Fpt University

FPT POLYTECHNIC

NHẬP MÔN KỸ THUẬT PHẦN MỀM

Bài 7: Testing Basic

www.poly.edu.vn

Điểm danh

Nội dung bài học

1. Unit Test Fundamentals
2. Manual Unit Testing
3. Automated Unit Testing

1. Unit Test Fundamentals

Unit Test Fundamentals

Nguyên tắc cơ bản của Unit Test là phải trả lời các câu hỏi:

- **WHAT and WHO**
- **WHY**
- **WHEN**
- **HOW**

Unit Test Technique: Introduce approaches and techniques to do unit test

Unit Test Fundamentals – WHAT and WHO ?

Unit Testing Actions:

- Xác nhận rằng các unit của phần mềm đang làm việc đúng.
- Unit là các phần nhỏ nhất trong phần mềm (nó có thể là function, procedure,... Trong lập trình hướng đối tượng thì Unit luôn luôn là method)
- Units khác với modules. Module được tạo thành từ các units

★ Unit Testing Deliverables:

- Tested software units
- Tài liệu liên quan (Unit Test case, Unit Test Report)

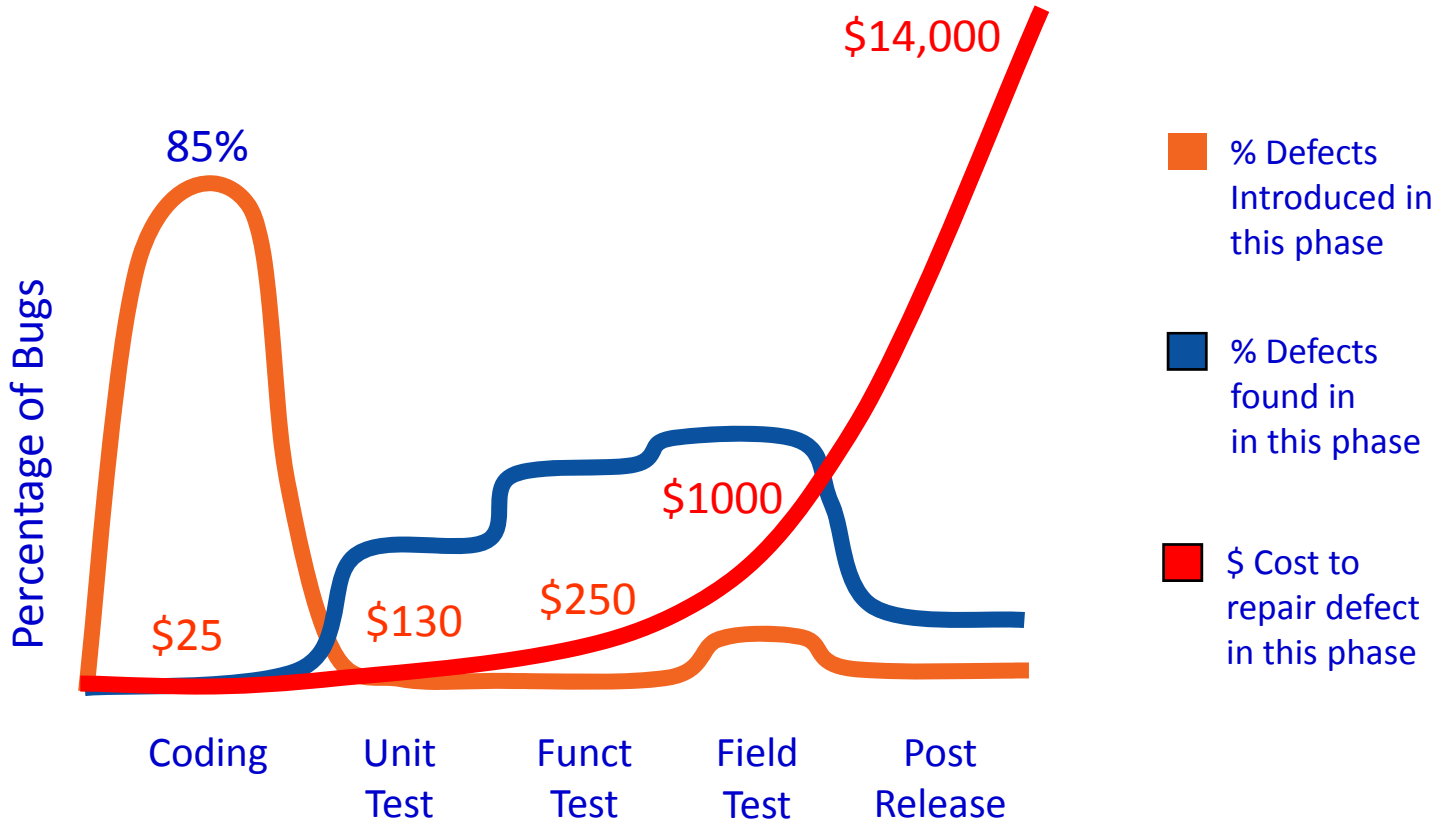
★ Unit Testing Conductor: Development Team

Unit Test Fundamentals – WHY?

- Đảm bảo chất lượng của các đơn vị phần mềm
- Phát hiện lỗi và các vấn đề sớm
- Nâng cao chất lượng phần mềm & giảm chi phí sữa chữa

Unit Test Fundamentals – WHY?

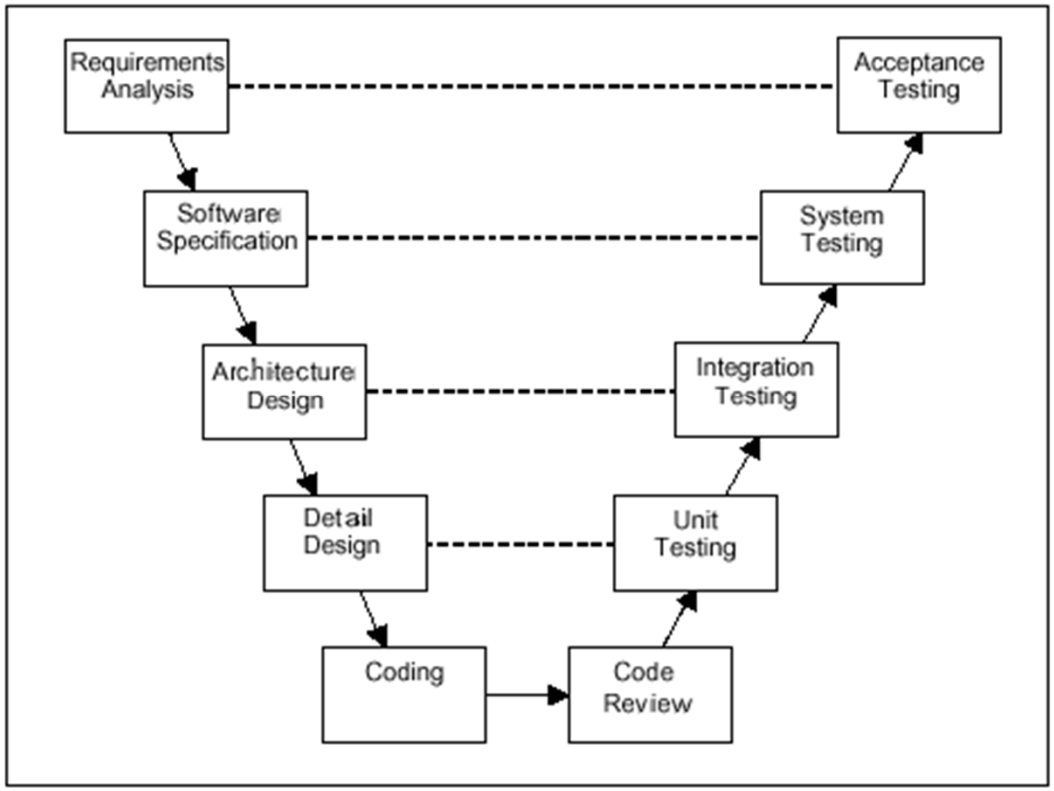
Cost of bugs



Source: Applied Software Measurement, Capers Jones, 1996

Unit Test Fundamentals – WHEN?

- Sau khi Coding
- Trước khi kiểm thử tích hợp (Integration Test)

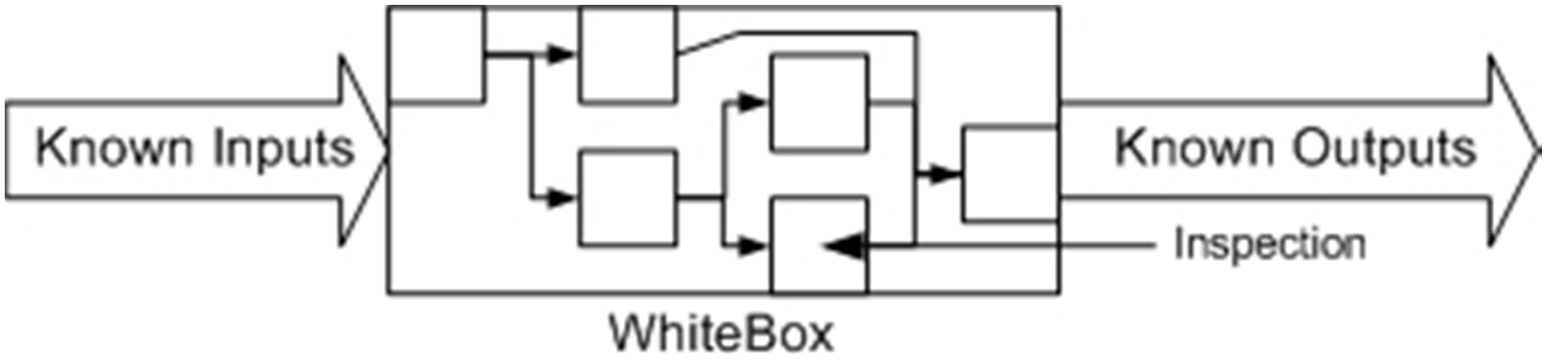


Unit Test Fundamentals – HOW?

A Simple Blackbox Specification

- Black-box testing
  - ★ Functional testing: chắc chắn mỗi unit hoạt động đúng theo thiết kế.
  - ★ Business testing: chắc chắn software program hoạt động đúng theo yêu cầu requirement.

Unit Test Fundamentals – HOW?



White-box testing

- ✓ Kiểm tra cú pháp của Code bởi trình biên dịch để tránh các lỗi cú pháp.
- ✓ Chạy Code trong chế độ gỡ lỗi, từng dòng, thông qua tất cả các đường dẫn độc lập của chương trình để đảm bảo rằng tất cả các code đã được thực hiện ít nhất một lần
- ✓ Kiểm tra cấu trúc dữ liệu local để đảm bảo rằng dữ liệu được lưu trữ tạm thời duy trì tính toàn vẹn của nó trong tất cả các bước của mã thực thi
- ✓ Kiểm tra điều kiện biên để đảm bảo rằng mã sẽ chạy đúng như là yêu cầu
- ✓ Xem lại tất cả các xử lý lỗi

2. Manual Unit Testing

Manual Unit Testing

1. Viết code
2. Uploading the code
3. Build
4. Running the code manually (sử dụng nhiều case từng bước để test)
5. Kiểm tra Log files, Database, External Services, giá trị các biến, Output trên màn hình,…
6. Nếu nó không chạy, lập lại các bước trên.

Manual Unit Testing – Hạn chế

- Developer nhớ được trường hợp nào thì test trường hợp đó
- Đến cuối dự án số lượng test case càng lúc càng nhiều, khả năng cover của lập trình viên giảm xuống!
- Nhiều test case bị trùng lắp
- Nhiều test case bị lack
- Team lead không thể review hết được
- ➔ Kết quả dự án chỉ trông chờ vào tester!!!!
- ➔ Rất nhiều lỗi phát sinh sau khi system test, đa phần các lỗi xuất phát do Dev test không kỹ từ lúc Unit Test!

Manual Unit Testing – Hạn chế

- Để giải quyết vấn đề trên, mỗi khi developer test xong phải viết tài liệu mô tả test case trên word hoặc excel !
- Điều này giúp team rất dễ dàng review. Tuy nhiên, cách làm này vẫn có thể có hạn chế!

Function Code	Function1	Function Name	Function A
Created By	<Developer Name>	Executed By	
Lines of code	100	Lack of test cases	-5
Test requirement	<Brief description about requirements which are tested in this function>		

Passed	Failed	Untested	N/A/B		Total Test Cases
0	0	15	5	1 1	15

		UTCID01	UTCID02	UTCID02	UTCID02	UTCID02	UTCID02	UTCID07	UTCID08	UTCID09	UTCID10	UTCID11	UTCID12	UTCID13	UTCID14	UTCID15
Condition	Precondition															
	a															
	-2	O														
	-1							O								
	0		O	O	O											
	1					O	O									
	b															
	0		O	O												
	-2					O	O	O								
	2				O											
	c															
	0		O													
	1			O	O	O										
	3							O								
	5						O									
Confirm	Return															

Manual Unit Testing – Hạn chế

- Các dự án lớn thì số lượng tài liệu test case thường cũng rất lớn!
- Các dự án lớn thì requirement thường hay thay đổi
- Mỗi khi requirement thay đổi➔ Phải sửa code➔ phải cập nhật lại tài liệu testcase ➔và lại manual retest , rất tốn sức
- ➔Càng đến cuối dự án, lượng việc sinh ra càng nhiều , viết test case document trở thành “địa ngục “ thực sự !➔ dev không còn đủ sức update test case document, tài liệu nhanh chóng bị lạc hậu, hoặc việc update chỉ là đối phó!
- Một số trường hợp không thể dùng Excel Unit TestCase

So, what is the solution?

Manual Unit Testing -

Thảo luận

So, what is the solution?

3. Automated Unit Testing

Automated Unit Testing

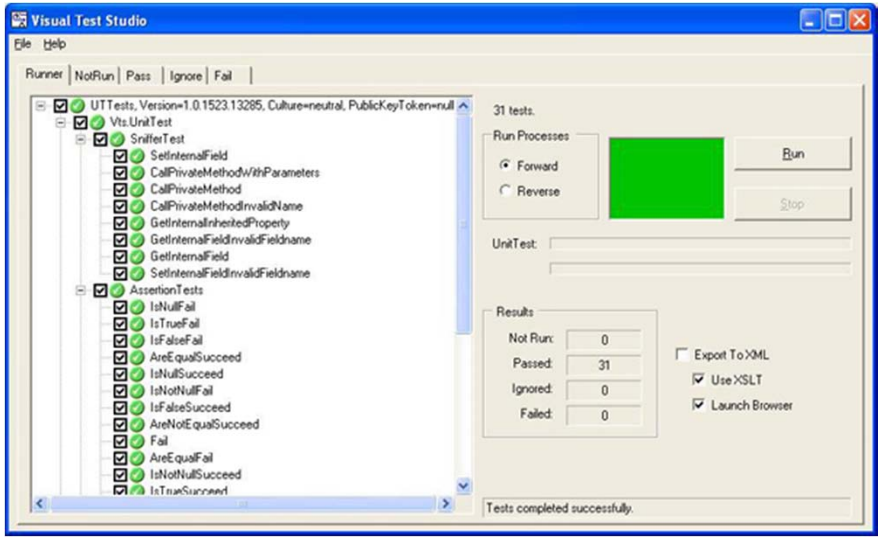
Coding Process with Automated Unit Tests

- Viết code
- Viết một hoặc nhiều test cases script
- Auto-compile and run (Biên dịch tự động và chạy)
- Nếu tests fail -> fix lỗi.
- Nếu tests pass -> lặp lại với method khác.

Automated Unit Testing

Công cụ sử dụng:

Java: JUnit, J2MEUnit
C/C++: cppUnit
Python: pyUnit
Perl: PerlUnit
Visual Basic: vbUnit
C# .NET: Nunit,csUnit

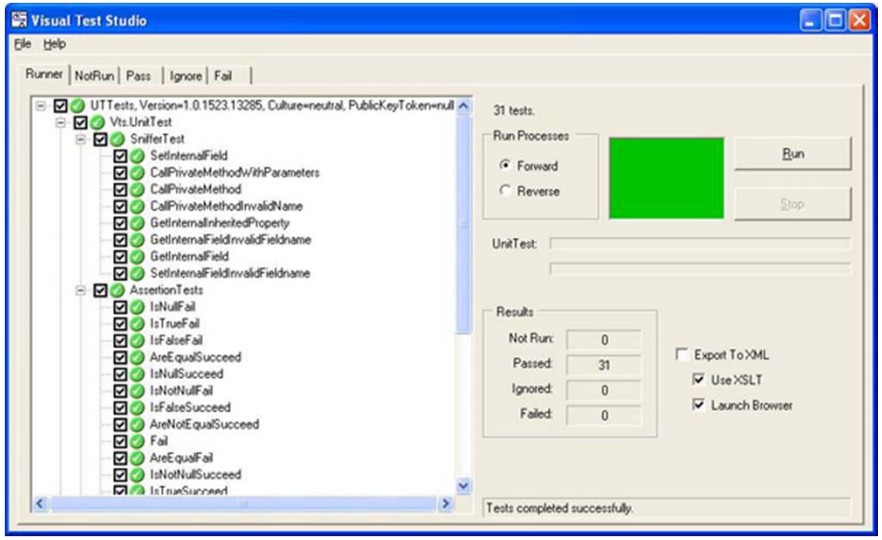


- Tham khảo:
  - http://www.testingfaqs.org/t-unit.html
  - www.junit.org
  - http://www.codeproject.com/gen/design/autp5.asp

Automated Unit Testing

Công cụ sử dụng:

Java: JUnit, J2MEUnit
C/C++: cppUnit
Python: pyUnit
Perl: PerlUnit
Visual Basic: vbUnit
C# .NET: Nunit,csUnit



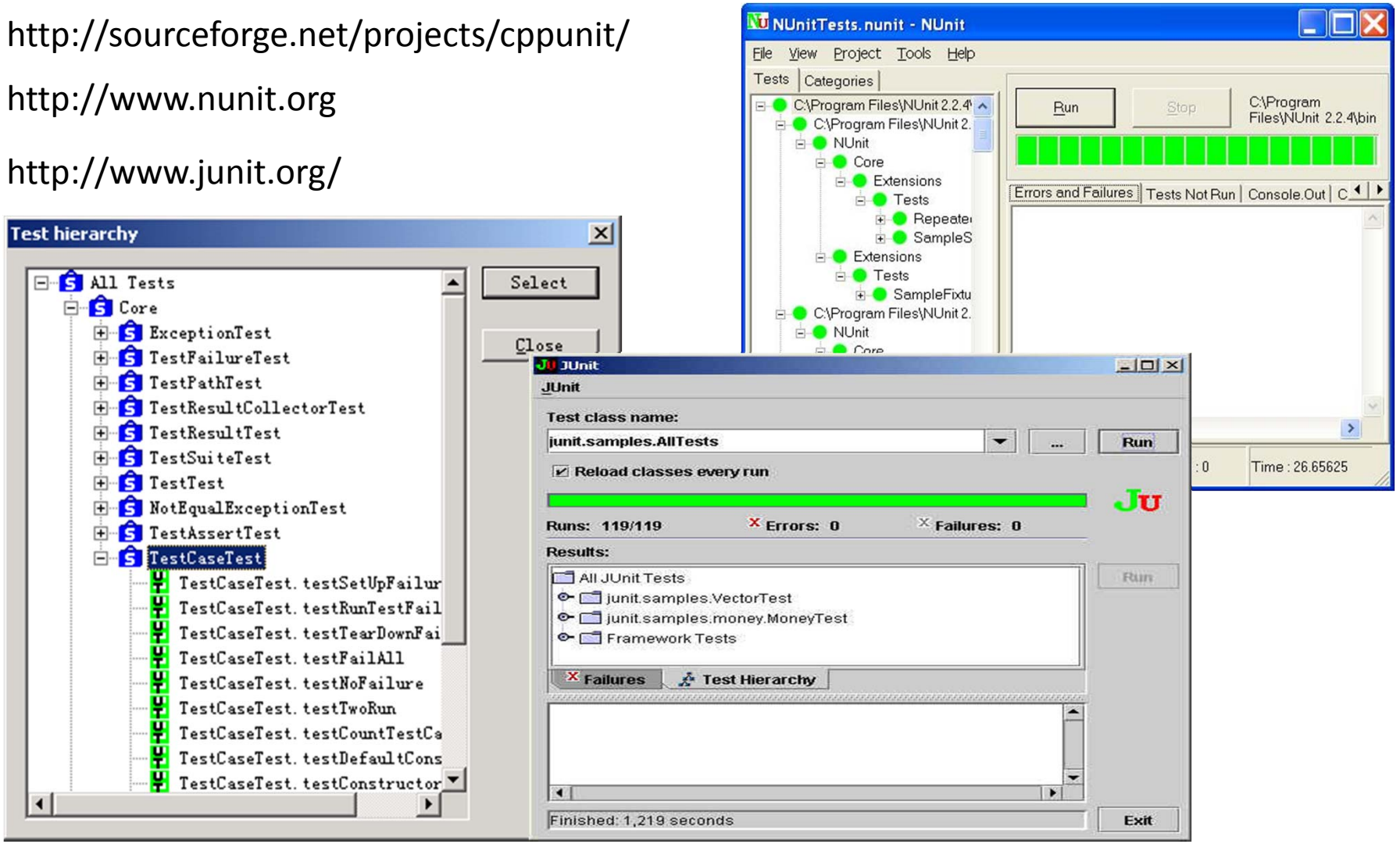
- Tham khảo:
  - http://www.testingfaqs.org/t-unit.html
  - www.junit.org
  - http://www.codeproject.com/gen/design/autp5.asp

Automated Unit Testing

http://sourceforge.net/projects/cppunit/

http://www.nunit.org

http://www.junit.org/



Unit Testing Document

Schroders Screen Unit Test

I. Control

Project	SIM.<AppCode>	Component name	Screen 01
Author signature & visa	Tran Duy Vinh	Finished date	yyyymmdd

II. Screen Generic Test-cases

#	Operations	Expected results	OK or NA
A	**Layout Check**		
1	**Page properties**		
1.1	Check that layout fit well with defined minimal resolution screen	No horizontal scrollbar. No double vertical scrollbar	
1.2	Check window's title (spelling) for all application's languages	The title has a correct spelling in all application's languages according to the design document	
1.3	Check that cache is disabled	Do a back navigation to check refresh	
1.4	Image button always contains a tool tip	Images as link should have appropriate message tip/hint	
1.5	Items/datas in dropdown box is alphabetically sort	Normally sorted by alphabetically	
1.6	Check version number is appear within the application	Normally in About page	
1.7	Check if about box is available and accessible	Web Site should have About box/page	
1.8	Check if copyright information is centered in the	"© 2008 Schroder Investment Management (S)	

Comment [Loc1]: Project Leader put NA if he considers that the test case is not relevant in the project context.

THẢO LUẬN MỞ

Nhóm sinh viên cùng làm Unit Test Document theo yêu cầu Giảng viên

Chuẩn bị Workshop 5

- Nhóm sinh viên chuẩn bị Unit Test Document cho đề tài đã chọn

Tổng kết nội dung bài học

1. Unit Test Fundamentals
2. Manual Unit Testing
3. Automated Unit Testing


KẾT THÚC