

mike6606

[Blog Park](#) | [Home](#) | [New Essa](#) | [contact](#) | [subscribe](#) | [manage](#) |

Essays - 40 Articles - 0 Comments - 0 Views - 25651

Nickname: mike6606
Age: 8 years 10 months
Fans: 0
Followers: 5
+ Plus follow

< March 2023 >

day	One	Two	Three	Four	Five	Six
26	27	28	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	1
2	3	4	5	6	7	8

Search

Frequently used links

[My essays](#)
[My comment](#)
[My involvement](#)
[Latest comments](#)
[My labels](#)

Essay archives

[January 2021 \(1\)](#)
[January 2018 \(3\)](#)
[January 2017 \(8\)](#)

Read the leaderboard

1. Software Engineering Practitioners' Research Methods Chapter 13 Answers (1578)
2. Software Engineering Practitioners' Research Methods Chapter 1471 Answer s (<>)
3. Software Engineering Research Methods for Practitioners Chapter 1290 Answers (<>)
4. Software Engineering Practitioners' Research Methods Chapter 34 Answers (1282)
5. Software Engineering Practitioners' Research Methods Chapter 1264 Answer s (<>)

Recommended leaderboards

1. Software Engineering Practitioners' Research Methods Chapter 1 Answers (<>)

Software Engineering Practitioners' Research Methods Chapter 22 Answers

Problem:

Using your own words, describe the difference between verification and validation. Do both make use of test-case design methods and testing strategies?

Answer:

Following are the differences between verification and validation:

Verification	Validation
Verification refers to a set of tasks that are carried out to check whether the software correctly implements a specific function.	Validation refers to a different set of activities that are carried out to ensure that the software is built as per the customer requirements.
Verification focuses on process.	Validation focuses on product.
It checks whether the product that is built is correct or not.	It checks whether the right product is built or not.
The objective of verification is to ensure that the product meets the requirements and design specifications.	The objective of validation is to ensure that the product meets the user requirements.
It deals with internal product specifications.	It deals directly with users and their requirements.
The techniques of the verification are reviews, meetings, inspections etc.	The techniques of the validation are white box testing, black box testing, gray box testing etc.
The quality assurance (QA) team does verification.	The testing team does validation.
Verification is done before performing validation	Validation is done after performing verification

Both Verification and Validation make use of test case design methods and testing strategies. The testing strategy is based on the spiral model. A test plan specifies the tests to be conducted and the process of testing to uncover any errors in the product. A test case consists of a test data to be given along with the expected output.

Problem:

List some problems that might be associated with the creation of an independent test group. Are an ITG and an SQA group made up of the same people?

Answer:

592-13-2P SA CODE: 4578

SR CODE: 4467

A common practice of software testing is that it is performed by an independent group of testers after the functionality is developed but before it is shipped to the customer. This practice often results in the testing phase being used as project buffer to compensate for project delays, thereby compromising the time devoted to testing. The role of an ITG is to remove the inherent problems associated with letting the builder test the thing that has been built. Independent testing removes the conflict of interest that may otherwise be present. After all, ITG personnel are

while testing is conducted, the developer must be available to correct errors that are uncovered. The ITG is part of the software development project team in the sense that it becomes involved during analysis and design and stays involved (planning and specifying test procedures) throughout a large project. However, in many cases the ITG report to the software quality assurance organization, thereby achieving a degree of independence that might not be possible if it were a part of the software engineering organization.

The ITG may get involved in a project too late—when time is short and a thorough job of test planning and execution cannot be accomplished.

An ITG and an SQA group are not necessarily the same. The ITG focuses solely on testing, while the SQA group considers all aspects of quality assurance.

Problem:

Is it always possible to develop a strategy for testing software that uses the sequence of testing steps described in Section 22.1.3? What possible complications might arise for embedded systems?

Answer:

Testing Strategy

Consider the steps in testing strategy:

1. Unit Testing
2. Integration Testing
3. Validation Testing
4. System Testing

From the above testing steps, it is not possible to develop a strategy for testing embedded systems.

Because, while conducting through the unit testing, the complexity of a test environment is accomplished to the unit testing. It may not justify the benefit.

When these modules are felled behind the schedule in Integration, testing process is complicated by the scheduled availability. This situation occurs especially in the unit-tested modules

But, especially for embedded systems, the validation testing cannot be conducted outside of the target hardware configuration. For this reason, the validation and system testing are combined.

Hence, it is not always possible to develop software by supposed testing sequence.

Problem:

Why is a highly coupled module difficult to unit test?

Answer:

592-13-4P SA CODE: 4578

SR CODE: 4467

In highly coupled modules, errors can propagate to different module (tested module). So the paths need to be tested will increase due to the shared data or other module invocation. Highly coupled components are tightly connected to each other.

A highly coupled module is one who has at least part of its interface closely tied to one or more other modules. This means that the closely related modules can not be completely tested independently (nominal unit testing) but must be tested as a unit (more nearly sub-integrated testing). The larger (in terms of code size and complexity) the code, the more difficult it is to test. Further, action occurring in different modules is harder to trace, code read, design realistic tests for, etc.

Problem:

The concept of “antibugging” (Section 22.2.1) is an extremely effective way to provide built-in debugging assistance when an error is uncovered:

a. Develop a set of guidelines for antibugging.

b. Discuss advantages of using the technique.

Answer:

4633-17-5P SA: 9420

SR: 6376

a)

Antibugging: A good design anticipates error conditions and establishes error-handling paths to reroute or cleanly terminate processing when an error does occur. Yourdon calls this approach antibugging.

A single rule covers a large number of situations: All data moving across software interfaces (both external and internal) should be validated.

The guidelines are

- The functions or methods must be able to turn over at least an error code or an exception.
- Any error code or possible exception must be tested and used.
- An "abnormal" error must stop the execution of the program and write the state of the variables and system in a disk file

b)

Advantages:

1. Errors don't "snowball".
2. The errors are immediately found.
3. The errors cannot affect the remainder of the.
4. The programmer discovers and is motivated with quickly repairing the introduced errors.

c)

Disadvantages:

1. Does require extra processing time and memory.
2. Introduced many codes and tests which will never be used.
3. A program which stops completely will create a bad impression for the user.

Problem:

How can project scheduling affect integration testing?

Answer:

592-13-6P SA CODE: 4578

SR CODE: 4467

When the project scheduling is done, each module finished date will be established or set. In the other hand integration testing has to wait for all modules to finish and to be unit tested (if possible). So any changes to the project scheduling will directly affect the integration testing.

Because integration testing is done after a certain point in the project life time any changes to the schedule before that point will affect the integration test schedule.

Problem:

Is unit testing possible or even desirable in all circumstances? Provide examples to justify your answer.

Answer:

No. If a module has 3 or 4 subordinates that supply data essential to a meaningful evaluation of the module, it may not be possible to conduct a unit test without "clustering" all of the modules as a unit.

The quality of the units is better and integrated quality of the system is also better. A bug discovered as of these first tests will further save much time in the development.

Problem:

Who should perform the validation test—the software developer or the software user? Justify your answer.

Answer:

Validation testing is done in two phase: Alpha testing and beta testing.

users.

Problem:

Develop a complete test strategy for the *SafeHome* system discussed earlier in this book. Document it in a *Test Specification*.

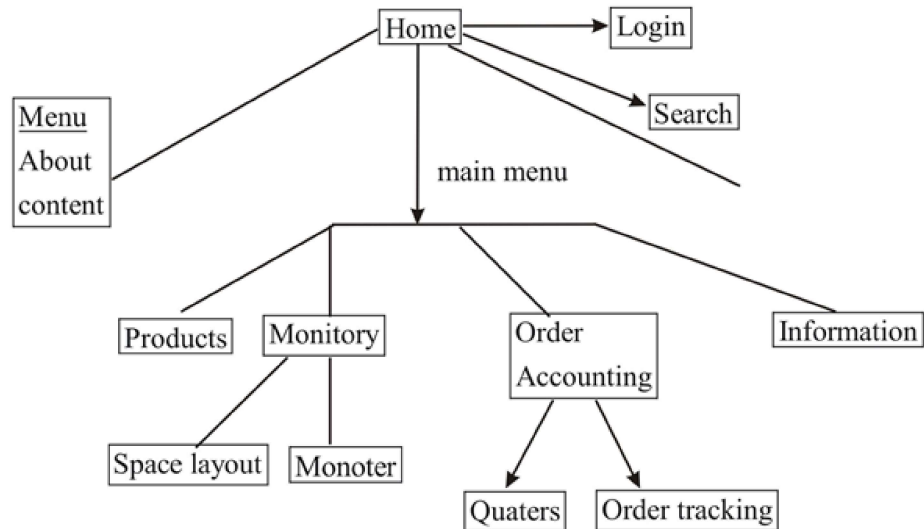
Answer:

592-13-9p SA CODE: 4578

SR CODE: 4467

Test strategy for the safe home system:

Safe home system design, that is consists of various features.



Test strategy for above one:

Generally, the objective of testing is to find the greatest possible number of errors with a manageable amount of effort applied over a time span.

Test strategy follows the below process.

- (1) System Engineering
- (2) Requirements
- (3) Design
- (4) Coding
- (5) Unit testing
- (6) Integration testing
- (7) Validation testing
- (8) System testing

So, from the above strategy, we need to test for the login page in safe home application.

Login page consists of

Member ID	<input type="text"/>
Password	<input type="password"/>
<input type="button" value="login"/>	

For this we may apply

Test case Id - 01

Objective - To check the user login

Precondition - log in page should be open with its all conditions.

Test data - User name and password

Step 1

Write correct user name and password in respective text boxes.

Step 2

Check on the log in button

Expected Result:

Used login successfully and home page should be open

Result

(1) Actual Result:

(2) :

Comments

Problem:

As a class project, develop a *Debugging Guide* for your installation. The guide should provide language and system-oriented hints that you have learned through the school of hard knocks! Begin with an outline of topics that will be reviewed by the class and your instructor. Publish the guide for others in your local environment.

Answer:

592-13-10P SA CODE: 4578

SR CODE: 4467

→ In the second phase, debugging is occurs during the latter stage of testing, involving multiple components or a complete system.

→ Third debugging process is place at production / deployment. When the software under test faces real operational conditions

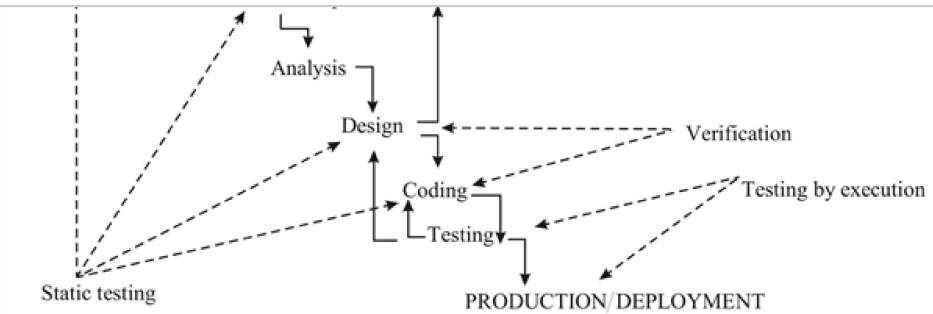
This process may called as "problem determination"

Debugging process is done by following stages.

→ Cooling

→ Testing

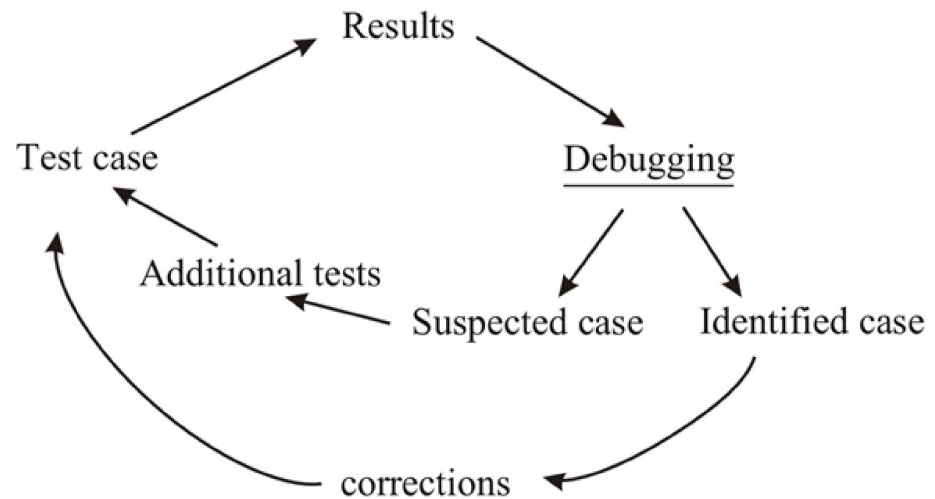
→ Production / deployment



Debugging:

Debugging is the one of the successful testing process. That occurs, when a test case uncovers an error. This process begins with the execution of a test case. Results are assessed through expected and actual parameters.

Debugging process is done through below fields.



In the software development, debugging activities are in three types. That is

→ During the coding process, the programmer translates the design into an executable code. In this process, errors made by the programmer

Solution: CHAPTER 22: SOFTWARE TESTING STRATEGIES

22.1 Verification focuses on the correctness of a program by attempting to find errors in function or performance. Validation focuses on "conformance to requirements—a fundamental characteristic of quality.

22.2 The most common problem in the creation of an ITG is getting and keeping good people. In addition, hostility between the ITG and the software engineering group can arise if the interaction between groups is not properly managed. Finally, the ITG may get involved in a project too late—when time is short and a thorough job of test planning and execution cannot be accomplished. An ITG and an SQA group are not necessarily the same. The ITG focuses solely on testing, while the SQA group considers all aspects of quality assurance.

22.3 It is not always possible to conduct thorough unit testing in that the complexity of a test environment to accomplish unit testing (i.e., complicated drivers and stubs) may not justify the benefit. Integration testing is complicated by the scheduled availability of unit-tested modules (especially when such modules fall behind schedule). In many cases (especially for embedded systems) validation testing for software cannot be adequately conducted outside of the target hardware configuration. Therefore, validation and system testing are combined.

22.4 A highly coupled module interacts with other modules, data and other system elements. Therefore its function is often dependent of the operation of those coupled elements. In order to thoroughly unit test such a module, the function of the coupled elements must be simulated in some manner. This can be difficult and time consuming.

Advantages: Errors don't "snowball."

Disadvantages: Does require extra processing time and memory (usually a small price to pay).

22.6 The availability of completed modules can affect the order and strategy for integration. Project status must be known so that integration planning can be accomplished successfully.

22.7 No. If a module has 3 or 4 subordinates that supply data essential to a meaningful evaluation of the module, it may not be possible to conduct a unit test without "clustering" all of the modules as a unit.

22.8 Developer, if customer acceptance test is planned. Both developer and customer (user) if no further tests are contemplated. An independent test group is probably the best alternative here, but it isn't one of the choices.

22.9 Answers will vary

22.10 Answers will vary



[mike6606](#)

followers - [0 followers](#) - 5

0

0

[+ Plus follow](#)

« Previous: [Software Engineering Research Methods for Practitioners Chapter 20 Answers](#)

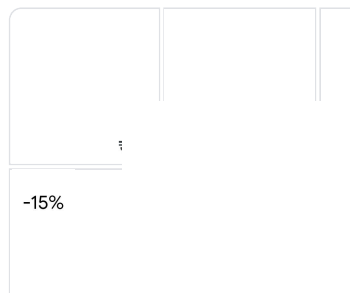
» Next: [Software Engineering Research Methods for Practitioners Chapter 26 Answers](#)

posted @ 2021-01-22 00:50 mike6606 read (590) Comments (0) Edit Favorite report

[Refresh Comment](#) [Refresh page](#) [returns to the top](#)

Log in to view or post comments, [log in](#) now or [browse](#) the blog home page

【Alibaba Cloud】2-core 2G cloud server as low as 99 yuan/year, <> cloud products enjoy non-stop discounts



Editor's Choice:

- Modern Image Performance Optimization - Fault Tolerance and Accessibility of Image Resources
- Talk: The Evolution of Service Architecture
- I'm on timeout with Redis again
- Clever use of CSS variables to create high-end grid animations
- Record the optimization of a lock

Reading Ranking:

- 3 mistakes to avoid when using Vue 10
- ASP.NET Core Web API interface throttling
- 【Fault Bulletin】CC attack is coming again, making March worse ·

Modern Image Performance Optimization and Experience Optimization Guide - Fault tolerance and accessibility of image resources

- How to JWT authenticate and authorize WebAPI in Net6.0

