đợi của nó.

1. **Phân biệt giữa Verification và Validation:**

|  |  |
| --- | --- |
| Verification | Validation |
| Xác minh là quá trình xác nhận rằng phần mềm đáp ứng đặc điểm kỹ thuật của nó, được thực hiện thông qua kiểm tra và hướng dẫn. | Xác thực là quá trình xác nhận rằng nó đáp ứng các yêu cầu của người dùng. Đó là thử nghiệm thực tế. |
| Sản phẩm có đúng không? | Nó có phải là sản phẩm phù hợp |

1. **Phân biệt QA vs QC:**

|  |  |
| --- | --- |
| **Quality Assurance (QA)** | **Quality Control (**QC) |
| Đảm bảo chất lượng là một tập hợp các hoạt động có kế hoạch và có hệ thống cần thiết để cung cấp sự tin tưởng đầy đủ rằng các sản phẩm và dịch vụ sẽ phù hợp với các yêu cầu cụ thể và đáp ứng nhu cầu của người sử dụng. | Kiểm soát chất lượng là quá trình mà chất lượng sản phẩm được so sánh với các tiêu chuẩn áp dụng và hành động được thực hiện khi phát hiện thấy sự không phù hợp |

1. **Software Development Life Cycle (SDLC)**
2. **Định nghĩa:**

* SDLC, Vòng đời phát triển phần mềm là một quá trình được ngành công nghiệp phần mềm sử dụng để thiết kế, phát triển và kiểm tra phần mềm chất lượng cao. Mục tiêu của SDLC là tạo ra một phần mềm chất lượng cao đáp ứng hoặc vượt quá mong đợi của khách hàng, hoàn thành trong thời gian và chi phí ước tính.
* SDLC có 6 giai đoạn:
  + Requirement gathering and analysis
  + Design
  + Implementation or coding
  + Testing
  + Deployment
  + Maintenance

1. Các mô hình phổ biến của SDLC:

* Waterfall Model
* V Model
* Agile Model

1. Phương pháp Scrum
2. Phương pháp Kanban
3. **Software Testing – Types & Methods:**
4. **Software Testing – Types**:

* **Manual Test**:
  + Kiểm thử thủ công bao gồm kiểm tra phần mềm theo cách thủ công, tức là không sử dụng bất kỳ công cụ tự động nào hoặc bất kỳ tập lệnh nào. Trong loại này, người kiểm tra đảm nhận vai trò của người dùng cuối và kiểm tra phần mềm để xác định bất kỳ hành vi hoặc lỗi không mong muốn nào. Có các giai đoạn khác nhau để kiểm tra thủ công như kiểm thử đơn vị, kiểm thử tích hợp, kiểm tra hệ thống và kiểm thử chấp nhận người dùng
* **Automation Testing:**
  + Kiểm thử tự động, còn được gọi là Test Automation, là khi người kiểm thử viết các kịch bản và sử dụng một phần mềm khác để kiểm tra sản phẩm. Quá trình này liên quan đến việc tự động hóa quy trình thủ công. Kiểm thử tự động được sử dụng để chạy lại các kịch bản kiểm tra đã được thực hiện theo cách thủ công, nhanh chóng và lặp đi lặp lại.
  + Ngoài kiểm thử hồi quy, kiểm thử tự động hóa cũng được sử dụng để kiểm tra ứng dụng từ quan điểm tải, hiệu suất và căng thẳng. Nó làm tăng phạm vi kiểm tra, cải thiện độ chính xác và tiết kiệm thời gian và tiền bạc so với kiểm tra thủ công.

1. **Software Testing – Method:**

* **Black – box Testing:**
  + Kỹ thuật kiểm tra mà không có bất kỳ kiến thức nào về hoạt động bên trong của ứng dụng được gọi là kiểm thử hộp đen
  + Người kiểm tra không biết gì về kiến trúc hệ thống và không có quyền truy cập vào mã nguồn
* **White – box Testing:**
  + Kiểm thử hộp trắng là việc điều tra chi tiết về logic và cấu trúc bên trong của mã. Thử nghiệm hộp trắng còn được gọi là thử nghiệm kính hoặc thử nghiệm hộp mở.
  + Để thực hiện kiểm tra hộp trắng trên một ứng dụng, người kiểm tra cần biết các hoạt động bên trong của mã.

1. **Software Testing Level:**
2. **Unit test:**

* Đơn vị là phần nhỏ nhất có thể kiểm tra được của một ứng dụng như hàm, lớp, thủ tục, giao diện. Kiểm thử đơn vị là một phương pháp mà các đơn vị mã nguồn riêng lẻ được kiểm tra để xác định xem chúng có phù hợp để sử dụng hay không.
* Các bài kiểm tra đơn vị về cơ bản được viết và thực hiện bởi các nhà phát triển phần mềm để đảm bảo rằng mã đáp ứng thiết kế và yêu cầu của nó và hoạt động như mong đợi
* Unit Test được thực hiện bằng cách sử dụng phương pháp Kiểm thử Hộp Trắng

1. **Integration test:**

* Kiểm thử tích hợp kiểm tra sự tích hợp hoặc giao diện giữa các thành phần, tương tác với các phần khác nhau của hệ thống như hệ điều hành, hệ thống tệp và phần cứng hoặc giao diện giữa các hệ thống
* Các hình thức test:
  + Bottom up:
  + Top down
  + Hybrid

1. **System Test:**

* Bao gồm các thử nghiệm dựa trên rủi ro và / hoặc đặc tả yêu cầu, quy trình nghiệp vụ, trường hợp sử dụng hoặc các mô tả cấp cao khác về hành vi của hệ thống, tương tác với hệ điều hành và tài nguyên hệ thống.
* Kiểm tra hệ thống thường là kiểm tra cuối cùng để xác minh rằng hệ thống được cung cấp đáp ứng các đặc điểm kỹ thuật và mục đích của nó..
* Kiểm thử hệ thống theo kiểm thử chức năng và phi chức năng:
  + Kiểm thử chức năng:
    - Requirement-based testing: Thử nghiệm dựa trên yêu cầu
    - Business-process-based testing: Thử nghiệm dựa trên quy trình kinh doanh
  + Kiểm thử phi chức năng:
    - Sanity testing
    - Re-test
    - Regression Test
    - Security Test
    - Recovery Test
    - Usability Test
    - Performance Test
    - Load Test
    - Stress test:

1. **User Acceptance Test:**

* Sau khi kiểm tra hệ thống đã sửa chữa tất cả hoặc hầu hết các khiếm khuyết, hệ thống sẽ được giao cho người dùng hoặc khách hàng để kiểm tra chấp nhận.
* Kiểm tra chấp nhận về cơ bản được thực hiện bởi người dùng hoặc khách hàng mặc dù các bên liên quan khác cũng có thể tham gia.
* Có 2 loại:
  + Alpha test
  + Beta test

1. **Test Case:**
2. **Khái niệm:**

* Trường hợp thử nghiệm là một tài liệu, trong đó có một tập hợp dữ liệu thử nghiệm, điều kiện tiên quyết, kết quả mong đợi và điều kiện hậu kỳ, được phát triển cho một kịch bản thử nghiệm cụ thể để xác minh sự tuân thủ theo một yêu cầu cụ thể.
* Test Case đóng vai trò là điểm bắt đầu cho quá trình thực thi thử nghiệm và sau khi áp dụng một tập hợp các giá trị đầu vào, ứng dụng sẽ có kết quả cuối cùng và rời khỏi hệ thống tại một số điểm kết thúc hoặc còn được gọi là điều kiện hậu thực thi.

1. **Các loại kỹ thuật Testcase:**
   1. **Kỹ thuật kiểm thử tĩnh:**

* Thử nghiệm tĩnh là loại kiểm tra trong đó code không được thực hiện. Nó có thể được thực hiện bằng tay hoặc bằng một bộ công cụ
* Các kỹ thuật:
  + Informal review
  + Technical Reviews
  + Walkthrough
  + Inspection
  + Static code Review
  1. **Kỹ thuật kiểm thử động:**
* Thực hiện khi code đang ở chế độ thực thi.. Khi code được thực thi, sẽ truyền 1 giá trị vào đầu vào, sau đó xem kết quả hoặc đầu ra để so sánh với kết quả dự kiến ban đầu, kiểm thử động được chia làm 3 loại:
* Kỹ thuật dựa trên đặc điểm kỹ thuật:
  + Boundary Value Analysis (BVA) Phân tích giá trị biên
  + Equivalence Partitioning (EP) Phân vùng tương đương (EP)
  + Decision Table Testing Kiểm tra Bảng Quyết định
  + State Transition Diagrams Sơ đồ chuyển đổi trạng thái
  + Use Case Testing Kiểm tra trường hợp sử dụng
* Kỹ thuật dựa trên kinh nghiệm:
  + Error Guessing (Đoán lỗi): Đoán lỗi là một kỹ thuật kiểm tra sử dụng kỹ năng, trực giác và kinh nghiệm của người kiểm tra trong việc kiểm tra các ứng dụng tương tự để xác định các lỗi có thể không dễ nắm bắt bằng các kỹ thuật chính thức hơn
  + Exploratory Testing (Thử nghiệm thăm dò): Kỹ thuật kiểm tra thăm dò được áp dụng kết hợp kinh nghiệm của người kiểm tra cùng với cách tiếp cận có cấu trúc để kiểm tra, thường được thực hiện như một kỹ thuật kiểm thử hộp đen, người kiểm tra học hỏi những điều cùng với kinh nghiệm và sự sáng tạo tạo ra các thử nghiệm tốt mới để chạy.