Solutions Manual

Software Quality Assurance

From Theory to Implementation

Daniel Galin

For further instructor material please visit:

www.booksites.net/galin

ISBN 0273705660

PEARSON
Addison Wesley

Harlow, England • London • New York • Boston • San Francisco • Toronto • Sydney • Singapore • Hong Kong
Tokyo • Seoul • Taipei • New Delhi • Cape Town • Madrid • Mexico City • Amsterdam • Munich • Paris • Milan

Pearson Education Limited
Edinburgh Gate
Harlow
Essex CM20 2JE
England

and Associated Companies around the world

Visit us on the World Wide Web at:
http://www.pearsoned.co.uk

First published 2005

ISBN: 0273705660

Contents

Part I

Introduction

CHAPTER 1

The software quality challenge

Review questions

1.1

There are three major differences between software products and other industrial products.

1. Identify and describe the differences.

2. Discuss the ways in which these differences affect SQA.

Solution

1. Software products differ from other industrial products with respect to the following characteristics:
   - **Product complexity** – a typical software product allows tens of thousands of operational options. Typical industrial products and even advanced industrial products do not reach this level of variety of options.
   - **Product visibility** – since software products are invisible, defects in the software are not visible unless testing procedures are applied. In contrast, industrial products are visible and most defects are visible to the production team by changes in color or shape. Even non-professionals can observe color changes that reveal defects in chemical and food products, or the changes in shape or structure that reveal defects in household products.
   - **Development and production process** – Defects in industrial products usually occur during each stage through which the product has to pass, namely development, product production planning, and manufacturing. At each stage the product is examined and tested independently by a different team. In contrast, software products are examined and tested during the development stage only. The product production planning and the manufacturing stages deal mainly with the duplication and packaging of the software product without contributing to the detection of software defects.

2. The first characteristic, product complexity, makes the task of quality assurance for software much more difficult, as the product must function correctly for all of the defined options, even those that are highly complicated and rarely used. The other two characteristics make software defects more difficult to detect. The combined effect of these software characteristics creates a situation in which software developers cannot be certain that the software product they supply is defect-free.

1.2

It is claimed that no significant SQA activities are expected to take place during the phase of production planning for software products.

1. Discuss this claim.

2. Compare the required production planning for a new automobile model with the production planning efforts required for the new release of a software product.

Solution

This review question refers mainly to the issues discussed in review question 1.1, but whereas question 1.1 compares a software product to an industrial product, this review question focuses on a comparison between the software product development process and the development process of an industrial product.

1. This claim is correct. The product production planning stage and the manufacturing stage in the case of software products deal mainly with duplication and packaging of the product, with no activities of review or testing of the software code. Therefore, it is not expected that software defects will be detected at these stages.

2. There is considerable similarity between the development of a new automobile model and a new software project. Both processes are characterized by creativity and require specific reviewing and testing procedures. It should be noted that many of the complicated operational options of automobiles are controlled by software (operated by the car's computer or computers).

1.3

Seven issues characterize the professional software development and maintenance environment.

1. Identify and describe these characteristics.

2. Which of these environmental characteristics mainly affect the professional efforts required for carrying software development and maintenance projects. List the characteristics and explain why a professional effort is needed.

3. Which of these environmental characteristics mainly affect the managerial efforts required for carrying out software development and maintenance projects. List the characteristics and explain why such efforts are needed.

Solution

1. The characteristics of the software development and maintenance environment are:
   - Contract conditions and commitments defining content and timetable.

- Conditions of the customer–supplier relationship, as exemplified by the need for consultation with customers and securing of their approval.
- Teamwork requirements.
- Need for cooperation and coordination with other software and hardware development teams both internally and externally.
- Need for interfaces with other software systems.
- Need for continuity in carrying out a project when team members change.
- Need for ongoing maintenance of the software system over several years.

2. Each of the abovementioned characteristics affects both the professional and the managerial efforts, though usually to differing extents. The following characteristics affect mainly the professional side:
   - **Contract conditions defining content** – the need to prepare a document listing functional and other requirements of the project.
   - **Conditions of the customer–supplier relationship** – the need to maintain ongoing contacts with the customer's professionals for presentations of development products, consultations with the customer, and securing customer approval of the development products.
   - **Teamwork requirements** – the need for the team leader to take responsibility for professional instruction of the team members and checking of their products.
   - **Cooperation and coordination with other software and hardware development teams both internally and externally** – the need to understand the tasks performed by other teams to the extent that enables proper professional communication.
   - **Interfaces with other software systems** – the need to acquire familiarity with the interfacing standards or interfacing design of equipment units and/or software packages.

3. The following characteristics affect mainly the managerial side:
   - **Contract conditions defining timetable** – the commitments relating to completion of a project and usually also to completion of each stage of the development process.
   - **Teamwork requirements** – the need to recruit a team and appoint a team leader to manage and to supervise the work of each of the team members.
   - **Continuity in carrying out a project when team members change** – the need to recruit, sometimes at short notice, a replacement team member having the professional knowledge and experience similar to those of the departing member.
   - **Ongoing maintenance of the software system over several years** – the need to ensure the constant availability of updated documentation on the software system, and to maintain a professional team well acquainted with the software system and capable of providing support at short notice.

Topics for discussion

1.1

Educational systems are assumed to prepare the students to cope with real life conditions. Examine the procedural requirements of a software development project or final software project, and determine what of the requirements could be considered as preparatory to professional life situations as discussed above.

Solution

Only part of the software development environment can be practiced within the framework of educational systems.

Let us examine this issue with reference to each of the seven environmental characteristics:

- Contract conditions and commitments defining content and timetable. Students' projects simulate contract conditions to some extent. A typical students' project includes definitions of the required functions as well as the time schedule for completion. Budget commitments are naturally not applicable.

- Conditions of the customer−supplier relationship, as exemplified by the need for consultation with customers and securing of their approval. The instructor–student relationship simulates to some extent the relationship between customer and supplier.

- Teamwork requirements. Projects done by student teams incorporate some aspects of team work, but are usually carried out without an appointed team leader.

- Need for cooperation and coordination with other software and hardware development teams both internally and externally. This is usually not applicable in students' projects.

- Need for interfaces with other software systems. This is applicable in some cases.

- Need for continuity in carrying out a project when team members change. This is usually not applicable in students' projects.

- Need for ongoing maintenance of the software system over several years. This is usually not applicable in students' projects.

1.2

Referring to the seven environmental characteristics of software development and maintenance, consider the characteristics of future software products, discussing whether the professional and managerial burden of coping with these characteristics in future is expected to be higher or lower when compared with the current performance of these activities.

Solution

- **Contract conditions and commitments defining content and timetable.** Customers can be expected to be much more demanding with respect to full implementation of functional and other requirements. Typical time schedules for similar development projects are expected to be substantially shorter than those currently allowed.

- **Conditions of the customer–supplier relationship.** The nature of future projects is likely to demand a much closer relationship between client and supplier.

- **Teamwork requirements.** Teamwork will continue to be required, though it is expected that new technologies will be implemented to support teamwork.

- **Need for cooperation and coordination with other software and hardware development teams both internally and externally.** These characteristics will become increasingly critical for the successful handling of projects. More comprehensive standardization will facilitate more effective coordination and cooperation.

- **Need for interfaces with other software systems.** The number of interfaces and the intensity of their use can be expected to increase. More comprehensive standardization of interfaces between software systems and between software and hardware will facilitate more effective development of interfaces.

- **Need for continuity in carrying out a project when team members change**. No significant change is expected.

- **Need for ongoing maintenance of the software system over several years.** No significant change is expected.

1.3

The interfaces of a salary processing system are exhibited in Frame 1.2.

1. Suggest what are the main benefits of applying computerized interfaces instead of transferring printouts.

2. Give two additional examples where input interfaces are applied.

3. Give two additional examples where output interfaces are applied.

4. Suggest additional situations where the use of input and output interfaces is not applied and should be recommended.

5. Would you advise all information transfers from one organization to another be performed by computerized interface? Discuss the reasons behind your answer.

Solution

1. The main benefits are:
   - Reducing the time period required to handle the input and update the system's database, and contributing to better up-to-dateness of the information provided by the system.
   - Drastically reducing the percentage of input errors, thus significantly improving the accuracy and completeness of the system's outputs.
   - Drastically reducing the human resources required to handle the input, both for keying in the input data and for correcting identified errors. This reduction will significantly reduce the costs of handling the input.

2., 3. A software interface serves two software systems, since it serves as an output interface for one and as an input interface for the other. Let us examine the following two examples:

Example 1: A monthly procedure of money transfers from the bank account of an employer to the employees' bank accounts. The required interface is between the software system for calculation of salaries and the bank's system for transferring money to customers' accounts.

Example 2: A centralized price update procedure for a network of stores. This procedure distributes the price updates and sale prices fixed by the network's head office, thus ensuring that all stores apply a uniform pricelist. Efficient operation of this procedure requires an interface between the price and sales management system of the head office and the point-of-sale system of the store.

The following table summarizes the two examples presented above:

Procedure carried out by the interface	Software system for which it serves as an output interface	Software system for which it serves as an input interface
Procedure for money transfers from the employer's bank account to employees' bank accounts	Employer's software system for salary calculation	Bank's system for transferring money to customers' accounts.
Centralized price update procedure for a network of stores	Head office price and sales management system	Store's point-of-sale system

4. Another example: Interface between a personal computerized health log system and a clinic's computerized information system. The computerized personal health system would probably be encapsulated as a "smart card", whose database includes

personal medical records. The clinic's system allows the medical staff to decide on the treatment needed by the patient, and to record diagnoses, medical treatments, medications, etc. A standard interface between personal health log systems and the clinic's information system, stationed in clinics, hospitals, first aid centers, etc., is required in order to benefit from the medical information it provides. The service provided by this two-way interface is summarized in the following table:

Procedure carried out by the interface	**Software system for which it serves as an output interface**	**Software system for which it serves as an input interface**
Transfer of patient's medical information to the clinic system	Personal computerized health log system	Clinic's computerized information system
Transfer of records of current medical treatment to update the patient's health log system	Clinic's computerized information system	Personal computerized health log system

5. In some situations an automatic computerized transfer of data may be not desirable.
   - Where an interface is hardly used, the expense is not economically justifiable.
   - If the data sources are of low quality, the transfer of information will be inaccurate and incomplete. In such cases the data and any necessary corrections should be checked manually before any input into the database of the recipient can be considered reliable.

1.4

The need to carry out work by a team demands additional investment in coordination of the team members. Discuss whether these managerial efforts could be saved if the work were performed as a "one-man job".

Solution

Teamwork obviously requires a team leader, who needs to spend much time on coordination among the team members, so that the work done by each of them can be assembled into a unified software system. The costs of these activities are overheads to software development costs. Obviously these extra costs are negligible by operating via a "one-man show". It should be emphasized, however, that a substantial part of a team leader's time is invested in checking the work of the team members. If the project is "a one-man show" the task of checking still needs to be performed by another member of the staff, possibly the head of department.

1.5

It is clear that a software development project carried out by a software house for a specific customer is carried out under content and timetable obligations, and is subject to the customer–supplier relationship.

1. Discuss whether a customer–supplier relationship is expected when the software developed is to be sold to the public as a software package.

2. Discuss whether a customer–supplier relationship is expected when software is developed for "in-house" use, as in the case where a software development department develops an inventory program for the company's warehouses.

3. Some managers claim that the closer relationships are to a formal pattern, the greater the prospects are for the project's success. Discuss whether implementing customer-supplier relationships in the situations mentioned in (1) and (2) are a benefit for the company (referring to the internal customer and supplier) or an unnecessary burden to the development team.

Solution

1. In the development of COTS (commercial off the shelf) software packages, the customer is the marketing department that initiates and approves the development project.

2. In the development of an internal project, the customer is the initiating department, the finance department, or the logistics department.

3. A formal relationship between an internal supplier (systems development department) and an internal customer can be expected to be beneficial for both parties by:

   - Providing more realistic project planning for scheduling and budgeting.
   - Providing more comprehensive and realistic plans with respect to functionality of the software products.
   - Contributing to better scheduling and budgetary control of the project.

   It might be claimed that a formal atmosphere among the internal parties to the project will reduce the creativity of the development team.

1.6

It has been claimed that environmental characteristics create the need for intensive and continuous managerial efforts parallel to the professional efforts that have to be invested in order to ensure the project's quality or, in other words, to assure the project's success. Discuss the reasons behind this claim, including an analysis of the managerial effort created by each of the SQA environmental characteristics.

Solution

The characteristics of the managerial environment in software development and maintenance are discussed in Review question 1.3, section (3).

The following characteristics require effort mainly on the part of management:

– **Contract conditions defining timetable.** The possibility of schedule failures can be minimized only by continuous and intensive follow-up of a project's progress at the management level, especially in problematic cases where additional team members are needed or in a situation that needs to be resolved by negotiation with the customer. Rigorous follow-up procedures will ensure earlier detection by management of deviations from schedule and their easier correction.

– **Teamwork requirements.** Teamwork requires managerial abilities in addition to professional qualities on the part of the team leader. One of the managerial aspects of teamwork is the need to recruit and instruct team members.

– **Need for continuity in carrying out a project when team members change.** Management needs to recruit, sometimes at short notice, a replacement team member having the professional knowledge and experience similar to those of a departing team member.

– **Need for ongoing maintenance of the software system over several years.** Whereas professionals engaged in a development process may resign from the company without being obliged to complete their work, management personnel are usually committed to these projects over relatively long periods. It is the duty of management to ensure the constant availability of updated documentation on the software system during this period, and to keep the professional team well acquainted with the software system and capable of carrying out maintenance tasks at short notice.

Proper managerial support, as discussed above, allows the professional teams to focus on the functional requirements of the project.

CHAPTER 2

What is software quality?

Review questions

2.1

A software system comprises of four main components.

1. List the four components of a software system.
2. How does the quality of each component contribute to the quality of the developed software?
3. How does the quality of each component contribute to the quality of the software maintenance?

Solution

1. The four components of a software system are:
   - Computer programs (the "code")
   - Procedures
   - Documentation
   - Data necessary for operating the software system.
2. The contributions of these components to the quality of developed software are:
   - Computer programs (the "code") – obviously, its quality is the basic component for the quality of services and functionality of the software product.
   - Procedures which define the methods of the program development process, i.e. software development planning procedure, design review procedure, software testing procedure and progress control procedure, contribute to the quality of the software product.
   - Development documentation (the requirements report, design reports, program descriptions, software testing plan, etc.) allows efficient cooperation and coordination amongst the development team members, easier replacement of any team member who leaves the team and efficient reviews and inspections of the design and programming products.
   - Data, including parameters and code lists that adapt the software to the specifications as well as test case files are necessary for testing the software before completion of the development process is possible.
3. The contributions of these components to the quality of maintenance services are:

- Computer programs (the "code") – obviously, its quality is the basic component for the quality of services and functionality of the software product.
- The procedures that accompany the software system deal with both, the regular operation of the software system and its maintenance. The regular software operational procedures define the method of program employment and the responsibilities for performing input processing, output processing and control activities. The maintenance procedures define the processes and responsibilities for the correction of "bugs". Another group of procedures deals with changes and improvements of programs, their approval and performance. The quality of these types of procedures contributes to the quality of services the software system provides.
- Documentation supports both users and maintenance professionals. The user's documentation (the "user's manual" etc.) provides a description of the available applications and the appropriate method for their use. Their quality is a major factor regarding the ability of users to successfully and efficiently apply the software applications. The maintenance documentation (the "programmer's software manual", etc.) provides the maintenance team with all the required information about the code and the structure and tasks of each software module. This information is used when trying to locate causes of software failures ("bugs") or to change or improve an existing software system.
- Data including parameters code lists that adapt the software to the needs of the specific user are necessary for operating the software. Another type of essential data is the standard test data, used to ascertain that no undesirable changes in the code or software data have occurred and to determine what kind of software malfunctioning can be expected.

2.2

1. Define software error, software fault and software failure. Explain the differences between these undesirable software statuses.
2. Suggest a situation where a new type of software failure ("bug") appears in a software package that has been serving 300 clients for the first time six years since the software package was first sold to the public.

Solution

1. A **software error** can be a grammatical or logical error in trying to comply with one or more of the client's requirements included in one or more of the code lines.

   A **software fault** is a software error which can cause improper functioning of the software in general or of a specific application.

   A **software failure** is a failure that has been "activated" and causes improper functioning of the software as a whole or of a specific, faulty application.

2. Let's refer to a meteorological application based on remote measuring stations; in this case the remote unit is required to initiate a protective closing operation that prevents

damage being caused to cold-sensitive measuring equipment each time the temperature drops below minus 20ºC. The programmer's software error defines the lower temperature limit as minus 30ºC. The meteorological information system purchased by several Spanish organizations was installed in various sites in Spain, none of which suffers from low temperatures. The fault only turned into a failure once a Russian version of the system had been developed and installed in various northern sites of Russia, causing severe damage to the measuring stations' equipment.

2.3

1. List and briefly describe the various causes of software errors.
2. Classify the causes of error according to the groups responsible for the error: the client's staff, the systems analysts, the programmers, the testing staff – or is it a shared responsibility belonging to more than one group?

Solution

1. Nine causes of software errors are listed:

   a. Faulty definition of requirements
   The faulty definition of requirements, usually prepared by the client, is one of the main causes of software errors. The most common errors of this type are: erroneous definition of requirements, absence of vital requirements, incomplete definition of requirements and inclusion of unnecessary requirements.

   b. Client-developer communication failures
   Misunderstandings resulting from defective client-developer communication are additional causes of errors. Typical situations: Misunderstanding of the client's instructions relating to the requirement document and to changes requested either written or orally by the client. Additional misunderstandings are failures to understand and to give the needed attention to the client's response to design problems raised by the development team.

   c. Deliberate deviations from software requirements
   In several circumstances, developers may deliberately deviate from the documented requirements, often causing software errors. Common situations of this type: reuse of software modules taken from an earlier project, omission of part of the required functions in an attempt to cope with time or budgetary pressures and developer-initiated improvements, introduced without the client's approval.

   d. Logical design errors
   Software errors of this type are mainly failures of systems architects, software engineers, systems analysts, etc., to formulate the software requirements into the proper algorithms, boundary conditions, omission of required system states, etc.

   e. Coding errors
   The reasons that cause programmers to make coding errors include misunderstanding the design documentation, linguistic errors, errors in the application of CASE, other development tools, and so forth.

   f. Noncompliance with documentation and coding instructions
   One may ask why noncompliance with coding instructions should cause

software errors. It **is** believed that "non-complying" software is expected to increase the rate of errors made by development and maintenance teams. It is due to erroneous understanding by other members of the development team and maintenance team when performing corrections or changes.

g. **Shortcomings of the testing process**
Shortcomings of the testing process affect the error rate by leaving a greater number of undetected or uncorrected errors. These shortcomings result from: incomplete test plans, failure to document and report detected errors and faults, failure to promptly correct detected software faults and incomplete correction of detected errors.

h. **Procedure errors**
Procedure errors, especially in complex software systems, may incorrectly direct the user with respect to the activities required.

i. **Documentation errors**
The documentation errors that trouble the development and maintenance teams are errors in the design documents and in the documentation integrated into the body of the software. These errors can cause additional errors in further stages of development and during maintenance.

Another type of documentation error, one that mainly affects the user, is an error in the user manuals and in the "help" displays incorporated in the software.

2. The responsibility for errors

The group responsible is marked in the following table. Some of the errors are of shared responsibilities.

Type of error	**Client's staff**	**Systems analysts**	**Programmers**	**Testing staff**
a. Faulty definition of requirements	R			
b. Client-developer communication failures	R	R		
c. Deliberate deviations from software requirements		R		
d. Logical design errors		R		
e. Coding errors			R	
f. Noncompliance to documentation and coding instructions		R (design)	R (coding)	
g. Shortcomings of the testing process				R
h. Procedure errors		R		R
i. Documentation errors		R (design)	R (coding)	

2.4

1. What are the differences between the IEEE definition of SQA and the expanded definition used in this book?

Solution

- **Life cycle phase**. The IEEE SQA definitions referring to the software are limited to the development phase, while the expanded SQA definition covers the whole software life cycle, namely the development phase and maintenance during operation phase.
- **Subjects included.** The IEEE SQA definitions are limited to the technical aspects of the functional requirements, while the expanded SQA definition also includes activities dealing with scheduling and the budget of software development and maintenance.

2.5

Mr Johnson is a customer of the Adams and Lincoln stores belonging to the Eiffel chain (See section 2.3). His purchase records and returned goods records are as follows:

	Adams Store		**Lincoln Store**	
Month	**Purchases ($000)**	**Returned goods ($000)**	**Purchases ($000)**	**Returned goods ($000)**
Jan 2000	100	20	60	5
Feb 2000	120	10	40	--
Mar 2000	10	--	30	10
Apr 2000	80	5	50	10
May 2000	30	--	20	--
Jun 2000	60	20	30	10
Jul 2000	10	--	40	--
Aug 2000	60	5	10	--
Sep 2000	20	--	20	5
Oct 2000	20	5	40	10
Nov 2000	40	--	20	--
Dec 2000	20	--	60	5
Jan 2001	30	10	40	--
Feb 2001	60	5	30	5
Mar 2001	20	5	40	10

1. Find for which of the months – Jan. 2001, Feb. 2001 or Mar. 2001 – does Mr Johnson qualify for the 5% discount? What is the sum discounted? Calculate according to the correct procedure.

2. According to the erroneous procedures, find for which of the months – Jan. 2001, Feb. 2001 or Mar. 2001 – does Mr Johnson qualify for the 5% discount in the Adams store and in the Lincoln store? What is the sum discounted?

Solution

1. Correct calculations of Mr Johnson's discount

	Adams Store		**Lincoln Store**		**The Eiffel Store Chain**				
Month	**Purchases ($000)**	**Returned goods ($000)**	**Purchases ($000)**	**Returned goods ($000)**	**Total purchases ($000)**	**Total returned goods ($000)**	**Total 12 month purchases ($000)**	**Total 3 months purchases ($000)**	**Total 3 months returns ($000)**
Jan 2000	100	20	60	5	160	25			
Feb 2000	120	10	40	--	160	10			
Mar 2000	10	--	30	10	40	10			
Apr 2000	80	5	50	10	130	15			
May 2000	30	--	20	--	50	--			
Jun 2000	60	20	30	10	90	30			
Jul 2000	10	--	40	--	50	--			
Aug 2000	60	5	10	--	70	5			
Sep 2000	20	--	20	5	40	5			
Oct 2000	20	5	40	10	60	15			
Nov 2000	40	--	20	--	60	--			
Dec 2000	20	--	60	5	80	5	990	200	20
Jan 2001	30	10	40	--	70	10	900	210	15
Feb 2001	60	5	30	5	90	10	830	240	25
Mar 2001	20	5	40	10	60	15			

Mr Johnson's eligibility for discount – the correct calculations

Month	**Eligibility according to total 12 month purchases**	**Eligibility according to total 3 month returns**	**Discount due for purchases in Adams store ($000)**	**Discount due for purchases in Lincoln store ($000)**
January 2001	No	Yes (though on the edge)	0	0
February 2001	No	Yes	0	0
March 2001	No	No	0	0

2. *Incorrect calculations of Mr Johnson's discount*

	Adams Store		**Lincoln Store**		**The Eiffel store Chain**				
Month	**Purchases ($000)**	**Returned goods ($000)**	**Total 3 month purchases ($000)**	**Total 3 month returns ($000)**	**Purchases ($000)**	**Returned goods ($000)**	**Total 3 month purchases ($000)**	**Total 3 months returns ($000)**	**Total 12 month purchases ($000)**
Jan 2000	100	20			60	5			
Feb 2000	120	10			40	--			
Mar 2000	10	--			30	10			
Apr 2000	80	5			50	10			
May 2000	30	--			20	--			
Jun 2000	60	20			30	10			
Jul 2000	10	--			40	--			
Aug 2000	60	5			10	--			
Sep 2000	20	--			20	5			
Oct 2000	20	5			40	10			
Nov 2000	40	--			20	--			
Dec 2000	20	--	80	5	60	5	120	15	990
Jan 2001	30	10			40	--			
Feb 2001	60	5			30	5			
Mar 2001	20	5			40	10			

Mr Johnson's eligibility for discount – the incorrect calculations

	Adam and Lincoln Chain	**Adams Store**		**Lincoln Store**	
Month	**Eligibility according to total 12 month purchases**	**Eligibility according to total 3 month returns**	**Discount due ($000)**	**Eligibility according to total 3 month returns**	**Discount due ($000)**
January 2001	No	Yes	0	No	0
February 2001	No	Yes	0	No	0
March 2001	No	Yes	0	No	0

To sum up: Though the aforementioned calculations are substantially different, they yield the same result – no discount for Mr Johnson during the aforementioned months. It may serve as an example of an entirely erroneous calculation process that still yields a correct result. For this test case a black box test will not detect the major defect if the erroneous calculation process is applied.

Let us now compare the results in the case of a company that determines a customer whose purchases are $900,000 or more in the last 12 months is entitled to the discount, unless his returns rate for the last 3 months exceeds 10%.

Mr Johnson's eligibility for discount – the correct calculations

Month	**Eligibility according to total 12 month purchases**	**Eligibility according to total 3 month returns**	**Discount due for purchases in Adams store ($000)**	**Discount due for purchases in Lincoln store ($000)**
January 2001	Yes	Yes (though on the edge)	1.5	2
February 2001	Yes	Yes	3	1.5
March 2001	No	No	0	0
Total discount ($000)	------------------	------------------	8	

Mr Johnson's eligibility for discount – the incorrect calculations

	Adam and Lincoln Chain	**Adams Store**		**Lincoln Store**	
Month	**Eligibility according to total 12 month purchases**	**Eligibility according to total 3 month returns**	**Discount due ($000)**	**Eligibility according to total 3 month returns**	**Discount due ($000)**
January 2001	Yes	Yes	1.5	No	0
February 2001	Yes	Yes	3	No	0
March 2001	Yes	Yes	1	No	0
Total discount ($000)	--------------	--------------	5.5		

2.6

According to the IEEE definition of SQA, quality control (QC) is not equated with quality assurance (QA).

1. In what respects does QC vary from QA?

2. Why can QC be considered part of QA?

Solution

1. Both QC and QA have a common objective, namely to bring the quality of a software product to a required controlled level. However QA has an additional important objective – to achieve the aforementioned basic objective, while minimizing the costs of quality. Thus, QC varies from QA by the scope of tools applied. QC focuses on evaluating the quality of a developed product by inspection and testing, and withholding of the software product until the detected errors are repaired and the software product qualifies.

   The objective of quality assurance is to minimize the cost of guaranteeing quality by a variety of activities performed throughout the development and maintenance processes, and by adding activities that prevent the causes of errors and extensively applying them to detection and correction activities early in the development process.

2. As evaluation activities of the software product that comprise QC are a major part of QA (the other major part being prevention activities), consequently QC is a part of QA.

2.7

Examine the definitions of SQA and the objectives of SQA activities.

1. Is there a correspondence between the two definitions?

2. If yes, show how the objectives of SQA activities aim at the implementation of the SQA concepts.

Solution

1. Obviously, there is a correspondence.

2. In order to examine the conformance, I will refer to the components of the SQA expanded definition. The continuous improvement activities are directed to affect the productivity and effectiveness of efficiency of all processes carried out throughout the software life cycle (development and maintenance), and are marked in the table accordingly.

The following table presents the correspondence between the objectives of SQA activities and the SQA expanded definition.

	SQA Objectives of development activities			**SQA Objectives of management activities**		
Components of the expanded SQA definition	**Assure conformance to functional technical requirements**	**Assure conformance to managerial scheduling and budgetary requirements**	**Initiation and management of activities for the improvement**	**Assure conformance to functional technical requirements**	**Assure conformance to managerial scheduling and budgetary requirements**	**Initiation and management of activities for the improvement**
1. Systematic and planned activities	X	X	X	X	X	X
2. Conformance with functional development requirements	X		X			
3. Conformance with functional technical requirements of maintenance process				X		X
4. Keeping schedule and budgetary requirements of the development process		X	X			
5. Keeping schedule and budgetary requirements of the maintenance process					X	X

Topics for discussion

2.1

A programmer claims that because only a small proportion of software errors turn into software failures, it is unnecessary to make substantial investments in the prevention and elimination of software errors.

1. Do you agree with this view?

2. Discuss the outcome of accepting these views.

Solution

1. Considering the damage caused by software errors, one must invest as much effort as possible in eliminating software errors to reduce these damages to a minimum.

Like the manufacturing industries, the target and aspiration of the software industry is to achieve "zero defect" quality.

2. Some possible outcomes of the "no investments in errors prevention and detection" policy:
   - Lack of control over the extent of possible failures especially of severe failures.
   - A software failure may threaten a patient's life or a soldier in combat.
   - Software failures are not limited to user's waste of time until the failure is repaired, but cause damage to other systems. An example: failure of the software system that processes shipping documents not only causes loss of time of the shipping clerk but also loss of time of the warehouse team, truck drivers and others.
   - Software failures occur at unexpected times and can cause maximum damage. In addition to failures that occur during the running-in of the software system and during the first period of operation of the new software system, a situation causing an increase in software failures may occur even after a few years.

2.2

George Wise is an exceptional programmer. Testing his software modules reveals very few errors, far fewer than the team's average. He keeps his schedule promptly, and only rarely is he late in completing his task. He always finds original ways to solve programming difficulties, and uses an original, individual version of the coding style. He dislikes preparing the required documentation, and rarely does he do it according to the team's templates.

A day after completing a challenging task, on time, he was called to the office of the department's chief software engineer. Instead of being praised for his accomplishments (as he expected), he was warned by the company's chief software engineer that he would be fired unless he began to fully comply with team's coding and documentation instructions.

1. Do you agree with the position taken by the department's chief software engineer?

2. If yes, could you suggest why his/her position was so decisive?

Solution

1. Yes.

2. The use of non-standard coding and documentation methods by George Wise cause:
   - Extra difficulties to other programmers who have to develop software modules that need to interface with George's module that may result in errors.
   - Extra difficulties to an inspection team and testing team that may result in lower than regular rates of error detecting.

- Extra difficulty to replacement programmer who might be recruited to continue the work of George in the case of his leaving the company or being promoted to a higher position in another project. Misunderstanding George's coding and documentation may result in software errors.
- Extra difficulty in performing maintenance task of failure repairs, adaptation of the software to new customers and system improvement tasks.

2.3

Pressman's definition of quality requires the client to specify the software requirements because only documented requirements are binding for the developer. Any omissions or errors made by the client are considered as his/her fault, and not listed among the developer's errors.

1. How can a client be sure that his or her organization has the professional capabilities to cope with this issue?

2. In what ways can the developer support the client in this matter?

3. Suggest pro and con arguments to Pressman's definition of the client's responsibility.

Solution

1. Managements are expected to be aware of their capabilities in the various areas required in order to operate their organization. In many cases small and also medium size organizations do not employ an information systems professional and their managements should be aware of their lack of capabilities. In cases of major changes in the information systems, even in large organizations the internal IT department does not usually possess the knowledge and experience needed to cope with professional developments. It can be expected that the internal team will request external professional support.

2. The developer can provide professional support for the preparation of a requirement document. The contents of this document should be explained in detail to the client and be approved by them.

3. **Pro arguments:**

   a. It forces the client to be responsible for fully and accurately defining their requirements.

   b. The client is the one best acquainted with the environment and operational needs of the organization and therefore is the most suitable to define the project requirements.

   c. Transferring the responsibility for the definition of the requirements to the developer reduces the motivation of the client to supply full details of all the operation situations and risks the omission of important operational scenarios.

4. **Con arguments**:
   a. As a professional the developer is experienced with a wide variety of systems, their advantages and disadvantages. Equipped with this experience and professional know-how they are far better equipped to investigate the organization's needs and define the requirements.
   b. As an external professional they are expected to define requirements that incorporate updated capabilities and tools offered in the market. An internal team will tend to stick to the existing system and leave less space for the inclusion of new developments.

2.4

It is claimed that the expanded definition of SQA supports those who are interested in increasing client satisfaction.

1. Do you agree with this claim?

2. If yes, provide arguments to substantiate your position.

Solution

1. I agree.

2. a. The expanded SQA definition adds the elements of managerial schedule and budget control. As projects that do not keep to schedule and / or exceed their budget tend to be under severe managerial pressures to save project resources and perform procedures more quickly, the prospects of fully achieving all the quality goals and an acceptable extent of software errors are reduced substantially. Thus the inclusion of schedule and budget control in the SQA controls reduces the prospect of failure – increases the customer's satisfaction. (b) Another element added to the SQA expanded definition refers to software maintenance quality. Maintaining SQA efforts throughout the software life cycle leads to the continuing satisfaction of the client extended over long years of operation.

2.5

Examine the correct and erroneous procedures determining the discount qualification outlined in Table. 2.1.
List the procedure errors.

Solution

The errors are:

a. It is required that the discount will be determined entirely by the central information processing department and uniformly applied in all the chain stores. According to the erroneous procedure, the goods returns are individually examined in each store causing different discount eligibility lists in the various chain stores. In other words, a customer may find that he is eligible to a discount in one store but is not eligible in another store.

b. The eligibility for a discount is based on two parameters: total purchases and percentage of returns, both determined on a monthly basis – last 12 months' purchases and the last 3 months' returns. According to the erroneous procedure, the purchases parameter is determined only once a year and is valid throughout the next year.

CHAPTER 3

Software quality factors

Review questions

3.1

1. What are the three factor categories belonging to McCall's factor model?

2. What factors are included in each of the categories?

Solution

1. The three categories are:
   - Product operation category
   - Product revision category
   - Product transition category

2. The 11 factors are grouped into three categories as follows:

Product operation factors:	Correctness Reliability Efficiency Integrity Usability
Product revision factors:	Maintainability Flexibility Testability
Product transition factors:	Portability Reusability Interoperability

3.2

The software requirement document for the tender for development of "Super-lab," a software system for managing a hospital laboratory, consists of chapters according to the required quality factors as follows: correctness, reliability, efficiency, integrity, usability, maintainability, flexibility, testability, portability, reusability and interoperability.

In the following table you will find sections taken from the mentioned requirements document. For each section, fill in the name of the factor that best fits the requirement (chose only one factor per requirements section).

Solution

The answers are marked in indented font

No.	Section taken from the software requirement document	The requirements factor
1	The probability that the "Super-lab" software system will be found in a state of failure during peak hours (9 am to 4 pm) is required to be below 0.5%.	*Reliability*
2	The "Super-lab" software system will enable direct transfer of laboratory results to those files of hospitalized patients managed by the "MD-File" software package.	*Interoperability*
3	The "Super-lab" software system will include a module that prepares a detailed report of the patient's laboratory test results during his current hospitalization. (This report will serve as an appendix to the family physician's file.) The time required to obtain this printed report will be less than 60 seconds; the level of accuracy and completeness will be at least 99%.	*Correctness (referring to availability, accuracy and completeness)*
4	The "Super-lab" software to be developed for hospital laboratory use may be adapted later for private laboratory use.	*Flexibility*
5	The training of a laboratory technician, requiring no more than 3 days, will enable the technician to reach level C of "Super-lab" software usage. This means that he or she will be able to manage reception of 20 patients per hour.	*Usability*
6	The "Super-lab" software system will record a detailed users' log. In addition, the system will report attempts by unauthorized persons to obtain medical information from the laboratory test results database. The report will include the following information: the network identification of the applying terminal, the system code of the employee who requested that information, the day and time of attempt and the type of attempt.	*Integrity*
7	The "Super-lab" subsystem that deals with billing patients for their tests may be eventually used as a subsystem in the "Physiotherapy Center" software package.	*Reusability*
8	The "Super-lab" software system will process all the monthly reports for the hospital departments' management, the hospital management, and the hospital controller according to Appendix D of the development contract.	*Correctness*
9	The software system should be able to serve 12 workstations and 8 automatic testing machines with a single model AS20 server and a CS25 communication server that will be able to serve 25 communication lines. This hardware system should conform to all availability requirements as listed in Appendix C.	*Efficiency*
10	The "Super-lab" software package developed for the Linux operating system should be compatible for applications in a Windows NT environment.	*Portability*

3.3

What differentiates the Evans and Marciniak model from Deutsch and Willis model?

Solution

Both models include 10 out of the 11 factors suggested by McCall and an additional two factors not included in McCall's model, namely the verifiability and expandability factors.

Evans and Marciniak's model includes additional three factors not included in McCall's or in Deutch and Willis's models, namely the safety, manageability and survivability factors.

3.4

Consider McCall's model and the Deutsch and Willis model.

1. What are the formal differences between the models?

2. What are the content differences between the models?

3. What new subjects were actually added by the Evans and Marciniak model to McCall's model?

Solution

1. The formal differences refer to three factors. The Deutsch and Willis model adds two factors to the ones included in McCall's model, namely the verifiability and expandability factors. However, their model does not include the testability factor included in McCall's model.

2. When comparing the differences between McCall's reliability, testability and flexibility factors to Deutch and Willis's survivability, maintainability, and flexibility factors, one finds great resemblance. The actual difference between the models become much smaller. Actually the McCall model is extended by Deutch and Willis's model and includes only two additional factors: safety and manageability.

3.5

Southcottage Inc. is a manufacturer of washing machines and dishwashers. The requirements document for the new control unit included the following specifications:

a. The firmware should be suitable for all six variations of model 2002 washing machines.

b. The water level control module of the washing machine should be suitable for use as a water level control module in the new model 2002 dishwasher.

1. To which of the factors do the above requirements belong?

2. Explain your answer.

Solution

1. The specifications refer to the flexibility and reusability factors respectively.

2. Requirement (a) refers to flexibility in the use of a complete firmware product that by its very design will enable it to be used as firmware in several variations of washing machines by using a variety of parameter sets.

   Requirement (b) refers to the reuse of part of a firmware product, to a module that is introduced into entirely different firmware products and saves the efforts of developing a new module, which will only serve dishwashers.

3.6

Some people claim that testability and verifiability are actually different names for the same factor.

1. Do you agree?

2. If not, could you explain why?

Solution

1. I disagree.

2. Both testability and verifiability refer to the SQA processes during software development, requiring software features that make the SQA processes easier. However, while verifiability is limited to verification processes, testability includes validation and qualification processes performed throughout the development processes. Also, while verifiability is limited to the software development processes, testability requirements are extended to refer to the entire software life cycle. Testability requirements refer to failure detection and analysis during regular operation of software or when a failure occurs.

Topics for discussion

3.1

Four “but” complaints are mentioned in Section 3.1. All of them reflect items missing from the requirement documents.

1. To which factors do the missing requirements belong?

2. Can you suggest software quality requirements that could fill the gap?

Solution

1. First “but”: A high rate of software failures – reliability.

   Second “but”: Adaptation of the firmware product is possible to a new product environment – portability.

   Third “but”: A lack of program defect detection features and sufficient program documentation to support the maintenance team in defect detection – testability.

   Fourth “but”: More training resources are needed, many more than expected – usability.

2. First “but”: Full availability of all working stations and software application should exceed 98% of office hours. In other words, in a 50-hour-week operation the total failure time (fully or partially) should not exceed 1 hour a week.

   Second “but”: Bearing in mind the near-future product development programs, a set of additional new product operation environments should be defined.

   Third “but”: Defect detection features should be defined and the follow up of keeping documentation instructions should be controlled.

   Fourth “but”: Some quantitative requirements should be defined, for example: training time required to reach level C of customer service operator (Operating a CRM software package) should not exceed 8 hours.

3.2

Some professionals claim that increased software usability necessarily involves decreased efficiency. Others claim no dependence between software efficiency and usability.

1. Do you agree with the first or the second group?

2. Discuss your answer.

Solution

1. Both claims may be correct – depending on the actual situation.

2. Achieving better usability in many cases involves better design of the flow of calculations and especially better man–machine interface implemented in a well-designed GUI (graphical user interface). In this case no additional equipment or additional hardware capabilities are required and increased usability does not affect the efficiency.

In many other cases, the replacement of low capability equipment with higher capability equipment and additional hardware components are involved with the achievement of higher usability. In these cases higher usability necessarily causes decreased efficiency.

3.3

The city of Mountain View has decided to develop a software package that will serve the youth clubs operated by the city. The software's main tasks will be:

- Follow up of monthly payments of the members.
- Preparing lists of participants in the various courses offered by the clubs.
- Production of reminder notices to course participants who fail to appear regularly.
- Statistical reports about membership and participation in club activities.

The city already implements the following software packages:

- Tax collection.
- Public library.
- School follow up and achievements control.
- Water consumption billing.

The City Council has asked the Information Technology Unit to report to the council about the possibilities for reuse of the city software packages already available to the city in the youth club software package.

1. Could you suggest which modules of the existing city software packages could be reused in the new software? List your assumptions about the contents of the existing software packages and the required new software.
2. Could you grade the reused modules suggested in (1) according to the scope of adaptation efforts required to apply the reused module in the youth club software package.

Solution

1. An alternative for reuse of modules of the City's operational software packages is presented in the following table.

	City operational software packages			
Required Module	Tax collection	Public library	School follow up	Water consumption billing
1. Follow-up of members monthly payments		**Module A** Follow up of prescribers monthly payments		
2. Preparing lists of participants in the various courses offered by the clubs			**Module B** Class participants (specific lists for each teacher and class)	
3. Production of reminder notices to course participants who fail to attend regularly			**Module C** Attendance reminders sent to parents	
4. Statistical reports about membership and participation in club activities		**Module D** Statistical reports about prescribers population	**Module E** Statistical reports about attendance	

According to the above, 5 modules of only 2 of the City's packages are reused.

2. The 5 modules have been graded according to the adaptation efforts, which are:

   First grade, module A – Follow up of prescriber's monthly payments. Both the existing module and the required module deal with listed membership that pays monthly for membership.

   Second grade, module C – Attendance reminders sent to parents, is similar to the required module 3. Although the rules for proper attendance are different in schools and youth clubs, once improper attendance is discovered – the production of reminder notices is similar.

   Third grade, includes modules B, D and E which could be reused for required modules 2 and 4. Class participants (specific lists for each teacher and class). Preparing lists of participants in the club is relatively simple as registration to the courses is direct. School participation in a specific class is not registered directly but determined by a variety of parameters such as registration to program, inclusion in group of pupils, etc.

   – Statistical reports about library prescribers population and school attendance may vary substantially from those required for controlling and managing youth club activities.

3.4

It is said that failure to meet the interoperability requirements can negatively affect the correctness level of the software system, and even can cause nonconformance with correctness requirements.

1. Elaborate on the above statement and explain the mentioned interconnections between factors.

2. Provide an example of a situation where such effects are to be expected.

Solution

1. Most interoperability requirements refer to automatic transfer of data from one system to another, where one system may be an information system or computerized equipment that transfers its output into an information system. Failure to meet interoperability requirements are mainly of two types: a. Computerized data transfer fails and a manual data transfer has to be applied, or b. The data transfer process is defective and causes undesirable changes of data and entire data records to be deleted.

   Let us now consider the effects on four of the dimensions / components of the correctness factor: accuracy, completeness, up-to-datedness and availability. The expected effects of interoperability failures of the two types are presented in the following table.

Correctness dimension	**Type of interoperability failures**	
	a. Computerized data transfer fails and a manual data transfer has to be applied	**b. The data transfer process is defective causing undesirable changes of data**
Accuracy	Manual input processes resulting in much higher rates of input errors than computerized data transfer. Expected result – the input accuracy will be below the required level.	Defective transfer process that involves erroneous input records affects, by definition, the accuracy of the input, which may fall beneath the minimal level of accuracy.
Completeness	Manual input processes are under a much higher risk of losing records than the computerized system. Thus, non-compliance with completeness requirements could be expected.	Defective transfer process that involves missing input records affects, by definition, the completeness of the input, which may fall beneath the minimal level of completeness.
Up-to-Datedness	Manual input processes require much more time than the computerized system, this causes later updating of the database. Thus, non-compliance with up-to-date requirements could be expected.	Defective transfer process may lead to loss or damage to input data needed for updating the database. As a result the level of up-to-datedness may drop below the required up-to-datedness limit.

Correctness dimension	**Type of interoperability failures**	
	a. Computerized data transfer fails and a manual data transfer has to be applied	**b. The data transfer process is defective causing undesirable changes of data**
Availability	As availability refers to response time, where the processing is based on data available at the moment of the request, no effect is expected on availability, even in cases where a severe decline in accuracy and completeness exists.	No effect on availability is expected for type b failures. The reasoning is the same applied to type a failures.

2. Example: Store sales and central inventory system that serves a chain of furniture stores.

   The local information system installed in each of the stores managed the sales and local inventory. The local information system interoperates with a central inventory system, and has to transfer its transactions of sales and suppliers shipments to this central system. The central inventory system supplies information on the availability of an item in other stores of the chain. Application of this information enables a store to fulfill clients' orders when an insufficient quantity of the item is found in the local store – improved service to clients and more economic inventory management.

   Failure to comply with the interoperability requirements may cause inaccurate, incomplete and out of date information about the number of available items in other stores. This situation may result in the inability of a store to supply a client's order on time; the consequential additional costs to "repair" the failure and its consequences to the stores reputation are obvious.

3.5

It is claimed that with respect to subjects where qualitative and quantitative requirements can be defined, the quantitative alternatives should be preferred.

1. Provide three examples each of alternative qualitative and quantitative requirements.

2. Explain why the customer should prefer the quantitative option.

3. Explain why the software developer should prefer the quantitative option.

Solution

1. The examples are shown in the following table.

Quality factor	Qualitative requirement	Quantitative requirement
Reliability	The software package will provide a high reliability service to the store	Full failure time will not exceed 1% of the store service time
Reusability	Major parts of the developed software package can be reused in future software development projects of the Company	At least 75% of the software modules will be suitable for reuse in the Company's planned software projects, namely the club, the spa and the CRM systems
Usability	The GUI (graphical user interface) will assure easy training and high productivity of POS (point of sale) operators	A training course not exceeding 8 hours will enable new POS operators to serve at least 8 customers per hour. An experienced POS operator will be capable of serving at least 12 customers per hour

2. Customers' arguments pro quantitative definition of requirements:
   - It allows for easier evaluation of the software product and easier determination of its failure.
   - It reduces the tension between the supplier and the client regarding detected non-compliance with requirements as failures are determined objectivity according to quantitative (objective) measurements. In contrast, in discussions relating to qualitative requirements each side raises arguments based on its (subjective) evaluations.
   - It enables better planning of staff and procedures based on productivity and reliability figures, etc. rather than on qualitative general phrases.

3. Software developer's arguments pro quantitative definition of requirements:
   - It allows performing quantitative measurements in order to objectively determine the probability for being compliant with the requirements. In cases where expected non-compliance is detected it is much easier and less expensive to deal with the necessary corrections at an earlier stage than during the client's acceptance tests.
   - It reduces the tension between the supplier and the client regarding detected non-compliance with requirements as failures are determined according to objective quantitative measurements.

CHAPTER 4

The components of the software quality assurance system – overview

Review questions

There are no review questions in this chapter.

Part II

Pre-project software quality components

CHAPTER 5

Contract review

Review questions

5.1

The CFV case is described at the beginning of this chapter. From the vice president's short speech, it can be understood that the proposal preparation was conducted as follows: (a) a negotiating team was appointed by the management, (b) a proposal was prepared by the negotiating team, (c) management approved the proposal before it was presented to the customer, (d) management signed the contract.

1. Can you suggest steps that would reduce the possible losses caused by a faulty contract.

2. What relevant contract review subjects, listed in Appendices 5A and 5B, could have revealed the contract faults described in the CFV case.

Solution

1. Performing contract review activities could have revealed the need to investigate the extent of required staff instruction activities. The recommended proposal preparation process:
   a. Both a negotiating team and a contract review team were appointed by the management,
   b. A proposal was prepared by the negotiating team,
   c. Review of the proposal draft by the contract review team,
   d. Management approved the proposal before it was presented to the customer,
   e. Review of contract draft presented by the customer to CFV, by both the negotiation team and the contract review team,
   f. Management signed the contract.

2. The relevant proposal review subject:
   - Section 1.6: clarification of the required training and instruction efforts.
   - Section 1.7: clarification of the number of installations to be performed and their location.

5.2

List the various aspects involved with the examination of the customer's capabilities.

Solution

Appendix 5A lists the customer capabilities as follows:

7.1 Financial capability, including contract payments and additional internal investments.

7.2 Supply of all the facilities, data and responses to staff queries as they arise.

7.3 Recruitment and training of new and existing staff.

7.4 Capacity to complete all task commitments on time and at the required quality level.

5.3

One of the objectives of a contract review is to examine development risks

1. List the most common types of development risks.
2. What proposal team activities are required regarding each of the revealed development risks.

Solution

Appendix 6A is dedicated to software development risks and software risk management.

1. The most common development risks, as listed by Boehm and Ross (see Table 6A.1) are:
   - Personnel shortfalls: Lack and turnover of qualified personnel
   - Unrealistic schedules and budgets: Incorrectly estimated (too low) development time and budget
   - Developing wrong software functions: Development of software functions that are not needed or are incorrectly specified
   - Developing wrong user interface: Inadequate or difficult user interface (GUI)
   - Gold plating: Addition of unnecessary features ("whistles and bells") due to professional interests, pride, or user demands
   - Continuing stream of requirement changes: Uncontrolled and unpredictable changes in system functions and features
   - Shortfalls in externally furnished components: Poor quality of externally delivered system components
   - Shortfalls in externally performed tasks: Poor quality or unpredictable accomplishment of externally performed tasks

- Real time performance shortfalls: Poor system performance
- Straining computer science capabilities: Inability to implement the system due to lack of technical solutions and/or computing power

2. A wide range of activities (termed "software risk management activities") suggested for prevention, early identification and resolution of development risk are listed in Table 6A.2 as follows:

a. Prevention activities

Internal RMA

- Application of detailed and thorough analysis to requirements and estimated schedules and costs
- Efficient project organization, adequate staff and team size
- Personnel training
- Arranging for and training replacements to take over in case of turnover and unanticipated workloads
- Arrange for user participation in the development process
- Applying efficient change control (change requests screening)
- Apply intensive software quality assurance measures such as inspections, design reviews, and benchmarking

Subcontracting RMA

- Preparing comprehensive and thorough contracts with subcontractors and suppliers, including contract reviews

Customer RMA

- Formulating comprehensive and thorough contracts with customers, including contract reviews

b. Early identification of software risk items (SRIs)

Internal RMA

- Periodic checking for timely availability of firm professionals currently occupied with other projects

Subcontracting RMA

- Participating in internal progress control and software quality assurance activities of subcontractors to be incorporated in the contract

c. Resolution of SRIs

Internal RMA

- Arranging for participation of professional staff members having knowledge and experience with SRIs
- Scheduling SRI-related activities as early as possible to provide leeway in case of difficulties
- Prototyping SRI-related modules or project applications
- Preparing scenarios for complicated SRI-related modules or project applications
- Simulating SRI-related modules or project applications

Subcontracting RMA

- Arranging for "loans" of professionals with specialized knowledge and experience if the need arises
- Hiring consultants to support the team in the absence of sufficient expertise and experience

Customer RMA

- Negotiating with the customer to change requirements re risky parts of the project
- Negotiating with the customer to change schedules re risky parts of the project

5.4

The extent of a contract review depends on the project's characteristics.

1. Describe an imaginary project that requires an intensive and comprehensive contract review.
2. Describe an imaginary project where a small-scale contract review would be adequate.

Solution

1. Police fingerprint identification system

   The system should perform the following functions:

   - Scan fingerprints of suspected persons by a scanner installed in every police patrol car.
   - Send the scanned fingerprints by a wireless communication system to the fingerprint identification center located at the police headquarters.
   - Search the fingerprint database and identify the "owner" (in cases where the suspect has already been included in the file).

   - Send to the police patrol car by wireless data communication the findings – three options: (a) no identification – the suspect was not identified, no record in the finger print library. (b) full identification – the suspect was identified. In this case, the response sent to the police patrol car includes the identification of the "owner" and the full police records relating to the "owner". (c) group identification – the system identified several persons with some similarities to the suspected person. The response sent to the police patrol car includes the identification of all the "possible owners" and the full police records related to each of the "possible owners".

   In this case a comprehensive contract review is justified, based on the project's characteristics, mainly: (a) new technology of pattern recognition is to be employed, (b) the need for an advanced data wireless ciphered communication system, (c) the system has to serve a large number of users, (d) capability to search an enormous database of police records, (e) the need for large-scale training activities and (f) a short response time is required.

   Performing a contract review raises many difficulties.

2. Development of an inventory management system based on a COTS software package by a contractor that had already completed numerous projects of this type.

5.5

Performing a contract review raises many difficulties.

1. List the "built-in" difficulties to carrying out a large-scale contract review.

2. List the steps that should be taken to make a large-scale contract review feasible.

Solutions

1. Three basic "built-in" difficulties are typical to contract reviews:
   - Time pressure.
   - The need to invest work highly professional staff, which is suitable to perform the highly professional review tasks.
   - The potential contract review team members are usually very busy, and can hardly spare time for the review.

2. The steps to be taken:
   - Activities of contract review should be scheduled and integrated into the proposal team schedule.
   - The workload should be divided among several team members to reduce the load of each of the members.

- A team leader should be appointed to coordinate the efforts of the team members and to ensure that a situation does not arise whereby no team member(s) could cause the failure of the team to keep to the schedule.

5.6

List those issues involved with estimating the resources required for a project that should be considered by the contract review team.

Solution

Based on Appendix 5A, estimations of resources for the following activities should be reviewed:

- Analysis, design, coding and testing
- Correction of errors detected in the above development phases
- Documentation preparations
- Providing services under warranty obligations
- Performing technical design reviews and inspections
- Installation, follow-up of the new software package running-in stage and training the client's staff.

5.7

List the supplier's capability issues that should be considered by the contract review team.

Solution

Based on Appendix 5A, the contract review team should examine the following supplier's capability evaluations presented by the proposal team:

1. Professional pool of knowledge.
2. Availability of specialized staff (on schedule and in the required numbers).
3. Availability of computer resources and other development (including testing) facilities (on schedule and in the required numbers).
4. Ability to cope with the customer requirements demanding use of special development tools or software development standards.
5. Warranty and long-term software maintenance service obligations.

5.8

List the partner and subcontractor participation issues that should be considered by the contract review team.

Solution

Based on Appendix 5A, the contract review team should examine the following evaluations regarding partner and subcontractor participation issues presented by the proposal team:

1. Allocation of responsibility for completion of tasks by the partners, subcontractors, or the customer, including schedule and method of coordination.

2. Allocation of payments, including bonuses and penalties, among partners.

3. Subcontractor payment schedule, including bonuses and penalties.

4. Quality assurance of work performed by subcontractors, partners and the customer, including participation in SQA activities (e.g., quality planning, reviews, tests).

Topics for discussion

5.1

MJS, Mount Jackson Systems, signed a contract to develop a comprehensive CRM (Customer Relations Management) system for a large food preparation corporation. In order to fulfill the project's requirements, MJS employed three subcontractors. MJS's experience with the subcontractors turned out to be troublesome, especially in regard to not keeping timetables, high rates of software faults of all kinds, and many interface faults with system parts developed by other participants in the project.

The head of the software quality assurance unit stated that if his unit had carried out the contract review procedure, most of the described problems would have been averted.

1. What contract review subject is relevant to this case?

2. What process would you recommend when applying a contract review in this case?

Solution

1. The contract review subject relevant to the discussed case is: subcontractor's participation conditions.

2. It is required that MJS proposal team would lead a negotiation process with potential subcontractors that will include definition of responsibilities for activities, binding schedules, payment schedules and quality assurance processes together with coordination methods between MJS and its subcontractors. The contract review

team should evaluate the draft of the proposed agreement. The agreements with the proposed subcontractors should be signed before MJS submits its project proposal to the food preparation corporation.

5.2

An SQA professional claims: "I find all the reasons given for a proposal draft review to be justified. I also believe that a review contributes to the quality of the proposal, especially in clarifying and precisely defining requirements, and in preparing more realistic estimates, among other issues. However, once the proposal has been presented to the customer, there is no need for a contract draft review. The task of reviewing the final negotiations results and the final version of the contract should be left to the legal department and to management."

1. Do you agree with the above statement? List your arguments.
2. In what situations is a contract draft review not necessary?
3. In what situations is a contract draft review absolutely necessary?

Solution

1. I don't agree with the above statement. The main arguments are:
   - Negotiations held between the supplier and the potential client after a proposal was submitted and prior to signing a contract are quite common. In many cases, the negotiations are not limited to the cost of the project and payment schedule but deal with changes and adaptations of requirements. In these situations, the contract draft review yields an additional independent evaluation of these changes, which may reveal undesirable consequences.
   - Even changes regarding bonus and penalty paragraphs which seem a financial issue may benefit from a contract draft review that would evaluate the prospect of the bonus and penalty situations.
   - In cases where a substantial period of time has elapsed between the submission of proposal and the contract signing, the supplier may expect meaningful changes on his side. Changes may refer mainly to development risks, to the company's capacity to perform the project, to the customer's capacity to fulfill his commitments and to subcontractors' commitments. A contract draft review in a situation of this nature is highly recommended.
2. A contract draft review is not necessary when:
   a. A contract draft review was part of the client's request for proposal.

   b. There were no client-supplier negotiations held after the proposal had been submitted.

   c. The contract was signed shortly after the proposal was submitted.

d. Limited negotiations regarding the project budget and its schedule were held before the contract was signed.

3. In situations of the nature mentioned in (1) a contract draft review is necessary.

5.3

Many organizations do not apply their contract review procedures to internal projects even though they perform comprehensive contract reviews for all their external projects.

1. List arguments that support this approach.

2. List arguments that oppose this approach.

3. Suggest types of internal projects where omission of a contract review could result in severe damage to the organization (mention the main components of damage listed for each project type).

Solution

1. Supporting arguments:
   - Internal projects are based to a great extent on personal relations among the IT department and the internal customer. Any formal steps may harm the "good will" that was created through the personal relations.
   - Performing a contract review is a costly process and causes undesirable delay in the commencement of the project.

- Anyway you can't punish an internal team or apply bonuses and penalty methods.

2. Opposing arguments:
   - A contract review means that some kind of agreement, in the form of minutes of a summary meeting or other form will be signed between the IT and internal customer departments.
   - A more formal relationship in performing internal project assures less loose ends; both sides gain from it.
   - It assures the more thorough preparation of the requirements document followed by an independent evaluation by the contract review team.
   - Performing a contract review assures more accurate resource estimates and schedules and a better understanding of the project's risks. So, the internal customer gains by getting a more realistic picture of the planned project.

3. Internal projects where the completion time is critical (e.g., software package for management of schools that misses the opening of the new school year, or a new computer game that misses the high sales period of the new year's holidays) may serve as an example of internal project types that may benefit from thorough

contract review. Typical tasks undertaken by internal customers and later found incapable of performing are training and documentation commitments. Evaluation of the actual capability of the internal customer by a contract review team might produce a better organization for carrying out these activities (and save the heavy damages that might have occurred).

5.4

One of the objectives of a contract review is to examine the customer's capability to fulfill his commitments. Accordingly, a comprehensive list of contract review subjects is suggested in Appendix 5A. Some managers believe that because the supplier can sue the customer in case he does not fulfill his commitments, there is no justification to invest resources in reviewing the customer's capabilities.

1. Do you agree with these managers?

2. If you disagree, list your arguments in favor of a comprehensive examination of the customer's capabilities.

3. Can you describe a real or imaginary situation where a customer's capability failures *could* create substantial direct and indirect damages to the software developer ("the supplier").

Solution

1. I disagree with the managers.

2. According to Appendix 5A, the customer's incapability might be financial or one that causes a delay in completion of the project.

   In cases of financial difficulty of the customer, and especially when it is permanent (customer's bankruptcy), no remedy is obtained by suing the customer. In the case of the supplier's major project, even a customer's temporary financial problem could be responsible for the supplier being unable to fulfill his financial commitments to his subcontractors, employees, etc. Usually legal means cannot compensate the supplier for its extra expenses and for required efforts and indirect damages.

   In situations where the customer fails to comply with the defined schedules or to carry out tasks he is responsible for in a competent manner, the direct losses are mainly due to the need to invest extra work, or staff being underemployed. Even if these direct losses are fully covered by contract articles, it rarely compensates the supplier for all his losses. Especially not covered are indirect losses in other projects due to late release of the project's team members scheduled to participate in other projects. Other indirect losses are due to a decline of the supplier's reputation, caused by late completion of software development projects (potential customers are influenced by the fact of late completion and usually disregard the question of who is responsible for it).

3. Let us refer to a software package developed for a hotel network, located in the US and in several overseas countries. The customer was committed to perform all training tasks in the US and overseas. The customer's task included preparing training course material in English, translating it into Spanish, French and German, and delivering the training courses to a total of 4,300 personnel members, 60% of them located oversees.

   At the time the software development process was completed, and the installation process was due to begin, only 30% of the US and 15% of the British hotel personnel had completed the training courses. The French, Spanish and German training material was still "in process". A delay of 6 months in the completion of the installation and running-in of the software package meant that the hotel chain could only benefit from its IT investment in the next tourist year. The main damage to the supplier, not covered by the contract compensation articles, was the delay in the release of team members to join their next assignment, causing severe delays to other projects.

5.5

A contract draft review of a properly prepared contract document is expected to yield no negative findings. Still, in reality, discrepancies in contracts do appear frequently.

1. List real cases and common situations where such discrepancies could arise.

2. In what situations are discrepancies in the contract draft expected to be least likely?

Solution

1. The following discrepancies may occur:
   - Some of the understandings reached in negotiations with the customer after the submission of the proposal are not included or presented incorrectly.
   - Some additions, which the customer considers negligible, are included without being discussed with the supplier. Additions of this nature may change conditions for bonuses, penalties or timetables for handling the project's products.

2. Discrepancies in the contract draft are not likely to be expected in cases where no negotiations were held in the period between submission of the proposal and signing the contract or in cases where the contract draft was part of the tender document, allowing no changes or additions to be performed.

5.6

The examination of alternatives is one of the major tasks of a proposal team, especially for tender proposals. However, in many cases, important alternatives are omitted or neglected by the proposal team.

1. List real cases and common situations where negligence in defining and examining important alternatives can be expected.

2. In what situations are these types of discrepancies least likely to occur?

Solution

1. Common situations:
   - A proposal is being prepared under high time pressure that leaves no time for negotiations with partners or subcontractors or to develop a new methodology.
   - A proposal is being prepared after a series of successful projects of a similar nature. In such a situation, a proposal team tends to follow an early success route and may not check alternatives.

2. These discrepancies are not expected to occur in cases of preparing a proposal where:
   - The subject matter is new and/or
   - The proposal refers to a large extent project and/or
   - The proposal team has the necessary time to examine alternatives and/or
   - The search for alternatives is part of the methodology (and procedure) of preparing proposals.

5.7

National Software Providers is very interested in the newly developing area of BI (Business Intelligence) for electronic commerce firms. As the company is very keen to gain experience in this area, they were especially interested in winning a tender issued by one of the leading cosmetics manufacturers. The proposal team estimated that in order to win the contract, their proposal should not exceed the sum of $650,000. Accordingly, their quotation was $647,000. As all the team members were aware that the cost of completing the project by the company's inexperienced development department would substantially exceed this sum, they decided that there was little use in investing efforts to estimate the actual costs of the project.

1. Do you agree with the team's decision not to estimate the project's costs.

2. If you disagree, what are your arguments in favor of estimating the costs?

Solution

1. No. I do not agree.

2. The arguments in favor of estimating the project costs are:

- With no cost estimation there is no project budget, so that management loses an important means for controlling the project's progress.
- With no project budget one of the main tools to evaluate the project manager's success is not available.
- No cost estimation means no resources estimation, the project manager has to prepare his own resource estimates in order to schedule his project to complete it at the required target date. According to this scenario, the expertise of the proposal team and the contract review team in preparing appropriate resource estimates do not support the project manager in this task, allowing for frequent cases of substantial underestimates (late completion) or overestimates (waste of resources) to occur.

5.8

Consider the case of a custom-made software package developed by a supplier according to the unique RFP (request for proposal) specifications of the customer.

1. What proprietary issues are expected in such a project?
2. What security issues related to the proprietary rights listed in your answer to (1) should be examined.

Solution

1. Typical proprietary issues of a custom-made software package:
   - Proprietary issues of reused software units or modules integrated in the new software package.
   - Proprietary issues of the completed new software package for resale to other organizations: the customer or the developer.
   - Proprietary issues of reuse of original units or modules developed in the project.
2. Issues of unlawful reuse of software units or modules by the developer during the development process as well as unlawful use of the completed software package or its units or modules.

5.9

Contract review subjects include a variety of financial issues.

1. Why should an SQA activity such as contract review to be so heavily involved in financial issues.
2. Is it likely that an SQA unit member will be able to review the financial issues? Who do you believe should do it, and how should the review be organized?

Solution

1. It is clear that if the project fails financially, the prospects of full implementation of quality assurance tools and processes are very meager. In this situation it is likely that the client will face many project failures, corrections of which will increase substantially the project's losses. It should be mentioned that the financial issues include those of the contract with the customer and those of contracts with subcontractors and partners.

2. The financial issues may be handled by a team member possessing the adequate financial education and experience or by an outside consultant.

5.10

A contract review can be performed by "insiders" (members of the organization's proposal team or other staff members) or by "outsiders".

1. What are the advantages and disadvantages of employing outsiders compared with insiders for a proposal draft review?

2. What are the advantages and disadvantages of employing outsiders compared with insiders for a contract draft review?

Solution

1. Advantages:
   - Outsiders assure more independence of reviews.
   - Outsiders may be chosen to possess experience and knowledge not available within the organization.

   Disadvantages:
   - Outsiders may require some time to become acquainted with the organization unit in which the proposed project is to be performed and its environment. Despite the efforts invested, an outsider rarely reaches full understanding of the unit and its surroundings.
   - There are always some security risks in employing outsiders and allowing them access to detailed information of the organization's future information systems.

2. The advantages of an outsider in the contract draft are much less than in the proposal draft stage, and refer mainly to his personal capabilities to check thoroughly the contract documents and his availability to do the task (in cases of unavailability of an insider to do the task). The disadvantages of an outsider at this stage are similar to those mentioned for the proposal draft stage of the contract review.

5.11

A medium-size firm submits 5–10 proposals per months, 10%–20% of which eventually evolve into development contracts. The company takes care to perform a thorough project draft review for each of the proposals.

1. Do the proposal draft reviews performed for each of the individual projects guarantee that the company will be capable of carrying out all the proposals that eventually evolved into development contracts? List your arguments.

2. If your answer to (1) is negative, what measures should be taken to reduce the risk of being unable to perform the contracts.

Solution

1. The issue of medium-size firms raised here is a classic problem. A firm may face a period of much higher rate of success in its proposal, creating a need for development resources, much higher than are available. The risk of being unable to perform project proposed is a real one, even if measures are taken to reduce its prospects and extent by negotiating with possible partners or subcontractors.

2. The measures to be taken:
   - Submission of proposals with partners.
   - General agreement with subcontractors and consultants to provide support in needed resources where there is a shortage of the firm's resources.
   - Specific agreements with subcontractors and consultants to join the project in case of need. These agreements are of special importance regarding specialized resources being scarce amongst the firm's personnel.

CHAPTER 6

Development and quality plans

Review questions

6.1

Significant similarity exists between the proposal draft review and the development plan.

a. Compare these documents with reference to the subjects reviewed.

b. Compare these documents while referring to the aim of preparing the individual documents.

Solution

a. The following table, based mainly on Appendix 5A and Table 6.1, summarizes the similarities and differences between the contract review and development plan documents. The subjects reviewed in a proposal draft review are grouped according to objectives. The comparison shows that a much wider range of subjects is reviewed by the proposal draft team than by the project plan team. The 11 software development plan elements relate to only 5 of the 9 proposal draft subject groups. Four of the proposal draft review subject groups have no parallel element in the development plan document.

Contract review subject groups (Proposal draft review)	**Development plan elements**	**Comments**
1. Customer requirements have been clarified and documented	None	
2. Alternative approaches for carrying out the project have been examined	a. Project's methodology and development tools b. Software development standards and procedures	The development plan relates to the specified alternative
3. Formal aspects of the relationship between the customer and the software firm have been specified	None	
4. Identification of development risks	a. Development risks and risk management actions b. Control methods	

Contract review subject groups (Proposal draft review)	**Development plan elements**	**Comments**
5. Adequate estimation of resources and timetable	a. Map of the development process b. Project milestones c. Project products, specifying “deliverables” d. Project interfaces e. Required development facilities f. Project staff organization g. Project cost estimates	The development plan is much more detailed than the proposal draft review
6. Examination of the firm’s capacity to perform the project	Project staff organization	
7. Examination of customer’s capacity to fulfill his commitments	None	
8. Definition of partner and subcontractor participation conditions	Coordination with external participants	
9. Definition and protection of software proprietary rights	None	

b. The aims (objectives) of the proposal draft review document are to examine the prospects in a wide range of subjects involved in a future project, while the development plan focuses on a smaller range of subjects and deals mainly with materializing plans sketched in the proposal stage.

The comparison of objectives based on Frame 5.1 and Section 6.1 is presented in the following table.

Proposal draft review objectives	**Development plan objectives**
1. Customer requirements have been clarified and documented	None
2. Alternative approaches for carrying out the project have been examined	None
3. Formal aspects of the relationship between the customer and the software firm have been specified	None
4. Identification of development risks	a. Resolving development risks b. Providing management with data needed for project control

Proposal draft review objectives	Development plan objectives
5. Adequate estimation of resources and timetable	a. Scheduling development activities (including those to be carried out by external participants) that will lead to the successful and timely completion of the project, and estimating the required manpower resources and budget. b. Recruiting team members and allocating development resources (according to activity schedules and manpower resource requirement estimates)
6. Examination of the firm's capacity to perform the project	Recruiting team members and allocating development resources (according to activity schedules and manpower resource requirement estimates)
7. Examination of customer's capacity to fulfill his commitments	None
8. Definition of partner and subcontractor participation conditions	None
9. Definition and protection of software proprietary rights	None

6.2

Development and quality plans have five objectives.

1. Can you list the objectives?
2. Suggest ways in which each objective contributes to the successful and timely completion of the project.

Solution

1. The objectives, as listed in Section 6.1 are:
    1. Scheduling development activities (including those to be carried out by an external participants) that will lead to the successful and timely completion of the project, and estimating the required manpower resources and budget.
    2. Recruiting team members and allocating development resources (according to activity schedules and manpower resource requirement estimates).
    3. Resolving development risks.
    4. Implementing required SQA activities.
    5. Providing management with data needed for project control.

2. The main contributions are:
   - **Scheduling development activities and estimating the required manpower resources and budget** – adaptation of schedules prepared in the proposal stage to the contracted commitments and updating the project resource tables and budget, reduces cases of delays and time pressures during the development process.
   - **Recruiting team members and allocating development resources** – assures the availability of the planned resources for the project, and consequently reduces situations of team member shortage.
   - **Resolving development risks** – early efforts in these matters may elevate problem resolution and decrease difficulties in the progress of the development process.
   - **Implementing required SQA activities –** it can be expected that integrating SQA activities into the development plan and organizing their implementation will contribute to reduce software errors, and consequently reduce the costs involved in corrections.
   - **Providing management with data needed for project control –** development plan and especially the activity schedule and risk management plans and activities provide management with a more thorough and effective follow up of development activities, allowing for earlier detection of project difficulties.

6.3

Some development elements are included in the requirement document, yet are not compiled by development planners.

1. Which elements of the development plan belong to this category?

2. Explain the importance of gathering this information from the customer's documents

6.3a (alternative and preferred wording for 6.3)

Some system analysts claim that requirements that belong to certain software quality factors (see Chapter 3) should not be considered when preparing a project development plan.

1. Do you agree with this claim? If yes – list the software quality factors that should not be considered.

2. If not – explain your arguments.

Solution

1. Actually all the elements of the requirement document have to be considered when compiling the development plan. None of the requirements belonging to the various software quality factors should be neglected or left untreated when compiling the activity schedule for the project.

2. Obviously, disregarding part of the requirements in the development plan means this requirement will not be fulfilled in the development of the software product.

6.4

Development process mapping is one of the most important elements of the development plan.

1. List the possible phases of the development process.

2. List possible inputs and outputs for each of the phases suggested in (1).

3. What components of each activity, as associated with each project phase, should be described in the development plan?

Solution

1. The possible phases according to the SDLC development model (see Section 7.1) are the following:
   - Requirements definition
   - Analysis
   - Design
   - Coding
   - Testing
   - Installation and conversion

2. Typical inputs and outputs of each phase are listed in the following table:

Development phase	**Inputs**	**Outputs**
Requirement definition	-----	a. System requirement document. b. Software requirement document.
Analysis	a. System requirement document. b. Software requirement document.	a. Preliminary design document. b. Software test plan.

Note: Operation and maintenance phase is not a development phase.

Development phase	**Inputs**	**Outputs**
Design	a. Preliminary design document. b. Software test plan.	a. Critical design document. b. Software test procedure. c. Software installation plan. d. Software maintenance manual.
Coding	Critical design document.	a. Software test report. b. Project code (complete version prepared for system tests).
Testing	a. Software test plan. b. Software test procedure. c. Project code (complete version prepared for system tests).	a. Tested and corrected version of the project code (approved for installation). b. Software user manual.
Installation and conversion	a. Tested and corrected version of the project code (approved for installation). b. Software installation plan.	

3. The components of each activity that should be described in the development plan are the following:
   - An estimate of the activity's duration.
   - The logical sequence in which each activity is to be performed, including a description of each activity's dependence on previously completed activities.
   - The type of professional resources required and estimates of how much of these resources are necessary for each activity.

6.5

The project's organization is an important element of the development plan.

1. List the components of the organization element.

2. Which of the components in (1) are based on components of project mapping?

3. Why is it necessary to mention team members by name? Isn't it sufficient to list the number of team members by their expertise as required for each phase of the project?

Solution

1. The components of project staff organization are:
   - Organizational structure: definition of project teams and their tasks, including teams comprised of a subcontractor's temporary workers.
   - Professional requirements of team members, including professional certification.
   - Number of team members required for each period of time.
   - Names of team leaders and team members.
2. The professional requirements and the number of team members are based on the project mapping. The organizational structure also depends to a great extent on the project mapping.
3. Defining the names of team members allows them to be assigned ahead for performing given tasks, thus assuring their availability for the project. It also permits the tracking of their actual availability, and assures early detection of difficulties regarding their unavailability.

6.6

Boehm and Ross (1989) mentioned 10 major software risk items.

1. Can you list the 10 SRIs?
2. For each of the SRIs mentioned in (1), suggest the three most effective RMAs for handling them (refer to Table 6A-2). Explain your choice.

Solution

1. The SRIs are listed in the following table (See Table 6A.1).
2. The suggested RMAs contribute to prevention, early identification and resolution of the SRIs, and are listed in the following table:

Item No.	**Software risk item (Boehm and Ross)**	**Risk management actions (RMAs)**
1	Personnel shortfalls	a. Efficient project organization, adequate staff and team size b. Arranging for and training replacements to take over in case of turnover and unanticipated workloads c. Arranging for "loans" of professionals with specialized knowledge and experience if the need arises

Item No.	**Software risk item (Boehm and Ross)**	**Risk management actions (RMAs)**
2	Unrealistic schedules and budgets	a. Application of effective contract review procedures b. Application of detailed and thorough analysis to requirements and estimated schedules and costs c. Negotiating with the customer to change schedules re risky parts of the project
3	Developing wrong software functions	a. Application of detailed and thorough analysis to requirements and estimated schedules and costs b. Apply intensive software quality assurance measures such as inspections, design reviews, and benchmarking c. Arrange for user participation in the development process
4	Developing wrong user interface	The RMAs suggested for item 3 apply also here. a. Application of detailed and thorough analysis to requirements and estimated schedules and costs b. Apply intensive software quality assurance measures such as inspections, design reviews, and benchmarking c. Arrange for user participation in the development process
5	Gold plating	a. Application of detailed and thorough analysis to requirements and estimated schedules and costs b. Arrange for user participation in the development process c. Negotiating with the customer to change requirements re risky parts of the project (cancel undesired and unnecessary added functions)
6	Continuing stream of requirement changes	a. Applying efficient change control (change requests screening) b. Apply intensive software quality assurance measures such as inspections, design reviews, and benchmarking c. Efficient project organization, adequate staff and team size (to cope with approved requirement changes)
7	Shortfalls in externally furnished components	a. Preparing comprehensive and thorough contracts with suppliers, including contract reviews b. Applying project progress control of the fulfillment of suppliers' commitments c. Negotiating with the customer to change schedules re risky parts of the project

Item No.	Software risk item (Boehm and Ross)	Risk management actions (RMAs)
8	Shortfalls in externally performed tasks	a. Preparing comprehensive and thorough contracts with subcontractors and suppliers, including contract reviews b. Participating in internal progress control and software quality assurance activities of subcontractors to be incorporated in the contract c. Apply intensive software quality assurance measures such as inspections, design reviews, and benchmarking
9	Real time performance shortfalls	a. Prototyping SRI-related modules or project applications b. Hiring consultants to support the team in the absence of sufficient expertise and experience c. Negotiating with the customer to change schedules re risky parts of the project
10	Straining computer science capabilities	a. Application of detailed and thorough analysis to requirements and estimated schedules and costs b. Hiring consultants to support the team in the absence of sufficient know-how and experience c. Negotiating with the customer to change requirements re risky parts of the project

6.7

Only 4 out of the 11 elements of a development plan and only 1 out of 5 of the quality plans are considered obligatory for small projects.

1. Do you agree with this choice. If yes – list your main arguments.

2. If you do not agree with this choice, present your improved list and explain your choice.

Solution

1. I agree with the choice. The elements specified minimize the efforts required to prepare development and quality plans, whilst at the same enable the following: carrying out of the required activities necessary to ensure that adequate and timely resources are utilized, performing software risk management activities and creating a basis for project process control.

   The rest of the elements in the case of small (and relatively simple) projects are expected to be part of the organization procedure and work instructions.

2. See answer to (1).

6.8

“Preparing full-scale development and quality plans for internal projects, and applying regular full customer–supplier relationships for the implementation of internal projects is highly beneficial to both sides”.

1. Explain the benefits of these procedures to the developer.

2. Explain the benefits to internal customers.

Solution

1. **The developer** can enjoy the following benefits of plan preparation:
   a. Allowing efficient control of a project’s progress.
   b. Prevention of most development risks and early detection of most other risk situations.
   c. Avoiding budget overruns.
   d. Avoiding damage to other projects caused by delays in release of professionals occupied in an internal project.
   e. Avoiding loss of market status, especially regarding the firm's reputation, caused by delayed completion of external projects triggered by late completion of internal projects.

2. **Internal “customers”** can enjoy the following benefits:
   a. Better coordination with the developer, based on the development and quality plans, yields better fulfillment of project’s requirements – higher satisfaction of the internal customer.
   b. Smaller deviations from planned completion dates and smaller budget overruns.
   c. Better control over the development process, including earlier identification of possible delays assisting in the search for and resolution of their causes.
   d. Fewer internal delay damages.

Topics for discussion

6.1

“As long as the proposal was properly prepared and approved, following an adequate contract review, there is no justification for redoing all this work. Its resource estimates and schedule can serve as the project’s plan....” You often hear claims like this one.

1. Do you agree with this claim? If not – list your arguments against it.

2. Suggest situations where it is clear that the proposal and its materials can serve as development and quality plans.

3. Suggest situations where it is clear that the proposal and its materials cannot serve as development and quality plans.

Solution

1. I do not agree. The arguments:
   - The plans examine possible changes that have occurred since the time the proposal was submitted.
   - The plans examine the actual up-to-date availability of team members, subcontractor's participation and laboratory resources, etc. available to the adapted project schedule.

2. Situations where it is clear that the proposal and its materials can serve as development and quality plans:
   - "Simple" small to medium-size projects, with no external participants.
   - Projects where only a short time period has elapsed since the time the proposal was submitted.

3. Situations where it is clear that the proposal and its material cannot serve as development and quality plans:
   - Complicated or large scale projects, especially those involving external participants, where a substantial time has elapsed since the submission of the proposal.
   - Projects where substantial changes of requirements result from negotiations that were held with the customer in the interim period after the proposal had been submitted and before signing the contract.

6.2

Martin Adams, an experienced project leader at David's Software Ltd., a medium-size software house, has been appointed project leader for development of an advanced help desk software system for a leading home appliance maintenance service. This is the 12th help desk system developed by his department in the last three years.

The current project is somewhat special with respect to its timetable. The contract with the customer was signed 6 days after submission of the proposal, and the development team is scheduled to begin working at full capacity, with 8 team members, 10 days later. The contract offers a significant early completion bonus for each week below 26 weeks, but determines high late completion penalties for each week after 30 weeks.

In a meeting with his superior, Adams claims that the comprehensive proposal documentation “as is”, which has been thoroughly checked by the contract review team, should serve as the project's development and quality plans. His superior does not agree with him and demands that he immediately prepare comprehensive project and quality plans, according to company procedures.

1. Do you agree with Adams? If yes – list the arguments that support his claim.

2. Do you agree with his superior? If yes – list the arguments that support the superior’s claim.

3. Considering the circumstances of the project, what, in your opinion, should be done in this case.

4. Comparing the circumstances described here to those of the opening anecdote, are there any justifications for different recommendations?

Solution

1. The scenario of the project basically supports the claim of Martin Adams, and the proposal material could serve as part of the development and quality plans. However, the project manager has to prepare some elaborations and additional details to complete the development and quality plans.

2. I agree partly with the superiors regarding the aspects involved with the short completion time and the bonuses and penalties involved. Accordingly, special care is required in examination of availability of any kind of resources, including external participants, and in resolving development risks. The level of these examinations should be beyond the ones performed for the proposal.

3. The recommended actions are already marked in the answers to (1) and (2).

4. According to the case presented at the beginning of the chapter a period of seven months has elapsed since the proposal was submitted, compared with few days in the above case. This very crucial difference justifies preparing new plans based on full updating of the proposal material. In our case, partial updating of proposal material should be sufficient.

6.3

An experienced project leader has identified 6 SRIs inherent in his project and estimated their Est(dam) and Prob(mat). The results are listed in the following table:

No.	SRIs	Prob(mat)	Est(dam) $
1	Networking at the customer's 23 sites will not be completed on time	0.2	150,000
2	Subcontracted modules will fail the acceptance tests	0.5	12,000
3	The programming team will be 2-3 programmers short for more than 2 months	0.7	50,000
4	The software quality assurance activities will fail to detect major software errors in the complicated discount module; these errors will be discovered by the customer during the guarantee period	0.05	600,000
5	The final test of the user's guide will detect significant errors that will cause a delay of more than 2 weeks in delivery to the customer	0.3	2,500
6	The planned server's capacity will be found insufficient in the final system tests	0.25	40,000

1. Determine the priorities for these SRIs, using the formula given in Section 6A.3.

2. Can you suggest an alternative method for prioritizing the SRIs?

3. Determine the SRI priorities according to the alternative method. Compare the resulting priority list with that obtained in (1), and discuss the implications of the differences, if any.

Solution

1. The results for SRIs priorities, according to the Exp(risk), are presented in the following table:

No.	SRIs	Prob(mat)	Est(dam) $	Exp(risk)	Exp(risk) Priority	Est(dam) priority
1	Networking at the customer's 23 sites will not be completed on time	0.2	150,000	30,000	2–3	2
2	Subcontracted modules will fail the acceptance tests	0.5	12,000	6,000	5	5
3	The programming team will be 2-3 programmers short for more than 2 months	0.7	50,000	35,000	1	3

No.	SRIs	Prob(mat)	Est(dam) $	Exp(risk)	Exp(risk) Priority	Est(dam) priority
4	The software quality assurance activities will fail to detect major software errors in the complicated discount module; these errors will be discovered by the customer during the guarantee period	0.05	600,000	30,000	2–3	1
5	The final test of the user's guide will detect significant errors that will cause a delay of more than 2 weeks in delivery to the customer	0.3	2,500	750	6	6
6	The planned server's capacity will be found insufficient in the final system tests	0.25	40,000	10,000	4	4

2. An alternative method to define SRI priorities is based on the estimated damage Est(dam). According to this method, the organization should consider the actual damages it may face and not expected values which are just a statistical measure to compare situations.

3. The priorities defined according to this method are presented in the right column of the above table.

   Application of the alternative method:

   – Calls to focus on SRI4 and invest any possible measures to reduce its probability to materialize and especially its estimated damage.

   – Changes the internal order of priorities of the SRIs defined to priorities 1–3 in section (1).

   – No changes of priority are found for the SRIs defined to priorities 4–6 in section (1).

6.4

It is said that three of the quality plan's elements must be coordinated with an element of the development plan – the mapping of the development process.

1. Can you identify these elements?

2. Explain the nature of the required coordination.

Solution

1. The following quality plan's elements must be coordinated with the mapping of the development process
   - Review activities
   - Software tests
   - Configuration management plans

2. The schedule of review and software activities should be coordinated with the schedule for completion of the software products (documents and code). At the same time correction, activities of errors identified in the reviews and tests and the continuation of development process should be coordinated with the review and test activities.

   The configuration management activities and especially the release of planned baseline versions of the software system must be coordinated with the progress of the development process and the completion of the production of the relevant software products.

6.5

Quoting from Section 6.3: "Quantitative measures are usually preferred to qualitative measures when choosing quality goals because they provide the developer with more objective assessments of software performance during the development process and system testing. However, one type of goal is not totally equivalent to the other."

1. How are quantitative goals applied during the development process?

2. Explain in what way quantitative goals enable more objective evaluation of performance when compared with qualitative goals.

Solution

1. Quantitative goals can be used in pre-test, which enable the developer to measure the current performance and decide about the maturity of the development efforts. For example: A quantitative goal of 99.8% accuracy is set for a computerized input device. In pre-test, the developer can objectively evaluate the current achievements.

2. Qualitative goals, such as "friendly", "efficient" or "easy to train" are open to wide range of evaluations, where usually the developer expresses his satisfaction while the customer is less enthusiastic or even dissatisfied. Situations like this, of disagreements between developer and customer, are not desired. Comparable quantitative goals are much less debatable. Usually the application of an additional test sample will resolve the disagreements.

Part III

SQA components in the project life cycle

CHAPTER 7

Integrating quality activities in the project life cycle

Review questions

7.1

Referring to the SDLC model:

1. What are the seven basic phases of the development process suggested by the model?

2. Suggest situations where the number of process phases should be reduced.

3. Suggest situations where the number of process phases should be increased.

Solution

1. The seven SDLC phases, as presented in Fig. 7.1 are:
   - Requirements definition
   - Analysis
   - Design
   - Coding
   - System tests
   - Installation and conversion
   - Operation and maintenance

2. In small-scale projects the phases of analysis and design may be integrated into one phase.

   In projects based on the implementation of a COTS software system – the coding phase is not required. In many cases the analysis and design phases may also be integrated.

3. In large-scale projects, the analysis and design phases need to be split into preliminary and critical stages. The requirement definition phase may be divided into system and software requirement definition phases.

7.2

With respect to the prototyping methodology:

1. List the conditions necessary for the prototyping model to be applied.

2. Can you suggest an imaginary project ideally suitable for the prototyping methodology?

3. Can you suggest an imaginary project that is obviously unsuitable for the prototyping methodology?

Solution

1. Conditions for successfully applying the prototyping model:
   - The project is planned to serve well-established services, where no major change is required
   - The project is relatively small and not complicated.
   - A group of users capable of evaluating prototype versions of the project are available.

2. An independent store (not affiliated to any chain), currently employing a computerized system covering part of the store functions and a manual system for the rest, decides to order a more comprehensive information system. The staff of the store is well acquainted with all the business processes, knowledgeable and well acquainted with the business processes and capable to evaluate the prototype versions.

3. An information system is planned to serve a new organization, with a new, relatively inexperienced staff. In addition, there is no other organization capable of supplying operational expertise to the new organization.

7.3

Comparing the SDLC and prototyping methodologies:

1. List the advantages of the prototyping compared to the SDLC methodology for development of small to medium-sized projects.

2. Explain why the advantages of prototyping cannot be realized for large software systems.

3. In what ways can prototyping support the development of large-scale projects?

Solution

1. The main advantages:

A shorter development process, due to the precise guidance of the user evaluation group.

Substantial savings of development resources (man-days), due to adherence to the precise requirements.

A better fit to customer requirements and the reduced risk of project failure.

Easier and speedier user comprehension of the new system.

2. The advantages of prototyping cannot be realized for the development of large scale systems for the following reasons:
   - Large scale systems involve, beside "common" modules, the development of new applications of high complexity, based on intense interfacing with other systems and application of advanced methodologies and algorithms, which are designed to solve new functional requirements. Most of the development efforts are invested in these new modules. Usually users' ability to contribute to these parts of the project is very limited.
   - Large scale systems are developed to include a wide scope of future new applications of the system. Naturally, users are not experienced in these new directions and are unable to provide substantial contributions.
   - Large scale system developers invest a substantial part of their efforts in solving security and reliability problems of the system. The ability of users to contribute in these aspects is usually limited.

3. Prototyping may support the development of large scale systems in limited areas, for the development of specific modules. Users may contribute to the development of the output modules that are planned to serve them, or input modules that they are expected to use.

7.4

Referring to the spiral model:

1. Describe the four activities to be repeated in each iteration of the development process. Explain why the four activities designated are to be repeated in each iteration of the development process.

2. What new activities were added to the classic SDLC model and what is their main contribution to the success of the projects?

Solution

1. The four activities to be repeated are:

   Planning

   Risk analysis and resolution

Engineering activities according to the stage of the project: design, coding, testing, installation and release

Customer evaluation, including comments, changes and additional requirements, etc.

The reasons for repeating these activities are:

Planning – updating the project plans according to customer's comments regarding the former iteration of the project and according to new and updated requirements presented by the customer. This activity assures relative up-to-datedness of project's plans.

Risk analysis – once the project's plans have been updated, updated risk analysis takes place, evaluating the consequences of new risks identified during the last project iteration and of changes introduced in the project's plans.

Engineering activities – professional activities of the project team.

Customer evaluation – assures improved fulfillment of customer requirements and saves development efforts invested in erroneous directions.

2. The activities added to the classic SDLC model are:
   - Customer's evaluation
   - Developer design and developer's construction, replacing the one engineering activity to be performed in each project's iteration.

   The contribution of adding the customer's evaluation activity is by stressing the importance of customer's involvement in the project and especially his professional evaluation of each iteration of the project.

   The contribution of replacing the one engineering activity performed in each project's iteration is by introducing the principle of the need to perform design or updating design in each project's iteration. It assures improved suitability to customer's updated requirements. This principle is of special importance to projects developed according to a prototyping development model.

7.5

Comparing the SDLC and spiral models:

1. Explain the advantages of the spiral model as compared with the SDLC model.
2. What characteristics of a project enable these advantages to be best realized?
3. Provide three examples of projects that would obviously benefit from application of the spiral model.

Solution

1. The SDLC model is a linear model, where the development activities are successive and allow for only marginal expression of the iterative nature of the development process. This attribute makes SDLC an ideal basis for project planning.

The advantages and contribution of the spiral model is in the presentation of the internal vital component of each phase:

- The emphasis of the need for the customer's thorough involvement in the development process expressed by comments and updated requirements on each iteration.
- The emphasis of risk management activities incorporated in each iteration.
- The need to perform necessary design and coding in each iteration (presented in the advanced spiral model).

2. The above model advantages are best realized in large-scale models involving additional development of a complicated nature for a major part of the project (only a small percentage of its software is reused).

3. 1. Development of an advanced medical application for the intensive care unit.

   2. Development of a real time software system to serve a radar system based on a new methodology.

   3. Development of an advanced system based on fingerprint library files and police archives to serve the Police Forensic Department.

7.6

With respect to verification, validation and qualification:

1. Explain the differences between these three aspects of SQA activities.

2. Can a project that successfully passed verification and validation reviews but failed part of the qualification review adequately supply users with the information needed? Explain your answer.

3. In which respects is the project described in (2) inferior to a project that passed all three reviews? In what way will this difference affect operation of the software system?

Solution

1. IEEE Std 610.12-1990 (IEEE, 1990) defines these aspects as follows:

   "**Verification** – The process of evaluating a system or component to determine whether the products of a given development phase satisfy the conditions imposed at the start of that phase."

   "**Validation** – The process of evaluating a system or component during or at the end of the development process to determine whether it satisfies specified requirements."

   "**Qualification** – The process used to determine whether a system or component is suitable for operational use."

While verification focuses on detecting incorrect implementation or deviations from the results of the former phase of software development, validation deals with identifying incorrect implementation or deviations from the functional specifications and other requirements defined for the project by the customer. Qualification emphasizes the detection of deviations from the work orders of the organization and from the standards of analysis, design and coding prepared by the development team, regardless of whether the software is correct or incorrect, according to its verification and validation tests.

2. It's possible for tested software to achieve correct functioning (no verification and validation errors) to fail qualification tests when the developed programs are correct but are not in accordance with the standards, procedures and work procedures of the organization. It is typical for new staff or subcontractor's staff not to be fully knowledgeable of these organizational requirements. In some cases, even experienced staff under time pressures may disregard standards, procedures and work instructions in order to save time.

3. Deviations from the software development standards, procedures and work instructions of the organization may cause difficulties in testing the software for replacement staff assigned to continue the work of a team member who left in the middle of the project.

   It will also cause difficulties for the maintenance team performing corrections, changes and additions to the software package.

7.7

Theoretically, verification reviews should be sufficient. Still, SQA professionals recommend performance of validation and qualification reviews as well.

1. What do they expect to gain by adding a validation review?

2. What do they expect to gain by adding a qualification review?

Solution

1. Validation reviews and tests evaluate whether the current stage meets the specified requirements, and thus assures better fulfillment of the customer's requirements, leading to greater satisfaction of the customer. The validation activities are expected to detect deviations from the original requirements, not detected by verification activities, that may accumulate to substantial deviations throughout the development phases, especially towards the end of the development process.

2. Qualification tests are aimed at detecting deviations from the development standards, procedures and work instructions as well as from sections of the software system of low quality (even if the processing and calculation results are correct). Software documentation and code that qualify save many of the resources required for corrections, changes and additions to the package and especially reduce the probability of errors in understanding the current software system.

7.8

Referring to the model for defect removal efficiency and costs:

1. What assumptions rest at the foundations of the model?

2. Which three of the model's data components are based on published survey results?

Solution

1. The model is based on the following six assumptions (see Section 7.4.2):

   The development process is linear and sequential, following the Waterfall Model.

   A number of "new" defects are introduced in each development phase.

   Reviewing and testing software quality assurance activities serve as filters; they remove a percentage of the entering defects and enable the remaining defects to pass on to the next development phase. For example, if the number of incoming defects is 30 and the filtering efficiency is 60%, then 18 defects will be removed and 12 defects will remain and pass on to be detected by the next quality assurance activity.

   At each phase, the incoming defects are the sum of defects not removed by the former quality assurance activity, together with the "new" defects introduced (created) in the current development phase.

   The cost of defect removal is calculated for each quality assurance activity by multiplying the number of defects removed with the relative cost of removing a defect.

   The remaining defects, unfortunately passed on to the customer, will be detected by him/her. In these circumstances, full removal entails the highest defect-removal costs.

2. The data components of the model based on published survey results are:
   - The distribution of software defects origins according to the software development phase.
   - The defect removal effectiveness for the various software development phases.
   - The relative cost of removing a detected defect for the various quality assurance activities.

Topics for discussion

7.1

Consider the expected severity of software system failure.

1. What are the main issues that cause the degree of severity?

2. Referring to your answer to (1), can you list three examples of software development projects displaying highly severe failures?

3. Referring to your answer to (1), can you list three examples of software development projects displaying low severity failures?

Solution

1. Main issues that cause the high degree of severity (see Table 8.1):
   - Jeopardizing the safety of a person
   - Causing severe financial losses
   - Causing malfunctioning of services
   - Causing malfunctioning of process control in chemical industry.

2. Examples of software development projects displaying highly severe failures:
   - Software for control of medical condition of patients hospitalized in the intensive care room.
   - Software for controlling the flight of a guided missile.
   - A software module that controls the customer's credit and prevents undesirable sales that exceed the credit limit.

3. Examples of software development projects displaying low severity failures:
   - Software for computerized games.
   - Software for managing the distribution of advertising pamphlets.
   - Software for controlling the random selection of background musical pieces.

7.2

A software development firm is planning a new airport luggage control project. The system is to control luggage transfer from the terminal to the planes, from the planes to the terminal's luggage release system, and from plane to plane (for transit passengers). The airport requires highest reliability for the system and wishes to initiate several new applications that have yet to be implemented in another airport.

1. What SQA methodology should be implemented for this project? List your arguments.

2. Would you recommend integration of additional methodologies in the plan? If yes – what are they and what are their main contributions to the project?

Solution

1. Considering the high severity of possible errors of this software system – the spiral model is recommended, in order to assure the utmost cooperation with the customer and absolute attention to software development risk.

2. Considering the great complexity of the project and the importance of the man–computer interface – integration of the prototype model may be recommended for parts of the project.

   Considering the possibility to reuse software modules developed for other airports – the use of the object-oriented model is recommended.

7.3

HRS Ltd is a software house that specializes in human resource management packages sold mainly to small and medium-sized organizations. Its incentive control and management recruitment software packages are already very popular.

1. What methodology should be applied by HRS? List your arguments.

2. The company wishes to penetrate the area of custom-made human resource management software systems for large organizations such as banks and government agencies. What methodology or combination of methodologies can best fit their new needs?

Solution

1. Considering the great importance of the products high quality – a prototyping methodology, where the marketing department's team of experts serves as the customer's team and thoroughly checks each prototype version, may be preferred. This methodology assures better adaptability of the software package to the customer's needs.

2. The combination that best fits: the prototyping, spiral and object-oriented methodologies. The prototyping methodology will be applied for new modules, where applicable. The spiral model will be the best to manage the customer-developer relations and the risk issues of a large-scale project. The object-oriented methodology will support the reuse of software modules from HRS products.

7.4

Software reuse has become an important factor in the software development industry.

1. Explain the advantages of software reuse.

2. How can a software development firm organize for efficient software reuse?

3. What similar trends you can identify in manufacturing industries (automobiles, home appliances, etc.).

Solution

1. The advantages of software reuse:

Economy – The cost of integrating a reusable software component is much lower than developing new components.

Improved quality – Used software components are expected to contain considerably fewer defects than newly developed software components due to detection of faults by former users.

Shorter development time – The integration of reusable software components reduces scheduling pressures.

2. A software development firm organizes efficient software reuse by establishing a software module library that will enable software developers to locate software modules to suit their needs.

3. Let us take the automobile industry as an example. In this industry, we may find "internal reuse" as well as "over the industry reuse". The "internal reuse" is implemented by sharing the same engine and gearbox in various car models manufactured by the company. The "over the industry reuse" is implemented by the same tires and electric accessories used in cars of different manufacturers.

7.5

Finding him/herself under time and budget pressures, a project leader has decided to introduce "an economy plan" that limits the quality assurance activities to a standard design review, as required by the contract with the customer (50% filter) and a comprehensive system tests (60% filter). Considering the model's contribution to defect-removal efficiency and costs:

1. What are the expected savings, if any, in resources invested for defect removal during the development process as opposed to the standard quality assurance plan?

2. What are the expected effects of the "economy plan" on customer satisfaction? Support your answer with a quantitative comparison to the standard plan.

3. Compare the overall results of the "economy plan" to the results of the standard and comprehensive plans.

4. Based on your answer to (3), can you suggest some general rules about choosing the preferred quality assurance plan?

Solution

An economy quality assurance plan based on the standard quality assurance plan (Figure 7.7) is presented below. The solution is based on the updated version of Section 7.4 of the main text. For the convenience of the readers it is presented here.

Requirement specification (POD=15)

	Ddoc	Dint	Duni	Ddes	Dreq	Total	
ID					15	15	
PD					15	15	**%FE=0**
RD					0	0	
RDRC					1	TRC >	**0 cu**

15

Design (POD=35)

	Ddoc	Dint	Duni	Ddes	Dreq	Total	
ID				35	15	50	
PD				17.5	7.5	25	**%FE=50**
RD				17.5	7.5	25	
RDRC				1	2.5	TRC >	**36.3 cu.**

25

Unit tests (POD=30)

	Ddoc	Dint	Duni	Ddes	Dreq	Total	
ID			30	17.5	7.5	55	
PD			30	17.5	7.5	55	**%FE=0**
RD			0	0	0	0	
RDRC			1	2.6	6.5	TRC >	**0 cu**

55

Integration tests (POD=10)

	Ddoc	Dint	Duni	Ddes	Dreq	Total	
ID		10	30	17.5	7.5	65	
PD		10	30	17.5	7.5	65	**%FE=0**
RD		0	0	0	0	0	
RDRC		1	2,5	6,4	1.6	TRC >	**0 cu**

65

Documentation reviews (POD=10)

	Ddoc	Dint	Duni	Ddes	Dreq	Total	
ID	10	10	30	17.5	7.5	75	
PD	10	10	30	17.5	7.5	75	**%FE=0**
RD	0	0	0	0	0	0	
RDRC	1	1	2.5	6.4	16	TRC >	**0 cu**

75

System tests (POD=0)

	Ddoc	Dint	Duni	Ddes	Dreq	Total	
ID	10	10	30	17.5	7.5	75	
PD	4	4	12	7	3	30	**%FE=60**
RD	6	6	18	10.5	4.5	45	
RDRC	2.5	2.5	6.2	1.6	40	TRC >	**48.9.6 cu**

30

Operation by customer (POD=0)

	Ddoc	Dint	Duni	Ddes	Dreq	Total	
ID	4	4	12	7	3	30	
PD	0	0	0	0	0	0	**%FE=100**
RD	4	4	12	7	3	30	
RDRC	6.9	6.9	17	44	110	TRC >	**804.8 cu**

The economy plan – The process of removing 100 defects

(a) + (b) + (c) Based on the above figure and Table 7.6 the comparison is presented in the following table:

Quality assurance activity	**Economy plan**		**Standard plan**		**Comprehensive plan**	
	Percentage of removed defects	**Cost of removing defects (cost units)**	**Percentage of removed defects**	**Cost of removing defects (cost units)**	**Percentage of removed defects**	**Cost of removing defects (cost units)**
Requirements specification review	---	---	---	7.5	9%	9
Design inspection	---	---	---	---	28.7%	35
Design review	25%	21.3	21.3%	21.3	7.4%	7.1
Code inspection	---	---	---	---	24.4%	31.8
Unit test - code	---	---	25.6%	50	4.2%	5.6
Integration test	---	---	17.8%	66.3	9.8%	18.8
Documentation review	---	---	13.9%	38.9	9.9%	10.9
System test	45%	48.9	7.0%	50.9	4%	13.2
Total for internal quality assurance activities	**70.0%**	**85.2cu**	**93.1%**	**234.9cu**	**97.4%**	**131.4cu**
Defects detected during operation	30.0	804.8	6.9%	139.2	2.6%	22
Total	**100.0**	**890cu**	**100.0%**	**374.1cu**	**100.0%**	**153.4cu**

4. The comparison yields some helpful rules:
   - Quality assurance activities should be performed as early as possible, when the cost of removing a detected error is of the quality assurance activities prior to the system test, and thus the percentage of undetected defects that are “injected” into the system tests is decreased.

- Additional quality assurance activities that provide more diversity of these activities, prove effective in reducing the percentage of defects that are “injected” into the system tests and especially in reducing the percentage of defects that are passed on to the customer.
- Additional software quality assurance efforts, invested in the early phases of the software development process, results in improved defect removal effectiveness. This “investment” proves to reduce the percentage of “passed” defects, where the costs of removing a detected defect are much lower than in the next development phases.

CHAPTER 8

Reviews

Review questions

8.1

There are four direct objectives and two indirect objectives attached to the various review methods.

1. List the direct and indirect objectives of each review method surveyed.
2. For each objective, indicate the review technique or techniques that contribute(s) the most to achieving that objective.

Solution

1. The objectives of the various review methods are marked in the following table. See Table 8.4.

	Review methods			
Objectives	**DR**	**Inspection**	**Walkthrough**	**Expert opinion**
Direct objectives				
1. Detect errors	**M**	**M**	**M**	**M**
2. Identify new risks	**M**			**S**
3. Locate deviations from templates		**M**	**S**	
4. Approve the design document	**M**			
Indirect objectives				
1. Knowledge exchange	**M**	**M**	**M**	**S**
2. Support corrective actions	**S**	**M**		

M=Main objectives, S=Secondary objectives

2. The review method(s) that contribute(s) the most to achieving that objective

Objectives	**Review method that contribute(s) the most to achieving that objective**
Direct objectives	
1. Detect errors	All methods (varies according to error type)
2. Identify new risks	DR
3. Locate deviations from templates	Inspection
4. Approve the design document	DR (the only method assigned for this objective)
Indirect objectives	
1. Knowledge exchange	DR (higher professional levels), inspection (peers level)
2. Support corrective actions	Inspection

8.2

One of the objectives of reviews is to identify deviations from templates, style procedures and conventions.

Explain the importance of enforcing templates and sticking to style procedures and conventions.

Solution

The solution is based on Section 2.3.

The quality risks of non-compliance – increase in the rate of errors – stem from:

- Team members who need to coordinate their own codes with code modules developed by "non-complying" team members can be expected to encounter more than the usual number of difficulties when trying to understanding the software developed by the other team member.

- Individuals replacing the "non-complying" team member (who has retired or been promoted) will find it difficult to fully understand his/her work.

- The design review team will find it more difficult to review a design prepared by a non-complying team.

- The test team will find it more difficult to test the module; consequently, their efficiency is expected to be lower, leaving more errors undetected. Moreover, team members required to correct the detected errors can be expected to encounter greater difficulties when doing so. They may leave some errors only partially corrected, and even introduce new errors as a result of their incomplete grasp of the other team member's work.

- Maintenance teams required to contend with the "bugs" detected by users and to change or add to the existing software will face difficulties when trying to understand the software and its documentation. This is expected to result in an excessive number of errors and the expenditure of an excessive amount of maintenance effort.

8.3

Some people claim that one of the justifications for a small design review team is the need to schedule the review session within a few days after the design product has been distributed to the team members.

1. Could you list additional reasons for preferring small DR teams apart from the anticipated delays in convening a DR session composed of large teams.

2. What reasons motivate attempts to schedule the review session as soon after distribution of the design reports to the team members as possible?

Solution

1. Apart from coordination difficulties typical to large teams, additional reasons for preferring small teams are:
   - In larger DR teams the participants tend to neglect the preparation phase – relying on "others that will certainly read the document thoroughly".
   - As at least some of the participants are not really acquainted with the contents of the document, a short presentation of the document is not sufficient, and much time, if not most of the session time, is used for the developer's presentation and clarification questions. In such situations only a short time is devoted to discussing issues of reviewer's findings and comments.
   - The DR session becomes tedious as repeated views are presented, mixed with requests for information raised by participants who have not done the required preparation.
   - The costs of the review are excessively high, caused by the large team and the long hours spent in the DR sessions.

2. A long delay in the DR session means delays in the expected list of instructions and required corrections (action items) and the needed approval of the completed phase of the development process. Such a delay involves:
   - Waste of the team's working days or at least a period of inefficient work, that might be found in need of rework once the DR team's instructions and required corrections are available.

- Increase in the possibility of late completion of the next task and missing the next milestone of the development process and instructions regarding the resumption of the development process.

8.4

One can expect that in many cases, participants in an inspection session are able to suggest solutions for a detected defect or, at least, point out possible directions for its solution. While it is clear that these suggestions are crucial for the development team, it is commonly recommended to avoid any discussion about solutions during the inspection session.

1. List your arguments in favor of this recommendation.

2. What other kinds of cooperation between the moderator and the review team would you prefer to observe in a session?

Solutions

1. It is recommended to avoid discussions about solutions during the inspection session as:
   - It saves the inspection team's time.
   - It avoids the team's inability to complete the session's planned agenda.

2. The moderator is also expected:
   - To avoid deviations by team members from the inspection issues.
   - To avoid personal comments by team members and focus on professional issues of the inspected document.
   - In cases of disagreement between the team members – to document the case and shift the discussion to a different forum or to a special meeting.

8.5

It is quite natural to expect participation of the document's author (the designer) in a review of any type.

1. What are the arguments in favor of his/her participation?

2. What are the differences in the part played and status of the author in each of the review methods discussed?

Solution

1. The document's author can provide the team with the required explanations and will be able to understand the team's comments better by listening to the discussion and being able to add clarification questions, rather than by reading the team's report.

2. In formal design reviews the author is required to open the session by a short presentation of the document. Later they are required to provide explanations to questions raised by the team. They may also express their comments to error claims discussed by the team members.

   In the inspection session it is the moderator who leads the discussion and usually does not present the document.

   In walkthrough sessions it is usually the author who initiates and coordinates the session. They serve as the presenter of the document and usually as the review leader.

8.6

The preparations made by of members of inspection teams are considered to be of greater depth and thoroughness when compared with the preparations for walkthroughs.

1. What activities are included in such high levels of preparation?

2. Do you think that inspection teams having 15 members can achieve similarly high levels of preparation?

Solution

1. Inspection team members are expected to participate in an overview presentation meeting and to thoroughly read the document prior to the inspection session. They are expected to use a checklist to support their review.

2. Participants of such a large inspection team rely on each other, namely expecting the others to read the document thoroughly. An inspection session of this type is expected to waste a great part of its time on explanations and comments not needed once the document has been read prior to the session.

8.7

Pressman lists 13 golden guidelines for successful design review (see Frame 8.3).

1. Four of the golden rules deal with design review infrastructure. Can you list these golden guidelines and elaborate on the importance of the infrastructure elements and how they affect software quality?

2. It is often claimed that the six golden guidelines dealing with the design review session are as applicable to inspection as they are to walkthrough sessions. Can you

list these common golden guidelines and explain the reasons for their broad applicability?

Solution

1. Design Review Infrastructure

   - Develop checklists for each type of design document, or at least for the common ones – supports the various review activities by assuring higher completeness of preparations for reviews and the reviews themselves.
   - Train senior professionals to treat major technical as well as review process issues. The trained professionals serve as a reservoir for DR teams. The importance of trained DR team leaders and members is that more thorough, effective and efficient reviews, are performed.
   - Periodically analyze past DR effectiveness regarding defect detection to improve the DR methodology. The importance of periodical analysis of the process of DRs is by producing improved checklists, improved methodology and procedures.
   - Schedule the DRs as part of the project activity plan and allocate the needed resources as an integral part of the software development organization's standard operating procedures. The importance of incorporating DR activities in the project schedule is the ability to perform DRs with no time pressures, namely a proper review and complete defect correction process.

2. All 6 guidelines are common to all review methods as they stem from the requirement of effective and efficient review session. Accordingly the recommended number of participants is similar, though their affiliation may vary.

Topics for discussion

8.1

A proposal for changing an inspection procedure involves adding a new reporting requirement, as follows: “At the end of the session or the series of sessions, the inspection leader will submit to management a copy of the inspection session findings report and a copy of the inspection session summary report.”

1. Consider the proposal and list possible arguments, pro and con, regarding the change.

2. What is your recommendation – to add the new reporting requirement or not? Explain.

Solution

1. Pro arguments:

- The report will provide management with an additional and earlier feedback source of information that will improve management's control.
- The requirement to report to the management will cause the report to be more balanced and of higher quality.

Con arguments:

- The awareness of the contents of the report being checked by management will change the nature of the inspection session to being less open. As authors will consider the possibility that a negative inspection report will harm their promotion prospects or the level of their next task, they will strongly object to any suggested criticism of the document and try to diminish the number of review findings.
- On the other hand, participants will hesitate to raise discussion on subjects that need clarification, taking into account that the discussion, no matter its results, may harm their colleagues when management checks the report.

2. It is recommended to avoid the suggested change since its disadvantages outweigh its advantages. The management has several alternative sources of information about the project's quality and detected defects, to mention one – the formal design review. The possibility to keep the open nature of a peer review, bearing no control of higher levels in the organization, is of special importance.

8.2

David Martin just finished his inspection coordinator course. After obtaining his first appointment, he plans to add his personal secretary to the inspection team for the purpose of serving as session scribe and producing the required reports. He assumes that her participation will free him for the coordination tasks and enable him to conduct the session successfully.

1. Is it advisable to employ a secretary (a non-information technology professional) as a scribe in an inspection session? List your arguments pro and con.

Solution

The secretary is expected to perform the reporting task efficiently. However, it is expected that for performing the task he/she will require many clarifications to ensure correct reporting. The clarification questions will break the natural flow of review discussion, at least causing some waste of time. In cases where no clarification is requested by the secretary one may expect a proportion of wrong or incomplete documentation. Reporting by the coordinator himself is expected to be much more accurate and to mention the main findings.

8.3

Compare the various review techniques.

1. In what aspects are design reviews more formal than inspections?

2. In what aspects are inspections more formal than walkthroughs?

Solution

The solution is based on Table 8.4

1. Design reviews are made more formal by the seniority of participants and especially by its authority to decide about the continuation of the project. It is also made more formal by the close follow up requirements, including a planned schedule for performing the corrections.

2. Inspections are made more formal than walkthroughs by the participants' preparations for the review session, by the training of participants, by the use of checklists and the follow up process. The use of inspection findings report as a corrective action source, usually not required from walkthroughs, also adds formality to the inspection procedure.

8.4

The chapter offers three different methodologies for team review of design documents.

1. Which of the methodologies should a software development organization choose?

2. Can more than one method be chosen and applied for the same document? Alternatively, is it recommended to apply all three methods? List your arguments.

Solution

1. It is recommended that an organization should apply all the methodologies as well as expert opinion. Formal design reviews are actually mandatory as a tool for approval of software development documents and products. Inspections and walkthroughs may be considered alternatives, so that the organization can choose to apply only one of these methodologies. However, it is recommended to employ in an organization both peer review methodologies so that every peer group that plans a peer review activity can use the methodology that suits them best. The use of expert's/consultant's opinion should be welcomed in cases where the organizations lack the expertise in specialized fields.

2. There is no contradiction in applying peer review methodologies in early stages and later a formal design review. Different peer review methodologies and consulting may be applied to different parts of the document. However, as a rule, it is not

recommended that a person will participate in more than one review activity that refers to the same part of the document.

In general, application of all three review methodologies and even adding experts' opinions may improve the quality of the resulting document, given that considerations of budget and project schedule are observed.

CHAPTER 9

Software testing – strategies

Review questions

9.1

Not a few software industry professionals maintain that the main goal of software testing is "to prove that the software package is ready".

1. Explain in your own words why this is not a suitable goal for software testing.

2. What other goals might replace the goal mentioned above, and what gains in the effectiveness of the testing team can be expected from this change?

Solution

1. The goal of proving that the software package is ready encourages the testing team to prepare a test plan that will support the development team by avoiding test cases with high potential of finding errors. By this they show cooperation with the developers with higher prospects of revealing no errors or only a few. The result is the high possibility of releasing a low quality product.

2. The real and preferred objectives are those mentioned in Frame 9.2, namely:

   Direct objectives

   - To identify and reveal as many errors as possible in the tested software.
   - To bring the tested software, after correction of the identified errors and retesting, to an acceptable level of quality.
   - To perform the required tests efficiently and effectively, within budgetary and scheduling limits.

   Indirect objective

   - To compile a record of software errors for use in error prevention (by corrective and preventive actions).

   The gains in effectiveness are due to a testing plan that is directed to sections of software of high error potential, according to testing experience, complexity of the section, experience of developers, etc. As a result, the testing resources are used more effectively, and within a given budget reveal a higher proportion of errors, especially those which are of high severity.

9.2

Explain in your own words why big bang testing is inferior to any method of incremental testing conducted for software packages that are not small.

Solution

Medium and large software packages are composed of many modules, interconnected by software code instructions. As a result, an error in one module may cause software faults in several other modules. A software fault does not necessarily happen in the module where the original error is located. Therefore by applying big bang testing we may frequently face a situation where we find it difficult to allocate the origin of the error that caused the discovered software fault. Therefore in a great many of these cases the proposed corrections are erroneous. Accordingly, by applying a big bang testing plan, it is difficult to achieve high software quality (revealing a high percentage of errors and performing the right corrections).

9.3

Module G12 is coupled with seven lower-level modules and only one upper-level module.

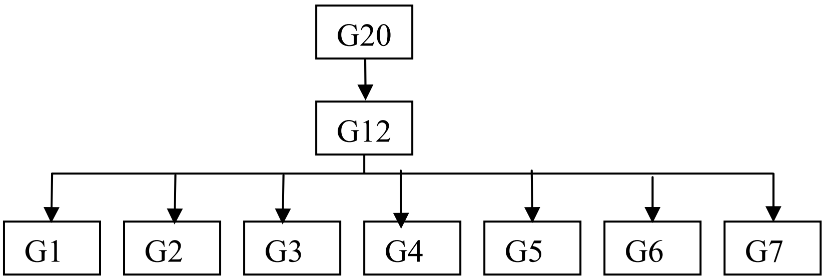
1. Discuss how the number of couplings affects the efforts required for incremental testing strategy.

2. Consider the case described above. What are the effects of module G12's specific coupling situation on the resources required to perform unit tests according to the top-down strategy and the bottom-up strategy.

Solution

1. The number of couplings affects the incremental testing efforts by the number of stubs and drivers required to allow testing of each of the modules.

2. The modular structure of the unit is as follows:



We examine the case of all 9 modules are to be unit-tested. Let us check the stubs and drivers that are required.

In top–down testing

Stage 1: Module G20 is unit tested. A stub for module G12 is required.

Stage 2: Module G12 is unit tested. Stubs for modules G1, G2, G3, G4, G5, G6 and G7 are required.

Stage 3: Modules G1, G2, G3, G4, G5, G6 and G7 are unit tested. No stub is required.

In bottom–up testing

Stage 1: Modules G1, G2, G3, G4, G5, G6 and G7 are unit tested. A driver for module G12 is required.

Stage 2: Module G12 is unit tested. A driver for module G20 is required.

Stage 3: Module G20 is unit tested. No driver is required.

9.4

Section 9.4 mentions the terms *path coverage* and *line coverage*.

1. Explain in your own words what the terms mean and list the main differences between these coverage metrics.

2. Explain why the implementation of path coverage is impractical in most test applications.

Solution

1. Path coverage requires that every possible calculation path be covered by a test case. Thus, path coverage requires that a line in the program be tested several times as required by its inclusion in the various calculation paths. On the other hand line coverage requires that each of the software lines be tested at least once. In other words, we may avoid tests of calculation paths when all their lines were previously tested.

2. The implementation of path coverage requires performing a large number of test cases as a typical medium software package allows for hundreds of thousands or even millions of different operation paths. This fact, and more specifically the time and budget considerations, makes it impractical to perform plan for path coverage in almost all software packages. Consideration of achieving path coverage is limited to programs where highly severe damages are expected in case of software failure, as in medical intensive care equipment, aviation equipment and weapon systems. In these cases one may apply path coverage to those modules that signify the highest damage potential.

9.5

“Bengal Tours” is a city center travel agency that specializes in tours and vacations in Canada. The agency regularly employs permanent 25 employees. During the spring and summer, the agency employs an additional 20–25 temporary staff, mostly pensioners and students. The agency is considering purchasing the right to use the software system “Tourplanex”, which supports the planning with flight and vacation site vacancies and price information. If purchased, the software will become the main working tool for the agency staff.

1. Discuss the importance of the training usability and operational usability tests to be performed by the agency before it purchases “Tourplanex”.

2. Suggest to “Bengal Tours” management training usability and operational usability tests that they should apply on the program.

Solution

1. The travel agency’s staff is expected to use the Tourplanex software package hundreds of times a day in order to retrieve information related to vacancies, prices and tour options according to customers’ inquiries and as a tool for preparing proposals. Therefore a “friendly” software, namely one that requires less time for an inquiry and easier to reach full control of, should be preferred. The travel agency will need quantitative results about the average time required in order to compare “Tourplanex” with competing software options.

   Great attention should be given to the training usability achievements of “Tourplanex” because of the temporary staff the agency is employing, and the time that should be invested in each of these recruited temporary employees, who are expected to serve for only few months.

2. The tests should be of two phases. Phase one should deal with training usability. The agency should use typical temporary employees who will be trained by the software company. The time spent in training should be listed and the performance of the trained employees should be measured in terms of minutes (seconds) required to reach the information needed for a typical inquiry. The performance should be tested again after few days to measure the performance improvement reached through regular application of “Tourplanex”. It should be noted that the lists of inquiries should be prepared ahead and should reflect the typical composition of inquiries in the agency. Once the quantitative results are available for management consideration, a comparison with the current situation and competitive software options may be performed.

Topics for discussion

9.1

"Bhealthy Ltd" is a medical insurance company that reimburses the cost of drugs and various other medical expenses to its customers. According to current procedures, customers are asked to present receipts of drug purchases together with the relevant physician prescriptions and other medical documents. Reimbursement is calculated according to the insurance agreement stipulations:

- Two lists of drugs are in effect for the purpose of reimbursement: class A and class B.

  Class A: 90% of the costs of each purchased drug are reimbursed by Bhealthy.

  Minimum customer participation of $5.

  For example: A $10 drug is reimbursed by $5 (customer's participation $5).

  An $80 drug is reimbursed by $72 (customer's participation $8–10%, reimbursement 90%)

  ***Class B*:** 50% of the costs of each purchased drug are reimbursed by Bhealthy (no excess).

- A check is prepared and sent to the customer. The insurance agreement states the period of 45 days for the company to complete the reimbursement.

- For some class A drugs the customer should prefer to buy the medications as a private customer as there is no reimbursement expected (drug's price is below $5).

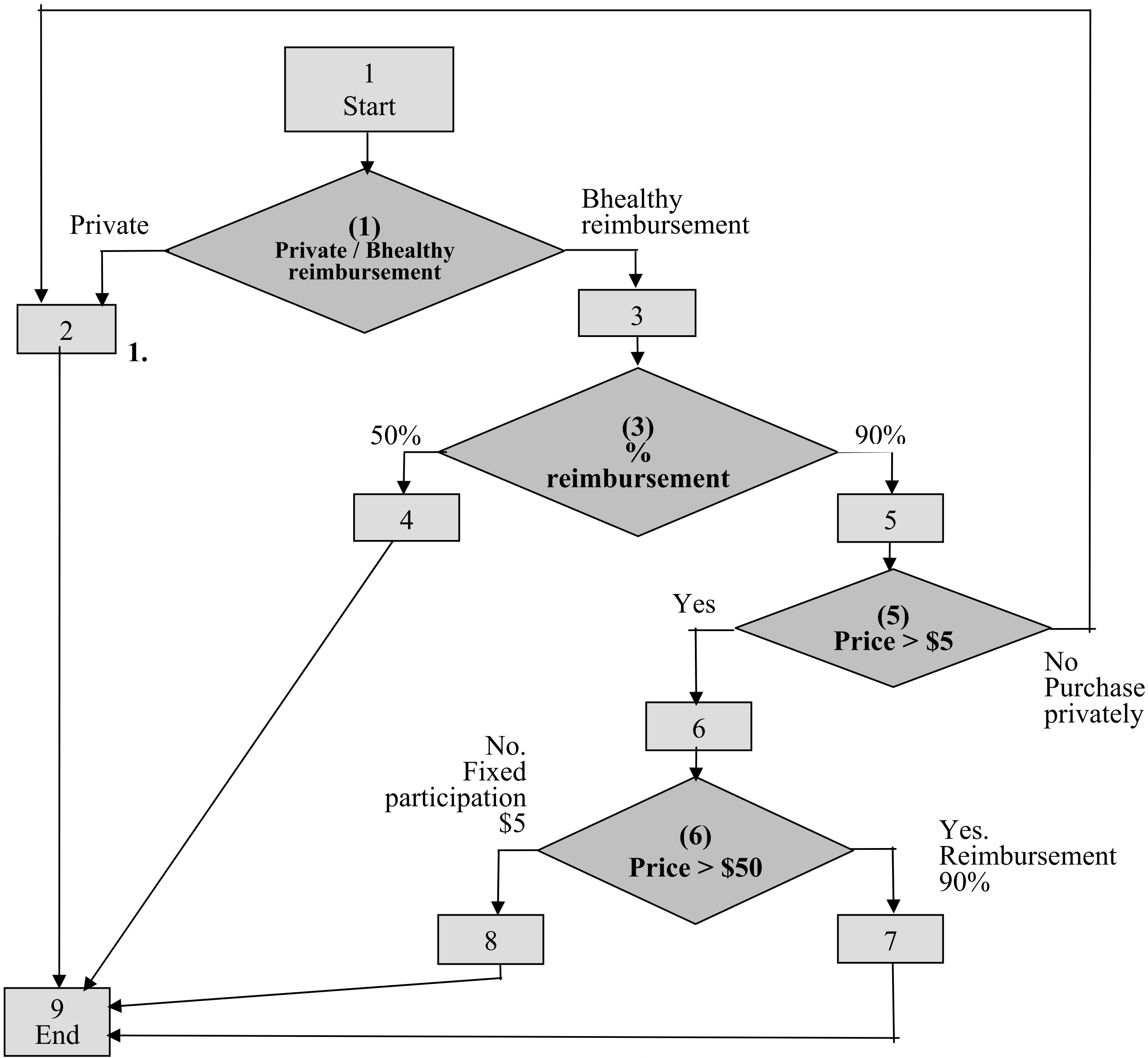
The procedure described proved to be very expensive for Bhealthy at the same time as provoking much subscriber dissatisfaction. The growth in of the number of subscribers as well as the problems of complying with the current procedure motivated a new agreement with the licensed pharmacies. The agreement authorized the licensed pharmacy to deduct the reimbursement sums from the drug invoices; Bhealthy will then reimburse the pharmacies monthly for the deducted sums.

Bhealthy decided to prepare a special pharmacy software package that combines regular pharmacy sales operations with the operations required by its agreement with the licensed pharmacies and its subscribers.

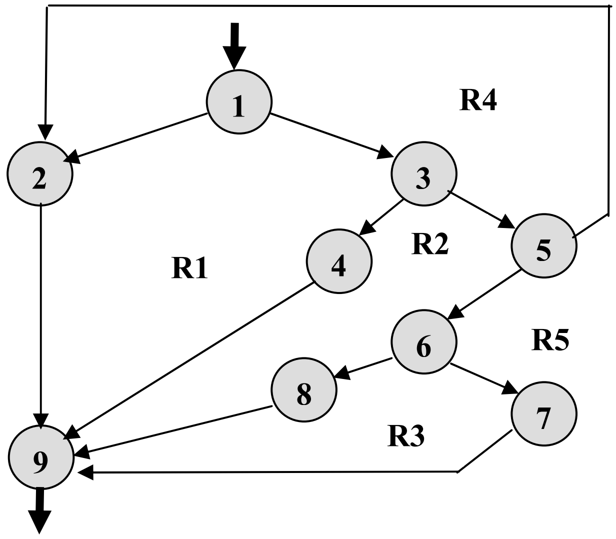
Consider the invoicing module that prepares invoices for Bhealthy prescriptions as well as for regular sales of prescriptions and other items at a licensed pharmacy.

1. Prepare a flow chart for the module.
2. Prepare a program flow graph for the module.
3. Calculate the cyclomatic complexity for the module.
4. Prepare the maximal set of independent paths according to (3). Document the basic paths and indicate the added edges of each independent path.

Solution



1. **Flow chart for the module**



2. **Program flow chart for the module**

3. (1) V(G) = R = 5

   (2) V(G) = E – N + 2 = 12 – 9 + 2 = 5

   (3) V(G) = P + 1 = 4 + 1 = 5

4. The maximum set of independent path

Path No.	The path	Edges added by the path	Number of edges by the path
1	1–2–9	1–2, 2–9	2
2	1–3–4–9	1–3, 3–4, 4–9	3
4	1–3–5–2–9	3–5, 5–2	2
4	1–3–5–6–8–9	3–6, 6–8, 8–9	3
5	1–3–5–6–7–9	6–7, 7–9	2

9.2

“Police Star 1000 System” is the new prestigious software system for recording all the verbal communication (line telephone, cellular telephone and wireless) nationwide to be instituted by the police force. One feature of the system is its ability to supply any voice record completed in the last 12 months within 15 minutes in 98% of the applications. The system is planned to be operative within 10 months.

1. Discuss the importance of conducting comprehensive load tests for the system.

2. Suggest the recommended guidelines for these load tests.

3. What basic data about police activities would you recommend to collect in order to plan the load test according to your recommended guidelines?

Solution

1. The system is expected to store an immense quantity of recordings. The task of allocating the required voice records becomes more and more difficult as the voice recording files grow. Therefore results of any test that is not performed for the full load of recording may be misleading.

2. The load test should be performed for a recording file of the size expected for a full year of recording. It should contain recordings of all the types of recording and of a great variety of sources.

3. Typical recordings: citizens complaints, interrogation of suspects, findings in surveys in crime sites, findings of laboratory investigations, summary of inquiry status and planned steps to be taken to continue the investigations.

9.3

"Super Saving Light" is a new software system for control of street illumination and enhancement of its economy, developed for municipality maintenance departments. Among its functions are:

1. Commencement and conclusion of street lighting according to daily timetable, scheduled annually.

2. Partial illumination (only one of each two lights will be activated) during the first and last 15 minutes of each illumination period activated by (1) .

3. Measurement of natural light conditions by special sensors to ascertain if natural lighting is insufficient (e.g., on cloudy days), leading to earlier commencement of street illumination and later conclusion of illumination. In these cases, only one of a trio of streetlights will be activated.

4. Reduction of illumination according to traffic density, monitored by a traffic sensor installed at every road section, which will reduce illumination as follows: If traffic density is below 1 vehicle per minute, only half of the street lights in the road section will be activated; if traffic density is below 0.3 vehicles per minutes, only one-third of the lights will be activated.

Mr Jones, head of the testing team, claims that black box testing is insufficient and that white box tests are necessary for testing "Super Saving Light".

Support Mr. Jones' claim with three software error examples based on the illumination rules described above. In the examples you choose, black box test results will be "OK", while white box testing of the same example will detect at least one error. For each example, explain why errors undetected by black box testing will be detected by white box testing.

Solution

The 3 test cases are presented in the following table.

The conditions	**Test case 1**	**Test case 2**	**Test case 3**
1. Scheduled time, not within the first or last 15 minutes			+
2. Scheduled time, within the first or last 15 minutes		+	
3. Insufficient natural light condition	+		
4. Traffic density below 1 vehicle per minute			+
5. Traffic density below 0.3 vehicle per minute			
Black box result	1/3 of lights activated	1/2 of lights activated	1/2 of lights activated
White box result	1/3 of lights activated	1/2 of lights activated	1/2 of lights activated

In the above test cases, which are relatively simple, no error has been found.

These test cases are not able to reveal error situations in the following examples:

a. The effect of condition 3 should not be applied to the scheduled illumination time.

b. The effects of more than one conditions is not cumulative. The minimum of the illumination directions should be applied. For example if conditions 2 and 5 are applied, the black box and white box results are 1/6 and 1/3 of lights activated correspondingly.

It should be noted that more elaborate black box test cases would reveal these errors.

9.4

Based on the "Super Saving Light" case described above:

1. What input variables are required for test cases and what are the required output variables?
2. Suggest three to five simple test cases having low potential to identify errors.
3. Suggest three to five test cases that you believe contain serious potential for error.
4. Suggest three to five test cases to deal with boundary value situations.

Solution

1. The required “Super Saving Light” tables:
   - The daily scheduled lighting timetable.
   - The street sections list.

   The required test cases input variables:
   - The date.
   - The time.
   - The street section.
   - The natural illumination.
   - The traffic density.

2. Test cases of low potential to discover errors

Condition	Test cases				
	1	**2**	**3**	**4**	**5**
Time relative to commencement of scheduled illumination	–34 min.	+12 min.		+100 min.	
Time relative to scheduled illumination conclusion			+55 min.		–45 min.
Natural illumination	Insufficient	Sufficient	Sufficient	Sufficient	Sufficient
Traffic density	3 veh/min	veh/min	veh/min	0.05 veh/min	0.9 veh/min
Result: lights to be activated	1/3	1/2	All lights	1/3	1/2

3. Test cases of high potential to discover errors

Condition	**Test cases**				
	1	**2**	**3**	**4**	**5**
Time relative to commencement of scheduled illumination	+64 min.	+12 min.		+100 min.	
Time relative to scheduled illumination conclusion			+55 min.		–45 min.
Natural illumination	Insufficient	Insufficient	Sufficient	Sufficient	Sufficient
Traffic density	0.2 veh/min	0.05 veh/min	0.05 veh/min	0.07 veh/min	0.9 veh/min
Result: lights to be activated	1/2	1/3	No lights	1/3	1/2

4. Test cases to deal with boundary value situations

Condition	**Test cases**				
	1	**2**	**3**	**4**	**5**
Time relative to commencement of scheduled illumination	+15 min.	+16 min.		+15 min.	
Time relative to scheduled illumination conclusion			+15 min.		–16 min.
Natural illumination	Insufficient	Insufficient	Sufficient	Sufficient	Sufficient
Traffic density	0.99 veh/min	0.29 veh/min	0.3 veh/min	0.29 veh/min	1.0 veh/min
Result: lights to be activated	1/2	1/3	No lights	1/3	All lights

9.5

Referring to the "Bhealthy" case discussed in Topic 9.1, the following is the list price for a sample of 12 medications, including the cases' reimbursement class:

Medication Name Code	**Medication Classification According to Bhealthy**	**Price ($)**
101	Not included	$3.45
102	Class B	$10.60
103	Class A	$5.50
104	Class A	$19.50
105	Class A	$4.50
106	Class B	$28.00
107	Not included	$74.99
108	Class B	$8.30
109	Class A	$3.90
110	Class B	$22.70
111	Class A	$5.20
112	Class A	$87.20

1. Based on the above price list, prepare the set of test cases required for implementing the maximal set of independent paths appropriate for your solution to Topic 9.1 question (4).

2. Assume that Bhealthy changes its minimum subscriber participation for class A medications from $5 to $6. Will the test cases in (1) have to be changed? If yes, make the necessary changes and present the updated test case file.

Solution

1. The solution is marked in the following table.

Path No.	The path	Bhealthy or private	Reimbursement %	Price >$5	Price >$50	Medication code
1	1–2–9	Private	0%	---	---	101, 107
2	1–3–4–9	Behealthy	50%	---	---	102, 106, 108, 110
4	1–3–5–2–9	Behealthy	0%	No	---	105, 109
4	1–3–5–6–8–9	Behealthy	90%	Yes	No	103, 104, 111
5	1–3–5–6–7–9	Behealthy	90%	Yes	Yes	None in the original list. Added medication 112

2. The comparison of the paths for $5 and $6 is presented in the following table.

Path No.	The path	Bhealthy or private	Medication code $5 participation	Medication code $6 participation
1	1–2–9	Private	101, 107	101, 107
2	1–3–4–9	Behealthy	102, 106, 108, 110	102, 106, 108, 110
4	1–3–5–2–9	Behealthy	105, 109	103, 105, 109, 111
4	1–3–5–6–8–9	Behealthy	103, 104, 111	104
5	1–3–5–6–7–9	Behealthy	None in the original list. Added medication 112	None in the original list. Added medication 112

CHAPTER 10

Software testing – implementation

Review questions

10.1

“Alpha phone” is a software package that includes the following among its features:

- It manages a household phone address book.
- It produces printouts of the phone book according to a variety of classifications.
- It analyses the monthly traffic of incoming and outgoing phone calls according to the classifications mentioned above.

You are called to perform a documentation test of the very elegant “alpha phone” users’ manual.

List at least 5 types of possible documentation errors in the manual.

Solution

Possible documentation errors:

The user’s manual provides the directions:

- For retrieval of only one phone number for each addressee, but does not provide the necessary directions for retrieval of all the addressee’s numbers in cases where more than one phone number was listed.
- For analyzing incoming and outgoing phone calls for entire months but does not explain how to retrieve this information for a period defined by commencing and ending dates.

The user’s manual provides erroneous directions regarding:

- Possible analyzing of phone call traffic according to the hour of the day.
- The possibility to produce a phone address book classified according to town.
- The possibility to produce a partial phone address book according to affiliation, etc.

10.2

"MPT star" is a program for calculating the annual municipal property taxes, based on the neighborhood, the type of property (house, store, apartment, etc.), the size of the property, the discounts to which the owner is entitled (pensioners, low income large family, single-parent family, etc.)

Suggest a framework for stratified sampling test cases from the citizens' file. List your assumptions about the population's distribution.

Solution

A proposed stratification is defined as follows:

1. Properties that are not entitled to any discount

2. Properties that are entitled to one type of discount

3. Properties that are entitled to two types of discount

4. Properties that are entitled to three types of discount

5. Properties that are entitled to more than three types of discount.

The stratification shown above is of progressive error potential. In other words, strata 2 is of higher error potential than strata 1, strata 3 is of higher error potential than strata 2, etc. Accordingly the higher the error potential the higher proportion of test cases justified.

Strata 1 may be sub-divided so that each category of properties is represented where the categories of lower frequency will be entitled to a higher percentage in the sample of test cases.

10.3

"In most cases, the test case file preferred should combine sample cases with synthetic cases so as to overcome the disadvantages of a single source of test cases and to increase the efficiency of the testing process." Taken from Section 10.2.2.

1. Elaborate on how applying a mixed-source methodology overcomes the disadvantages of a single-source methodology.

2. Elaborate on how applying a mixed-source methodology enhances testing efficiency. Provide a hypothetical example.

Solution

1. The main disadvantage of a random sample is its size and the very high efforts required to perform the tests. In other words, the need for a very high number of test

cases to assure a sufficient number of non-standard situations of high error potential to be covered by test cases, leads to an inefficient test procedure. The main disadvantage of synthetic test cases that stems from its planned nature is their inability to identify unexpected errors, errors that do not relate to the expected error situations list used by planners of the synthetic test cases.

2. Applying a mixed-source methodology reduces the disadvantages of each of the single-source methodologies. Adding synthetic test cases that focus on non-standard situations, allows us to reduce the size of the random sample significantly to one-third or even to one-tenth of its original size. The added random sample of test cases is due to include some test cases that relate to the unexpected situations.

   Hypothetical test case options:

Methodology	**Random sample**	**Synthetic test cases**	**Mixed**
Number of random test cases	1,000	---	200
Number of synthetic test cases	---	200	200

   The mixed methodology test case file is expected to be more effective than the single methodology test case files in error identification.

10.4

Software testing experts claim that applying a stratified sample of real-life test cases is more effective for identifying errors and more efficient than regular random sampling.

1. If you agree, list your arguments.

2. If you disagree, list your contradictory arguments.

Solution

1. I agree with the experts' claim. In most cases real life populations may be classified into sub-populations where most of the population falls into one or two sub-populations of simple cases, the rest of the cases fall into small size sub-samples, of rare and usually more complicated cases (of much higher error potential). Applying regular sampling means that most of the test cases will deal with the simple and common cases with many repetitions of the same situations, giving rise to a waste of most of the testing efforts directed to the large common sub-populations. Applying stratified sampling enables substantially reducing the sampling proportion of these common sub-populations and increasing considerably the sample size of rare sub-populations, e.g., three fold and even more. The result is an increased effectiveness and efficiency of the testing plan.

10.5

Reviewing the advantages and disadvantages of automated software testing:

1. Explain the main advantages and disadvantages of automated tests in your own words.

2. Referring to your answer to (1), suggest what project characteristics are most suitable for automated testing. List your assumptions.

3. Referring to your answer to (1), suggest what project characteristics are most unsuitable for automated testing. List your assumptions.

Solution

1. The main advantages of automated testing:

   a. The tests are performed accurately and completely as planned. Manual testers are some times inaccurate and do not perform test plans completely.

   b. The tests' documentation of findings is accurate in its lists and summary reports, when compared with manual reporting that may be inaccurate and incomplete.

   c. Automated computerized reports, compared to manual reports, are naturally more comprehensive and allow for a greater variety of information types to be provided.

   d. Performing the tests requires almost no human effort, while manual testing is a major "consumer" of tester's time.

   e. Performance of an automated test requires much less time than performing the same test manually. In addition, automated tests may be performed continuously 24 hours a day 7 days a week, while it is difficult to operate manual tests during the night shift or during weekends.

   f. The fact that performing automated tests is quick and does not require much in the way of human resources, allows us to plan full size regression tests, and so identify errors caused in unexpected parts of the package by erroneous corrections (in package parts the planners will decide not to include manual regression tests in order to save testers' resources).

   g. Automated testing allows performing test types that are practically impossible to perform manually, to mention here especially, the various types of load tests.

   The main disadvantages of automated testing:

   a. The software packages for automated testing are very expensive and the training of staff to the required level of expertise in operating automated tests is long and therefore very expensive.

   b. The development costs of automated testing package to deal with a new type of software are very expensive.

c. The preparation of an automated test plan requires high manpower resources (of a highly specialized team).

d. For many types of programming automated testing packages are not available.

2. Project types that are most suitable are the more complicated software packages, where only a small part of the software is developed based on recycled modules, provided an automated testing tool is available. In projects of this nature a detailed and accurate control of identified errors and follow up of their correction is essential, the capability to carry out full system regression tests is very crucial. To these project characteristics one should add projects which are planned to serve a great number of users, and where the availability requirements are to be achieved at all times, especially in periods of high usage. Examples of applications of this type are bank teller systems and police inquiry systems that serve “on line” policemen on patrol missions.

10.6

Mr. Aleppo, the head of the software development department, claims that beta site tests should be always carried out as early as possible in the development process as there are no disadvantages in this method.

1. Are beta site tests really a “disadvantage free” method? If not, what are the beta site tests' main disadvantages and risks?

2. Recommended guidelines that will minimize the risks and disadvantages in applying beta site tests as listed in (1).

Solution

1. Beta site tests support our test program by unexpected findings that no planned test program has identified. However there are several disadvantages:

   – The information received is not the result of systematic tests and, as expected, includes a majority of trivial findings, already identified by the testing team.

   – The error reports are “not professional” and in a large number of cases do not clearly describe the nature of the discovered error. Some of the errors reported by beta sites are later found to be non-errors reported by mistake.

   – The error documentation is in many cases not complete and/or not clear. In many cases it is not easy to reproduce the error conditions.

   – Much effort is required to examine the beta site reports and often it is not economically justified.

2. The following are guidelines for avoiding most of the disadvantages of beta site activities:

   – Leading a customer beta site group based on follow up of beta site reporting by the customers regarding the quality of information and the time to reaction.

– Involving only beta site group members in beta site activities.

– Candidates to the beta site group will be carefully chosen, mainly according to their software quality assurance level of activities.

– A member of the beta site group that continues to be unsuitable should not continue their membership in the group.

Topics for discussion

10.1

"RSM–Real Time Software Magicians Ltd." signed a contract with defense authorities for development of "Light in the Darkness". "Light in the darkness" is an advanced night vision system for infantry use. The system is based on a comprehensive pattern recognition model, whose development was completed last year at a prestigious university. It is expected to identify the presence of a human, standing, sitting or lying, from a distance of 100 meters. The system to be used by the soldier ("soldier's set") contains a unit for geographic identification based on the satellite GPS technology, and includes a ciphered communication system linking the soldier to headquarters.

Headquarters' central unit for processing the data received from the night vision system **(**from the front line) is not part of the "Light in the Darkness" project.

The "Light in the Darkness" system is constructed of 4 subsystems (integrations) and 13 units (modules) as follows:

Subsystems (Integrations)	**Units (Modules)**
1. Subsystem night recognition of humans	1.1 Unit that identifies humans according to a mathematical pattern recognition model
	1.2 Unit for display of the identified humans on the screen of the "soldier's set".
	1.3 Unit for calculating and displaying on the screen the distance to the identified humans.
	1.4 Unit for creating a display on the soldier's screen of the combined identification data from all the unit's soldiers
	1.5 Unit for warning the soldier according to the combined identification data
2. Subsystem for ciphered communication	2.1 Unit for communicating the geographic location of the military unit
	2.2 Unit for communicating the identification of the observed human figures

Subsystems (Integrations)	Units (Modules)
	2.3 Unit for performing the ciphered communication according to military communication standards
3. Subsystem for documentation of the soldiers sets' usage	3.1 Unit for recording set usage time by the soldier
	3.2 Unit for daily reporting of the soldier's set usage according to the soldier's set identification number
	3.3 Unit for producing notices about soldier's sets that exceed 1000 hours of usage (for preventive maintenance)
4. Subsystem for recording of communication times of the military units	4.1 Unit for recording communication times from the units
	4.2 Unit for processing daily reports of when communications were received

Due to timetable and budget considerations, it was decided to carry out only 5 unit tests and only 2 integration tests on the new system.

1. Provide support for the "RSM–Real Time Software Magicians Ltd." testing team by planning a comprehensive method to determine the priorities of the different modules to be included in the unit-testing plan. The priorities will be based on two criteria:

   – Severity level: the severity of the damages anticipated if this module fails during real application of the system

   – Risk level: the probability that the module will fail if it is not tested and corrected accordingly.

2. Apply the method suggested in (1) to rank the priorities of the 13 modules of "Light in the Darkness" described above. List your assumptions.

3. Adapt the same method to determine the priorities of the integrations to be tested as listed in the description of the example.

4. Considering that the modules have already been included in the unit test plan, does this fact change the method for determining the priorities you reached with respect to integrations in (3)? Applying the updated method to the integrations of "Light in the Darkness", is there any change in the resulting priorities?

Solution

1. An optional method for defining module priority will be discussed. The proposed method is based on:

   a. Rating of severity of module failure results by a 5-level scale.

   b. Rating of risk of failure by a 5-level scale.

c. Calculation of testing priorities by the multiplication formula:

[testing priority] = [severity level] X [risk level

The severity scale, based on Table 8.1, is:

Level	Description
5	Critical – prevents accomplishment of essential capability
4	Severe – adversely affects accomplishment of essential capability, where no work-around solution is known
3	Moderate – adversely affects accomplishment of essential capability, where a work-around solution is known
2	Low – user inconvenience that does not affect operational essential capabilities
1	Minor – inconvenience for development, maintenance or user's personnel but does not prevent realization of responsibilities

The risk scale is:

Level	Description
5	Very high
4	High
3	Moderate
2	Low
1	Very low

2. The ranking process of integrations priority is presented in the following table:

Sub-system (integration)	Severity level	Risk level	Priority rank	Assumptions
1. Sub-system night recognition of humans	5	5	25 – I	Failure of this integration, based on new technology, will endanger the soldiers that rely on it. Newly developed technology.

Sub-system (integration)	Severity level	Risk level	Priority rank	Assumptions
2. Sub-system for ciphered communication	5	2	10 – II	Failure of the communication system will cause failure of the whole system. The integration is based on established technologies.
3. Sub-system for documentation of the soldiers sets' usage	2	2	4	Relatively very low severity of failure results. Relatively very low risk of failure.
4. Sub-system for recording of communication times of the military units	2	2	4	Relatively very low severity of failure results. Relatively very low risk of failure.

3. The ranking process of modules priority is presented in the following table:

   The rating of severity level and risk level and the module testing priorities calculations are similar to those of section 2.

Sub-system (integration)	Severity level	Risk level	Priority rank	Assumptions
1.1 Unit that identifies humans according to a mathematical pattern recognition model	5	5	25 – I	Failure of this module, based on new technology, will endanger the soldiers that rely on it. Newly developed technology.
1.2 Unit for display of the identified humans on the screen of the “soldiers set”	5	3	15 – III	Failure of the display endangers the soldiers. The display technology is already well established.
1.3 Unit for calculating and displaying on the screen the distance to the identified humans	4	3	12 – IV	Once the image is displayed, the soldier may estimate the distance. The risk of failure seems moderate.
1.4 Unit for creating a display on the soldier’s screen of the combined identification data from all the unit's soldiers	5	5	25 – II	Failure of this module will endanger the soldiers that rely on it. Newly developed technology.

Sub-system (integration)	**Severity level**	**Risk level**	**Priority rank**	**Assumptions**
1.5 Unit for warning the soldier according to the combined identification data	3	3	9	Is an addition to the visual presentation so the failure results are moderate. The risk of development failure is moderate.
2.1 Unit for communicating the geographic location of the military unit	5	2	10 – V	Failure of the communication system will cause failure of the whole system. The module is based on established technologies.
2.2 Unit for communicating the identification of the observed human figures	5	2	10	Failure of the communication system will cause failure of the whole system. The module is based on established technologies.
2.3 Unit for performing the ciphered communication according to military communication standards	5	1	5	Failure of the communication system will cause failure of the whole system. The module is based on standard technologies.
3.1 Unit for recording set usage time by the soldier	2	2	4	Failure of this module does not affect military operations. The module is based on standard technologies.
3.2 Unit for daily reporting of the soldier's set usage according to the soldier's set identification number.	2	2	4	Failure of this module does not affect military operations. The module is based on standard technologies.
3.3 Unit for producing notices about soldier's sets that exceed 1000 hours of usage (for preventative maintenance)	3	2	6	Failure of the module may cause failure to perform preventive maintenance on time. The module is based on standard technologies.
4.1 Unit for recording communication times from the units	2	2	4	Failure of this module does not affect military operations. The module is based on standard technologies.

Sub-system (integration)	Severity level	Risk level	Priority rank	Assumptions
4.2 Unit for processing daily reports of when communications were received	2	2	4	Failure of this module does not affect military operations. The module is based on standard technologies.

In our case module 2.1 has the same priority as module 2.2. Module 2.2 may replace module 2.1 and become the fifth unit to be tested.

4. In cases where the high-severity high-risk module, that is planned to be unit-tested, is the only one in its integration, one can consider not testing the integration.

   In our case 4 out of the 5 units of integration 1 are included in the unit tests plan, and the remaining module 1.4 is of moderate severity and risk levels. One may therefore consider replacing integration 1 by integration 3 though its severity and risk levels are low.

10.2

H.C. – Hardware Center Ltd. – has developed a new billing software system to bill regular customers and H.C. credit cardholders, who include private as well as its corporate customers. Regular customers are billed as they complete their purchase; they are not entitled to discounts. All H.C. cardholders are entitled to a 4%–10% discount, depending on the purchase sum and the items, and are billed monthly. Corporate customers are entitled to additional 1%–5% discount, depending on their total purchases during the previous year.

The average monthly total number of bills is 30,000; 92% of the bills are regular customer purchase bills, 6% are monthly bills for private H.C. cardholders, and the rest are monthly bills for corporate H.C. cardholders. The testing unit has decided to use a regular random sample of 1,000 bills for its test of the new billing software. Mr Evans, the head of the SQA group, claims that a stratified random sample of 400 bills would be more effective for revealing the software errors and much cheaper to perform.

1. Do you agree with Mr Evans' claim? List your arguments.

2. If you agree with Mr Evans, describe the stratified sampling you would suggest and list your arguments for this choice.

3. List the assumptions that guided you in making your decisions.

Solution

1. According to the original testing plan a regular sample of 1000 will be applied. The sample includes 920 test cases, which refer to relatively simple calculations (no

discount, billed for each purchase and no monthly bill). The monthly bills for cardholders are much more complicated, but their testing is based on a relatively small number of test cases.

2. The following is a comparison of the regular sample with a stratified sample.

	Monthly bills	**Stratified sample**		**Regular sample test cases**
		% of sampling	**No. of test cases**	
Regular customers	27,600	0.5%	138	920
Private H.C. cardholders	1,800	10%	180	60
Corporate H.C. cardholders	600	13%	78	20
Total	30,000		396	1,000

The bills of the regular customers present a low error potential so it is possible to reduce drastically the sample size, from 3⅓% to ½%, with no real reduction of the possibility of detecting the errors. The sampling ratios for the monthly bills were raised to suit the higher error potential in these modules.

3. The assumptions:
   a. The monthly bills are much more complicated and justify a test case sample bigger than their proportion in the bills “population”.
   b. Repeated testing of simple test cases is actually a waste of testing resources.

10.3

Imagine the results of software system failure.

1. What are the main issues causing higher severity of failure?

2. Referring to your answer to (1), give 3 examples of software development projects that display the lowest severity of failure.

Solution

1. The main issues causing higher severity of failure are: on the one hand, risk to human life such as malfunctioning of intensive care equipment and on the other hand risk of paralyzing the activity of an organization such as a bank or production line.

2. Lowest severity projects: development of electronic games, library catalog applications and computerized control of advertisements display. It should be noted

that every failure causes damages to the developer and users, even if limited in its extent and in the number of people or organizations that are affected.

10.4

Chapter 2 of the STP (software test plan) and chapter 2 of the STD (software test description) are dedicated to the test environment (see Frames 10.2 and 10.3).

1. Discuss the alternative settings the planner can use, and explain the importance of professional planning of the test environment.

2. Suggest the risks incurred by inappropriate planning of the test environment.

Solution

1. The basic alternatives are the developer's site and the customer's site. Usually the developer's site is more convenient than the customer's. Normally, as the development has been performed in the developer's premises, the site is already extant, but changes are required to create a site that will simulate more accurately the customer's site. In other words, applying this alternative, the professional planning of the required hardware and operating systems configuration as well as the conditions and loads which are expected to prevail at the customer's site when the system is implemented to the customer's site is important. Applying the second alternative, namely performing the tests in the customer's site, is recommended for the system tests. However, care should be taken to perform the system tests under the operational conditions and configuration that are expected to prevail at the customer's site at the time of implementation of the new system. The tests are expected to reveal operational defects due to the inadequacy of the specific hardware and operating systems used by the customer as well as the adequacy of the new system's interfaces with the existing software systems and hardware.

2. Risks involved in inappropriate planning of the system tests environment:
   - System failures due to incompatibility of the new software system with the customer's operating system, communication system, etc.
   - System failures due to inability of the planned hardware and communication configurations to perform correctly in the specified maximum loads of users applications.
   - Waste of resources due to installing hardware and communication configurations of capacity much higher than necessary.

10.5

One would expect the STR (software test report) to be limited to a list of test results and some statistical summaries of the results. However, Chapters 1 and 2 of the report (see template in Frame 10.4) are divided into not less than 9 subsections devoted to the comprehensive description of the test, its site, the participants, and the test environment.

1. Refer to the 9 sections and explain the importance of the information to be reported in each.

2. Some of these sections provide information that could jeopardize the applicability of the test results. List the subjects and the circumstances in which doubt could be raised.

Solution

Chapters 1 and 2 of the STR template present 9 descriptions of the tests. Their importance is discussed:

1. **Test identification, site, schedule and participation**

   1.1 *The tested software identification (name, version and revision):* The accurate identification of the tested file enables the discovery of cases where an unexpected error list is caused by testing the wrong file.

   1.2 *The documents providing the basis for the tests (name and version for each document)*: As for 1.1

   1.3 *Test site:* Enables rechecking the configuration of the test file. Of special importance in cases of large scale projects where the tests are carried out at several different sites.

   1.4 *Initiation and concluding times for each testing session:* Enables identifying hardware and communication failures that were reported in the testing time and might affect the test results.

   1.5 *Test team members:* Enables inquiring about the test environment and specific tests in cases that the STR report does not supply sufficient information for special phenomena in the testing results.

   1.6 *Other participants:* As for 1.5

   1.7 *Hours invested in performing the tests:* Enables better scheduling and budgeting of future test plans.

2. **Test environment**

   2.1 *Hardware and firmware configurations:* Enables rechecking the adequacy of the site configuration to the situation at the customer's site. Also, enables identification of deviations in the actual configuration of the site from the planned site configuration.

   2.2 *Preparations and training prior to testing:* Enables checking if the planned training has been fully carried out according to plans. Also it allows collection of data about actual resources required and proper schedule for training.

   The following are situations, revealed according to subject mentioned above, that might jeopardize the test results

   – Performing the tests for a wrong test file

– Performing the tests for a wrong testing plan documentation

– Wrong hardware and firmware configurations

– Performing tests by a team not sufficiently trained.