

BỘ GIÁO DỤC VÀ ĐÀO TẠO

TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT THÀNH PHỐ HỒ CHI MINH

KHOA CÔNG NGHỆ THÔNG TIN

****

Báo Cáo Cuối Kì

KIỂM THỬ PHẦN MỀM

GVHD: Trương Thị Ngọc Phượng

SVTH:

* Phan Hữu Hiếu 16110329
* Phan Thị Quang Thư 16110480

Thành phố Hồ Chí Minh, tháng 12 năm 2019

Mục lục

[**1.** **CHƯƠNG 1: FUNDAMENTAL TEST PROCESS** 1](#_Toc26803310)

[1.1. Testing planning and control 1](#_Toc26803311)

[1.2. Testing analysis and design 1](#_Toc26803312)

[1.3. Testing implementation and execution 2](#_Toc26803313)

[1.4. Evaluating exit criteria and reporting 2](#_Toc26803314)

[1.5. Test closure activities 2](#_Toc26803315)

[**2.** **CHƯƠNG 2: SOFTWARE LIFE CYCLE BASED TESTING** 3](#_Toc26803316)

[2.1. Testing in software life cycle 3](#_Toc26803317)

[2.1.1. Testing trong quy trình Waterfall 3](#_Toc26803318)

[2.1.2. Testing trong những quy trình có vòng lặp 4](#_Toc26803319)

[2.1.3. Testing trong quy trình Agile 4](#_Toc26803320)

[2.2. Testing level 5](#_Toc26803321)

[2.2.1. Conponent testing 5](#_Toc26803322)

[2.2.2. Integration testing 6](#_Toc26803323)

[2.2.3. System testing 6](#_Toc26803324)

[2.2.4. Acceptance testing 7](#_Toc26803325)

[**3.** **CHƯƠNG 3: UNIT TESTING** 7](#_Toc26803326)

[3.1. Boundary value testing 7](#_Toc26803327)

[3.2. Equivalence class testing 7](#_Toc26803328)

[3.3. Decision table 7](#_Toc26803329)

[3.4. Path test 8](#_Toc26803330)

[3.5. Data flow testing 8](#_Toc26803331)

[**4.** **CHƯƠNG 4: INTEGRATION SYSTEM TESTING** 8](#_Toc26803332)

[4.1. Decomposition 8](#_Toc26803333)

[4.2. Graph 8](#_Toc26803334)

[4.3. path 8](#_Toc26803335)

[**5.** **CHƯƠNG 5: UI TESTING** 8](#_Toc26803336)

[**6.** **CHƯƠNG 6: VALIDATION TESTING** 8](#_Toc26803337)

1. **CHƯƠNG 1: FUNDAMENTAL TEST PROCESS**
   1. Testing planning and control

* Test plan chính là tài liệu tổng quan về việc kiểm thử một project đặc tả:
  + Phạm vi dự án
  + Hướng tiếp cận
  + Quy trình kiểm thử
  + Tài nguyên và nhân lực hiện có
* Là cơ sở để kiểm tra các sản phẩm, phần mềm trong một dự án
* Kiểm soát kiểm tra là giám sát các hoạt động và so sánh những gì thực sự xảy ra trong kế hoạch của dự án
* Một phần của kiểm soát kiểm tra là quản trị và duy trì quá trình thứ nghiệm, cơ sở hạ tầng test ->testware
* Các hoạt động kiểm tra phải được phân chia cho các hệ thống con riêng lẻ, tùy thuộc vào rủi ro dự kiến và mức độ nghiêm trọng của các hiệu ứng thất bại.
* Các hệ thống con quan trọng phải được chú ý nhiều hơn, do đó được kiểm tra chuyên sâu hơn
* Quy trình giám sát và kiểm soát cơ bản
* Tạo kế hoạch giám sát
* Cập nhật tiến độ
* Phân tích tiến độ và thực hiện điều chỉnh
* Tạo báo cáo
  1. Testing analysis and design
* Test analysis hoặc test designer sẽ thiết kế các testcase từ các yêu cầu liên quan
* Thiết kế các test case từ các yêu cầu chức năng và yêu cầu phi chức năng của phần mềm
* Các testcase cần bao phủ tất cả yêu cầu trong chiến lược kiểm thử
* Nếu cần kiểm thử tự động, test designer sẽ xây dựng các kịch bản dựa trên testcase/ test procedures
  1. Testing implementation and execution
* Test implementation là hoạt động trong đó các kiểm thử được tổ chức và được ưu tiên bởi các nhà phân tích kiểm thử
* Gọi ra việc kiểm tra cuối cùng để đảm bảo cho đội kiểm tra sẵn sàng thực hiện test
* Tester sẽ được bố trí công việc bởi test leader để thi hành kiểm thử
* Thi hành kiểm thử theo từng testcase
* Thực hiện kiểm thử đặc biệt
* Thực hiện kịch bản kiểm thử mà không được định nghĩa trong testcase
* Kiểm thử lại các lỗi đã được chỉnh sửa
* Tester sẽ tạo các báo cáo vè lỗi trong suốt quá trình kiểm thử lỗi và theo dõi chúng cho đến khi chúng đã được xử lý
* Ở công đoạn kiểm thử chấp thuận, khách hàng sẽ thi hành kiểm thử để kiểm định xem hệ thống phần mềm có thỏa mãn các nhu cầu người dùng không?
  1. Evaluating exit criteria and reporting
* Test manager hoặc test leader sẽ phân tích các lỗi trong hệ thống theo dõi các lỗi
* Tạo các báo cáo lỗi
* Đánh giá các kết quả kiểm thử, thống kê các yêu cầu thay đổi
* Tính và phân phối các thông tin đo lường hoạt động kiểm thử
* Tạo bảng tổng kết đánh giá hoạt động kiểm lỗi
* Xác định xem đã đạt chỉ tiêu thành công va hoàn thành kiểm thử chưa
  1. Test closure activities
* Một khi thực hiện kiểm thử được hoàn thành, các đầu ra chính được ghi lại và hoặc là chuyển tới người thích hợp hoặc là được lưu trữ -> test closure
* Kiểm tra hoàn thành kiểm thử
* Chuyển giao test artifacts
* Đóng gói các kết quả, log, báo cáo và các văn bản khác

1. **CHƯƠNG 2: SOFTWARE LIFE CYCLE BASED TESTING**
   1. Testing in software life cycle
      1. Testing trong quy trình Waterfall

* Mô hình kiểm thử

Requirement gathering -> System analysis -> Coding -> Testing -> Implementation -> Operation and maintenance

* Requirement gathering: Thu thập và phân tích yêu cầu được ghi lại vào tài liệu đặc tả yêu cầu
* System analysis: Phân tích thiết kế hệ thống phần mềm, xác định kiến trúc hệ thống tổng thể của phần mềm
* Coding: Hệ thống được phát triển theo từng unit và được tích hợp trong giai đoạn tiếp theo. Mỗi unit được phát triển và kiểm thử được gọi là unit test
* Testing: Công việc chính của giai đoạn này là kiểm tra và sửa tất cả lỗi
* Implementation: Triển khai hệ thống trong môi trường khách hàng và đưa ra thị trường
* Operation ang maintenance: Bảo trì hệ thống khi có bất kì sự thay đổi nào từ phía khách hàng ngưởi sử dụng
* Ưu điểm:
* Dễ sử dụng, tiếp cận và quản lý
* Sản phẩm phát triển theo các giai đoạn được xác định rõ ràng
* Xác nhận ở từng giai đoạn, đảm bảo phát hiện sớm các lỗi
* Nhược điểm:
* Ít linh hoạt, phạm vi điều chỉnh hạn chế
* Rất khó đo lường sự phát triển trong từng giai đoạn
* Không thích hợp đối với các dự án lớn, phức tạp
* Khó quay lại khi giai đoạn nào đó đã kết thúc
  + 1. Testing trong những quy trình có vòng lặp
* Ở mô hình này, các quá trình sẽ được lặp đi lặp lại theo từng giai đoạn, những giai đoạn sau sẽ là cơ sở để hỗ trợ hoàn thiện chức năng được xây dựng từ các giai đoạn trước. Vào cuối mỗi lần lặp sẽ tạo ra một phiên bản mới của phần mềm theo đó kiểm thử sẽ phải kiểm thử chức năng mới.
* Ưu điểm:
* Xây dựng và hoàn thiện các bước sản phẩm theo từng bước
* Giảm thời gian làm tài liệu so với thiết kế
* Một số chức năng làm việc có thể được phát triển nhanh chóng vá sớm trong vòng 1 đởi.
* Ít tốn kém hơn khi thay đổi phạm vi, yêu cầu
* Dễ quản lý rủi ro
* Nhược điểm:
* Yêu cầu tài nguyên nhiều
* Các vấn đề về thiết kế hoặc kiến trúc hệ thống có thể phát sinh bất cứ luc nào
* Yêu cầu quản lý phức tạp hơn
* Tiến độ của dự án phụ thuộc nhiều vào giai đoạn phân tích rủi ro
  + 1. Testing trong quy trình Agile
* Phát triển dựa trên mô hình lặp và gia tăng tập trung vào khả năng thích ứng của quy trình và sự hài òng của khách hàng bằng cách phân phối nhanh sản phẩm phần mềm.
* Test plan:
* Phạm vi test
* Những chức năng mới cần sẽ test
* Test level và test type
* Lòd test và performance test
* Phân tích cơ sở hạ tầng
* Các rủi ro và giải pháp
* Nguồn lực
* Sản phẩm bàn giao và các mốc thời gian
* Strategies
* Iteration
* Construction iterations
* Release
* Quadrant
  1. Testing level
     1. Conponent testing
* Cơ sở kiểm thử
* Các yêu cầu thành phần
* Thiết kế chi tiết
* Code
* Đối tượng kiểm thử
* Các thành phần (function, procedure, class, method…)
* Các module, chương trình
* Chuyển đổi dữ liệu thay đổi chương trình
* Bao gồm
* Kiểm thử chức năng và phi chức năng
* Các trường hợp kiểm thử bắt nguồn từ các đặc điểm kỹ thuật của thành phần, thiết kế phần mềm hoặc các mô hình dữ liệu
* Kiểm thử thành phần thực hiện khi truy cập code.
* Người kiểm thử là lập trình viên
  + 1. Integration testing
* Cơ sở kiểm thử
* Thiết kế hệ thống và phần mềm
* Kiến trúc
* Workflow
* Usecase
* Đối tượng kiểm thử
* Triển khai cơ sở dữ liệu các hệ thống con
* Cơ sở hạ tầng
* Giao diện
* Cấu hình hệ thống – cấu hình dữ liệu
* Kiểm thử tích hợp thành phần
* Kiểm thử tích hợp hệ thống
* Phân loại

Dựa vào cấu trúc

* Kiểm thử chức năng
* Kiểm tra hiệu năng
* Kiểm tra khả năng chịu tải
  + 1. System testing
* Cơ sở kiểm thử
* Chi tiết yêu cầu của hệ thống và phần mềm
* Use case
* Chi tiết các chức năng
* Báo cáo phân tích rủi ro
* Đối tượng kiểm thử
* Hệ thống, người dùng và hướng dẫn cách hoạt động
* Cấu hình hệ thống
* Phân loại
* Kiểm thử chức năng
* Kiểm thử hiệu năng
* Kiểm thử khả năng chịu tải
* Kiểm thử cấu hình
* Kiểm thử khả năng bảo mật
* Kiểm thử khả năng phục hồi
  + 1. Acceptance testing
* Cơ sở kiểm thử
* Yêu cầu người dung
* Yêu cầu hệ thống
* Usecase
* Các quy trình nghiệp vụ
* Báo cáo phân tích rủi ro
* Đối tượng kiểm thử
* Các quy trình nghiệp vụ trên hệ thống đã được tích hợp đầy đủ
* Các quy trình vận hành bảo trì
* Các phương thức người dung
* Các Forms
* Các báo cáo
* Phân loại
* Kiểm thử Alpha
* Kiểm thử Beta

1. **CHƯƠNG 3: UNIT TESTING**
   1. Boundary value testing
   2. Equivalence class testing
   3. Decision table
   4. Path test
   5. Data flow testing
2. **CHƯƠNG 4: INTEGRATION SYSTEM TESTING**
   1. Decomposition
   2. Graph
   3. path
3. **CHƯƠNG 5: UI TESTING**
4. **CHƯƠNG 6: VALIDATION TESTING**