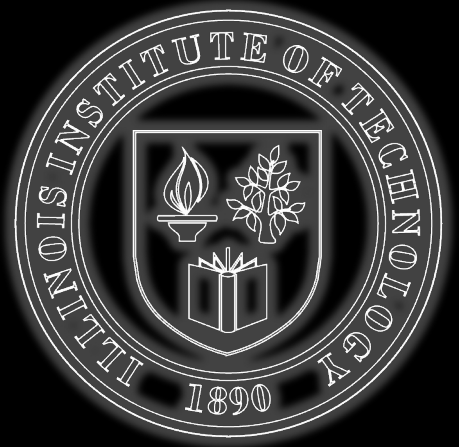
ILLINOIS TECH | College of Computing

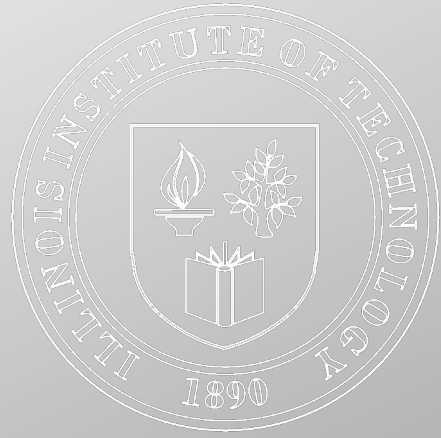
ITMD 536 Software Testing & Maintenance

Chapter 5
Test Management



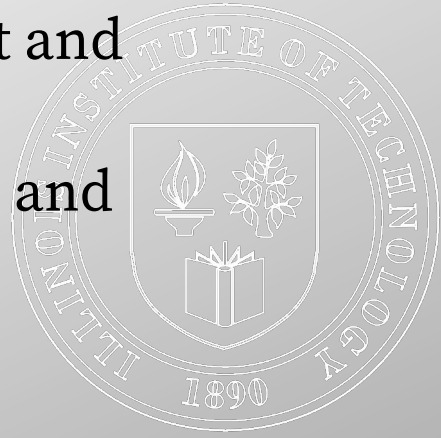
Objectives

- What is test organization?
- What is test management?
- What is test planning and estimation?
- What is test progress monitoring and control?
- What is configuration management?
- What is the risk and testing?
- What is incident management?



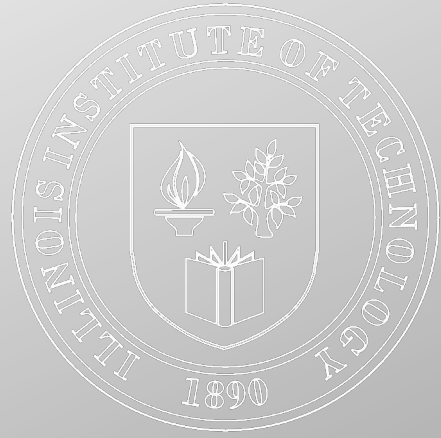
5. Test Management

- **Test Management:** The planning, estimating, monitoring and control of test activities, typically carried out by a test manager.
- First how to organize the testers and the testing.
- Second is how to plan and produce estimations, planning and strategizing of the test efforts.
- Third it addresses test progress monitoring, test reporting and test control.
- Fourth explains configuration management and relation to testing.
- Fifth covers the central topic of risk and how testing affects by product and project risks.
- Sixth it talks about the management of incidents, both product defects and other events that require further investigation.



5.1 Test Management

- **Test Management:** The planning, estimating, monitoring and control of test activities, typically carried out by a test manager.

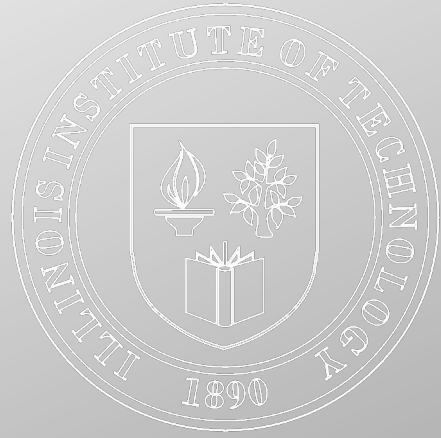


5.1 Test Organization

- **Test Manager (Test Leader):** The person responsible for project management of testing activities and resources, and evaluation of a test object. The individual who directs, controls, administers, plans and regulates the evaluation of a test object.
- **Tester:** A skilled professional who is involved in the testing of a component or system.

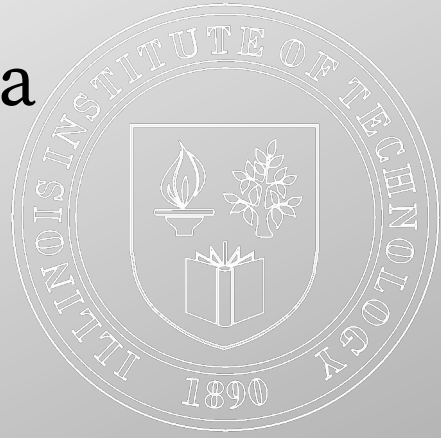
5.1.1 Independent and Integrated Testing

- **Independence Testing:** Independence is not an either/or condition, but a continuum.
- **Independent Testers:** Independent testers outside the development team, but they all report to the same project manager.



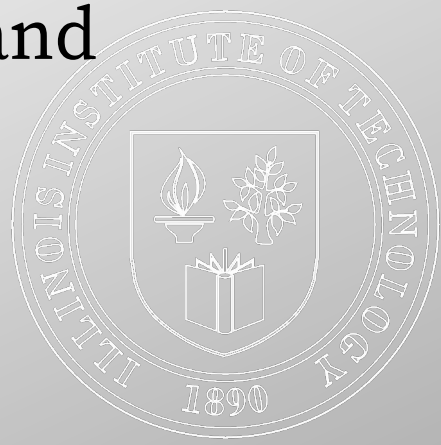
5.1.1 Independence Testing

- Benefits of Independence Testing:
  - Find different defects than a tester working within a programming team or a tester who is by profession a programmer (developer).
  - Find out hidden defects.
  - Tester training, testing tools, test equipment and it's a career path.



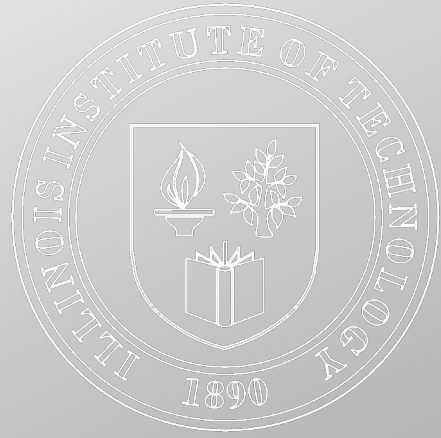
5.1.1 Independence Testing

- Risks as for Independent Testing:
  - Independent testers get isolated from the programmers, the designers etc.
  - Independ testers put more focus on defects and often refuses to accept the business prioritization of defects.
  - This creates communication issues between testers and developers.



5.1.1 Integrated Testing

- Integrated testing is the test manager manages the risks well with the independent testers.
- It is better to have independence in later levels of testing such as at (integration test, system test and acceptance test).

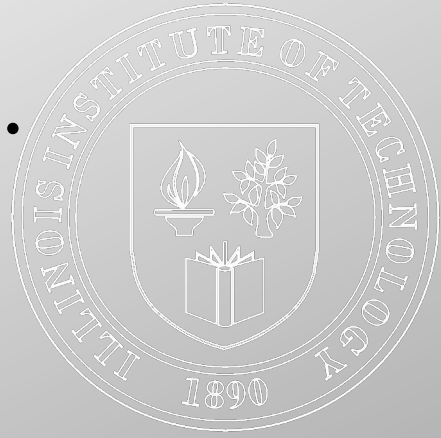


5.1.2 Working as a Test Lead

- Test Leaders are involved in the planning, monitoring and control of the testing activities and tasks.
- Test Leaders work with project managers, stakeholders to achieve the test objectives.
- They plan and estimate time, effort and cost is required and negotiate with management if necessary.

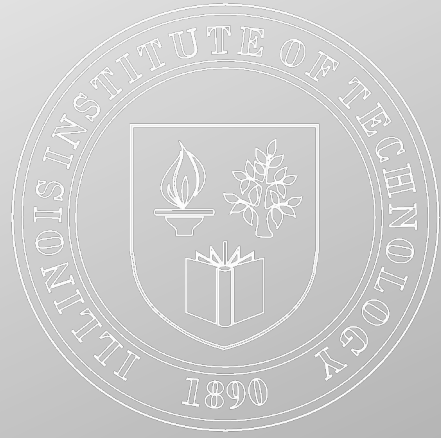
5.1.2 Working as a Test Leader

- Test leaders recognize if test automation is appropriate, plan the effort, select the tools, ensure the training is provided to the team.
- They lead, guide, monitor, analysis, design, implementation & execution of the test cases, test procedures and test suits.
- They ensure traceability of the tests to the test basis. (monitor, measure & control)



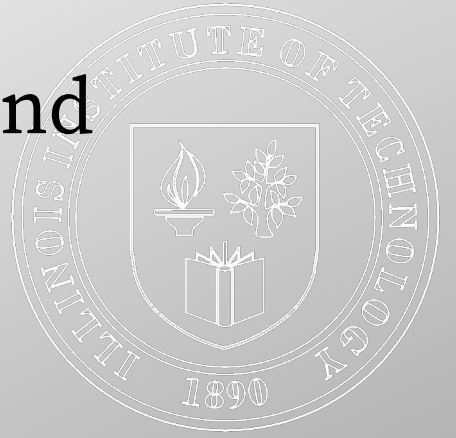
5.1.3 Working as a Tester

- Setup Test environment, assist system administration and network management.
- Tester execute and log the tests, evaluate the results and document problems found.
- Monitor testing, environment, and gather performance metrics.



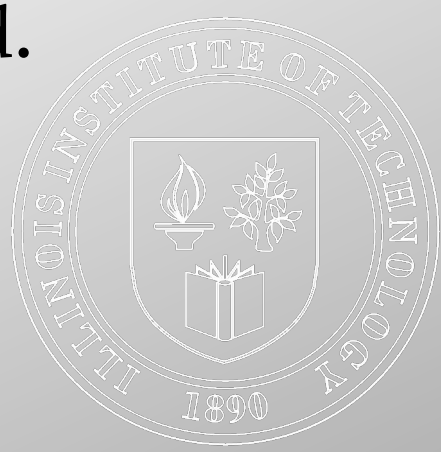
5.1.4 Defining the Skill Test Staff need

- Good test team have the right mix of skills based of the tasks and activities they need to carry out.
- Basic professional and social qualifications such as literacy, the ability to prepare and deliver written and verbal reports, ability to communicate effectively.
- **Application or business domain:** Tester needs to recognize and complete the ‘must work’ functions and features.



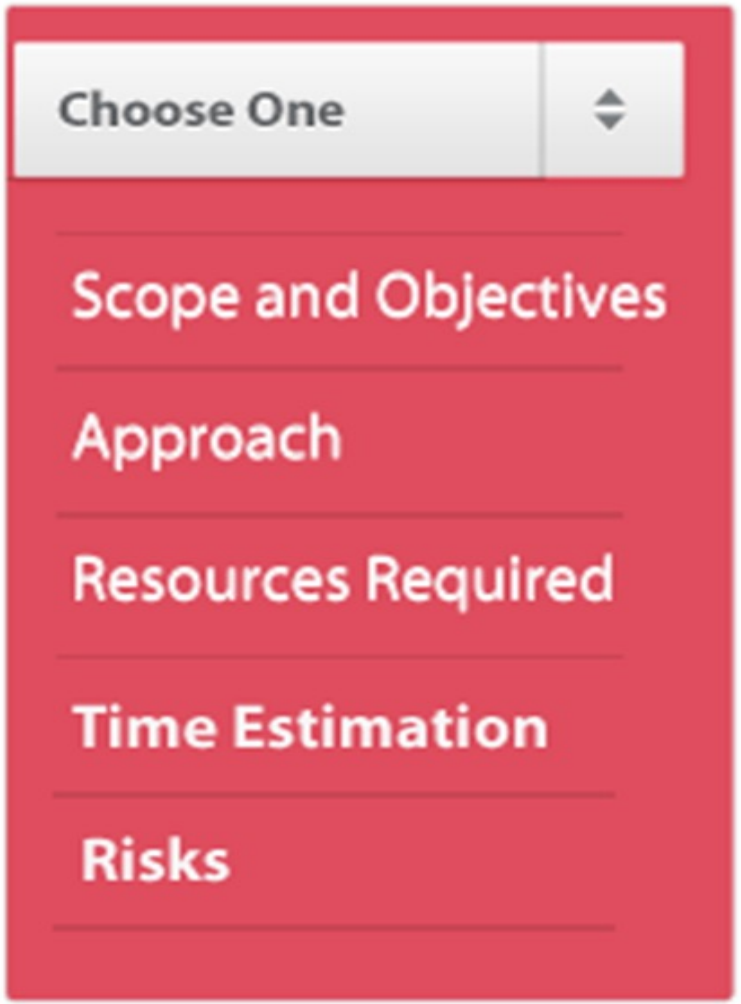
5.1.4 Defining the Skill Test Staff need

- **Technology:** Tester must be aware of issues, limitations and capabilities of the chosen implantation technology, in order to effectively and efficiently locate problems and recognize the 'likely to fail' functions and features.
- **Testing:** A tester must know the testing topics discussed in this book – and often more advanced testing topics – in order to effectively and efficiently carry out the test tasks assigned.
- Test automation is handled by testers who knows test automation.



5.2 Test Planning and Estimation

- **Test Planning:** The activity of establishing or updating a test plan.
- **Test Estimation:** The calculated approximation of a result related to various aspects of testing (e.g. effort spent, completion date, costs involved, number of test cases, etc.) which is usable even if input data may be incomplete, uncertain, or noisy.
- Test planning consist of writing a test plan, preparing an estimate and selecting test strategies tend to happen concurrently and ideally during the planning period for the overall project.



Test Plan Contents

What to test

What not to test

Test strategy

Entry criteria

Exit criteria

Metrics

Suspension criteria

Resources Required

Risks

Contingencies

Responsibilites

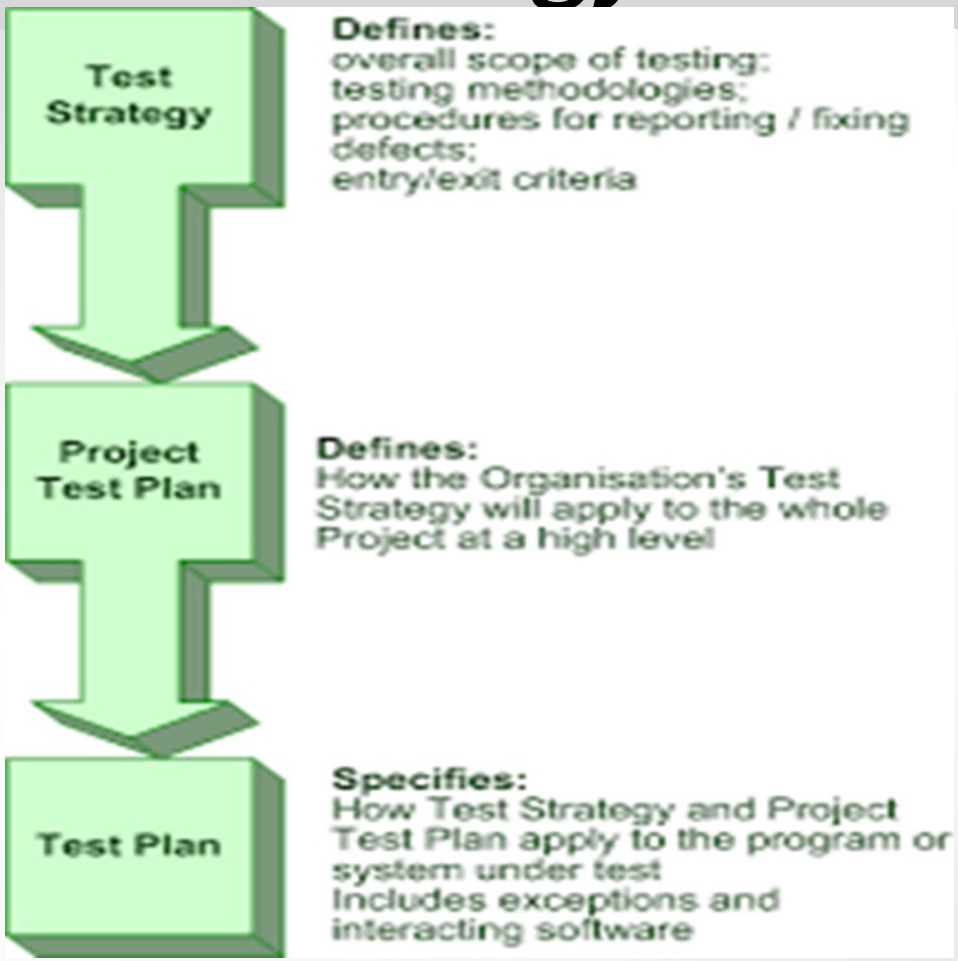
Resumption criteria

Release criteria

Defect logging

testnbug.com

Test Strategy and Test Plan



A Test Plan

- Test Planning at Each Phase of SDLC
- What is Included in a Test Plan
- Sample Test Plan Document

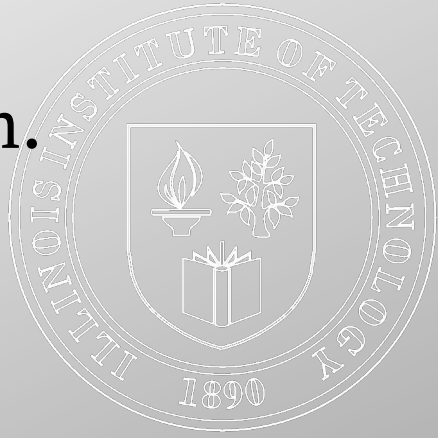
5.2.1 The Purpose and Substance of Test Plans

Why do we write a Test Plan?

The main reasons:

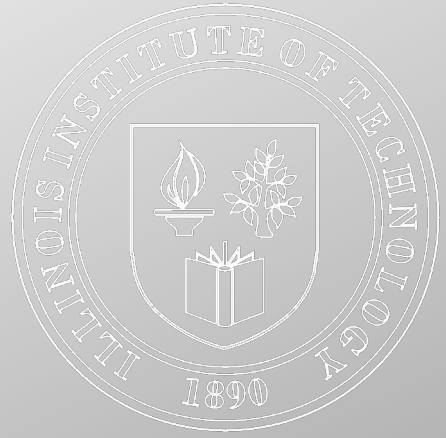
1. First, writing a test plan forces us to confront the challenges that await us and focus our thinking on important topics.

2. Using a test template when writing a test plans helps us to remember the important challenges. Use IEEE 829 test plan template or any other test plan template, or create your own.



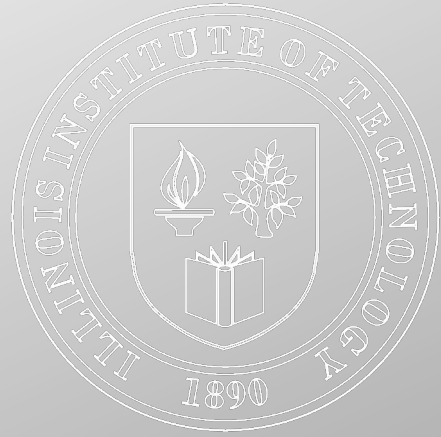
IEEE 829 Standard Test Plan Template

- Test plan identifier
- Introduction
- Test items
- Features to be tested
- Features not to be tested
- Approach
- Item pass/fail criteria
- Suspension-resumption
- Test deliverables
- Test tasks
- Environmental needs
- Responsibilities
- Staffing and training needs
- Schedule
- Risks & contingencies
- Approvals



IEEE 829 Format - Test Plan Outline

1. Test Plan Identified
2. References
3. Introduction
4. Test Items
5. Software Risk Issues
6. Features to be tested
7. Features not to be tested
8. Approach
9. Item Pass/Fail Criteria
10. Suspension Criteria and Resumption Requirements
11. Test Deliverables
12. Remaining Test Tasks
13. Environmental Needs
14. Staffing and Training Needs
15. Responsibilities
16. Schedule
17. Planning Risks and Contingencies
18. Approvals
19. Glossary



5.2.2 What to do with your brain while you planning tests

- Need to know following answers to create a test plan:
- What is in scope and what is out of scope for this testing effort?
- What are the test objectives?
- What are the important project and product risks?
- What constraints affect testing (budget limitations, hard deadlines, etc.)?
- What is most critical for this product and project?
- Which aspects of the product are more (or less) testable?
- What should be the overall execution schedule and how should we decide the order in which to run specific tests? (Product and planning risks, discussed later in this chapter, will influence the answers to the questions.)

5.2.2 What to do with your brain while you planning tests

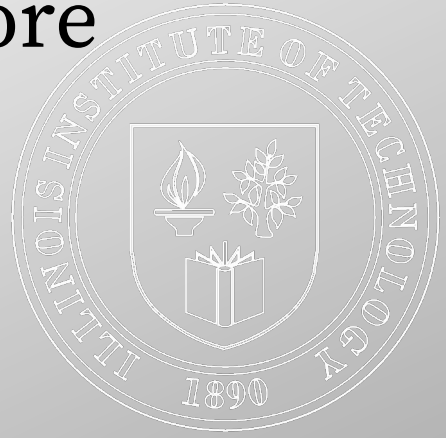
- Test '***entry criteria***':
- Acquisition and supply: the availability of staff, tools, systems, test environments, test data, and other materials required.
- Test items: the state that the items to be tested must be in to start and to finish testing.
- Test '***exit criteria***':
- Defects: the number known to be present, the arrival rate, the number predicted to remain, and the number resolved.
- Tests: the number prepared, run, passed, failed, blocked, skipped, and so forth.

5.2.2 What to do with your brain while you planning tests

- **Coverage:** the extent to which the test basis, risk, functionality, supported configurations, and the software code have been tested – or have not.
- **Quality:** the status of the important quality characteristics for the system, the estimated number of defects present or remaining, and other attributes.
- **Money:** the cost of finding the next defect in the current level of testing compared to the cost of finding it in the next level of testing (or in production).
- **Schedule:** the project schedule implications of starting or ending testing.
- **Risk:** the undesirable outcomes that could result from shipping too early (such as latent defects or untested areas) – or too late (such as loss of market share).
- *Successful project is a balance of quality, budget, schedule and feature considerations.*

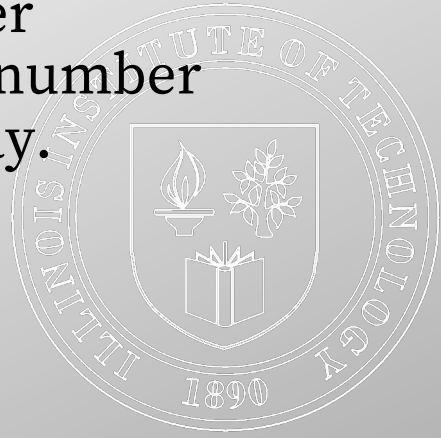
5.2.3 Estimating what will involve and what it will cost

- Test plan details will include performance testing information. The time and money estimation is required to include the performance testing.
- Performance testing is done by special skilled tester and will have additional cost and will require more time for testing.



5.2.4 Estimation Techniques

- There are two types of estimations:
- 1. Involves consulting the people who will do the work and other people with expertise on the tasks to be done.
- 2. Involves analyzing metrics from past projects and from industry data.
- Use Microsoft Project or whiteboard and sticky-notes the team can predict testing end-date and major milestones. This is called 'bottom-up' estimation.
- Analyzing metrics can be as simple or sophisticated as you make it. The simplest approach is to ask, 'How many testers do we typically have per developer on a project?' This is more reliable approach. This involves number of tests run by tester per day, number of defects found by tester per day.



Test Estimate in days using PERT formula				
	Optimistic (O)	Pessimistic (P)	Most likely (M)	PERT Estimate
				=(O+P+4*M)/6
User Requirement 1	6	12	8	8.33
User Requirement 2	12	20	15	15.33
User Requirement 3	30	45	33	34.50
Total Estimate in days	48	77	56	58.17

Top-Down, Bottom-Up and Phased Estimates

Focus on overall development process.

Generic Approach

- analogous method
- parametric method
- experience based

Focus on individual project process.

Task Based Approach

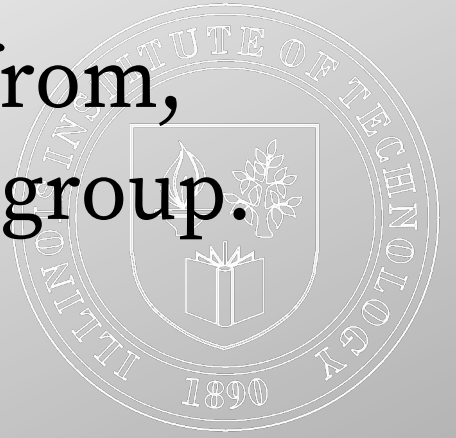
- add up WBS tasks.
- detailed estimating
- more definitive estimate

5.2.5 Factors Affecting Test Effort

- Product factors start with the presence of sufficient project documentation so that the testers can figure out what the system is, how it is supposed to work and what correct behavior looks like. Adequate and high-quality information about the test basis will help us do efficient job of defining the tests.
- The importance of non-functional quality characteristics such as usability, reliability, security, performance, and so forth also influences the testing effort. These test targets can be expensive and time consuming.

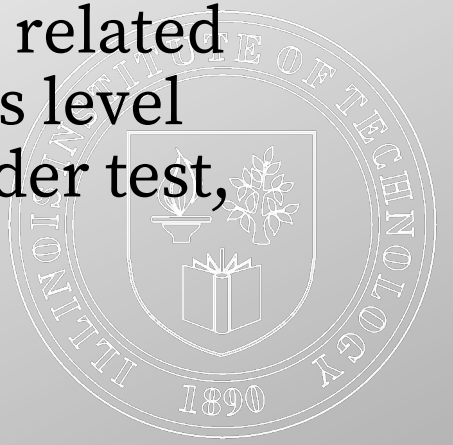
5.2.5 Factors Affecting Test Effort

- Solid relationships, reliable execution of agreed-upon commitments and responsibilities and a determination to work together towards a common goal are important.
- This is especially important for testing, where so much of what we test, use, and produce either comes from, relies upon or goes to people outside the testing group.



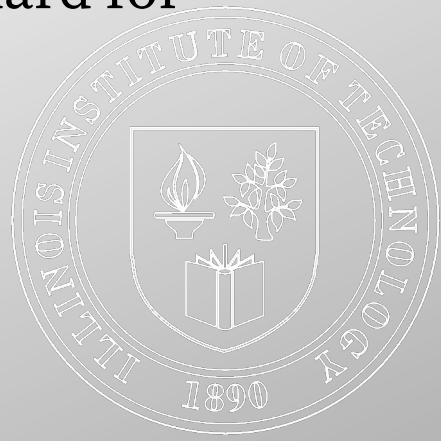
5.2.6 Test Approaches and Strategies

- **Test Strategy:** A high-level description of the test levels to be performed and the testing within the levels for an organization or program (one or more projects).
- A test strategy is the general way in which testing will happen, within each of the levels of testing, independent of project, across the organization.
- In your decision-making on the approach, you should take into account the project, product, and organization context, issues related to risks, hazards and safety, the available resources, the team's level of skills, the technology involved, the nature of the system under test, is it custom build or off the shelf.



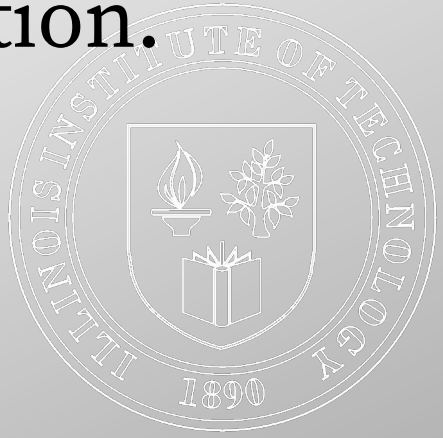
5.2.6 Test Approaches and Strategies

- ***Analytical:*** The risk-based strategy involves performing a risk analysis using project documents and stakeholder input, then planning, estimating, designing, and prioritizing the tests based on risk.
- Requirements-based strategy.
- ***Model-based:*** Build mathematical models for loading and response for e-commerce servers, and test based on that model.
- ***Methodical***: Checklist that you have put together over the years that suggests the major areas of testing to run or you might follow an industry-standard for software quality, such as ISO 9126, for your outline of major test areas.



5.2.6 Test Approaches and Strategies

- ***Process-or standard-compliant***: Adopt **IEEE 829** standard for testing or adopt an agile methodology.
- Process-or standard-compliant strategies have in common reliance upon an externally developed approach to testing with little – if any customization.

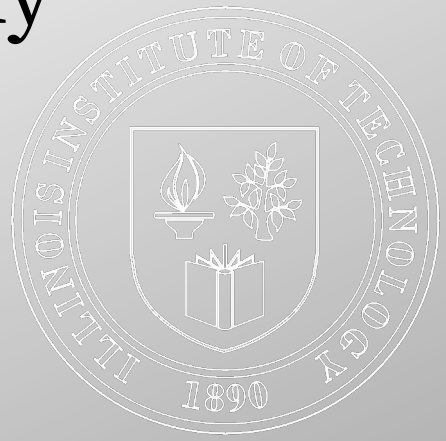


5.2.6 Test Approaches and Strategies

- ***Dynamic:*** Dynamic strategies, such as exploratory testing, have in common concentrating on finding as many defects as possible during test execution and adapting to the realities of the system under test as it is when delivered, and they typically emphasize the later stages for testing.
- Dynamic test strategies focus on the test execution period. This allows the location of defects and defect clusters that might have been hard to anticipate until you have the actual system in front of you. This is reactive approach.

5.2.6 Test Approaches and Strategies

- ***Consultative or directed:*** Consultative or directed strategies have in common the reliance on a group of non-testers to guide or perform the testing effort and typically emphasize the later stages of testing simply due to the lack of recognition of the value of early testing.

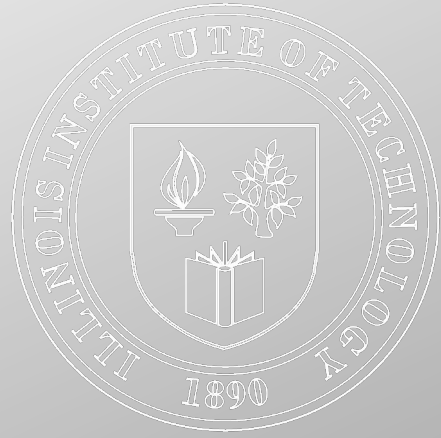


5.2.6 Test Approaches and Strategies

- ***Regression-averse:*** Automate all the tests of system functionality so that, whenever anything changes, you can re-run every test to ensure nothing has broken.
- Regression-averse strategies have in common a set of procedures – usually automated – that allow them to detect regression defects.
- Analytical test strategies involve upfront analysis of the test basis, and tend to identify problems in the test basis prior to test execution. This allows the early – and cheap – removal of defects. This is preventive approach.

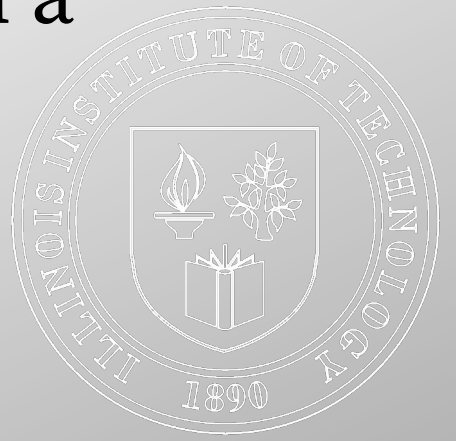
5.2.6 Test Approaches and Strategies

- How to pick which strategy?
- ***Risk:*** Testing is about risk management, so consider the risks and the level of risk.
- ***Skills:*** Strategies must not only be chosen, they must also be executed.



5.2.6 Test Approaches and Strategies

- ***Objectives:*** Testing must satisfy the needs of stakeholders to be successful.
  The objective is to find as many defects as possible with a minimal amount of up-front time and effort invested – for example at a typical test lab – then a dynamic strategy makes sense.

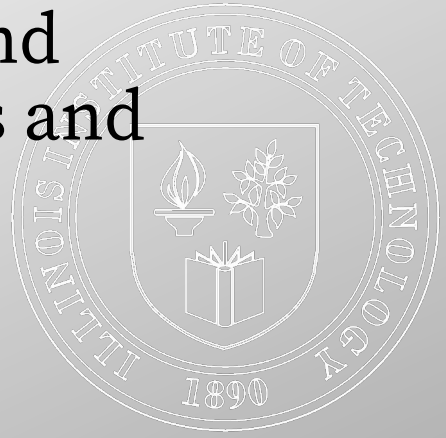


5.2.6 Test Approaches and Strategies

- ***Regulations:*** You not only have to satisfy the stakeholders, but also regulators. In this case, you may need to devise a methodical test strategy that satisfies these regulators that you have met all their requirements.
- ***Product:*** Some products such as weapons systems and contract-development software tend to have well-specified requirements. This lead to synergy with a requirements-based analytical strategy.
- ***Business:*** Business considerations and business continuity are often important. If you can use a legacy system as a model for a new system, you can use a model-based strategy.

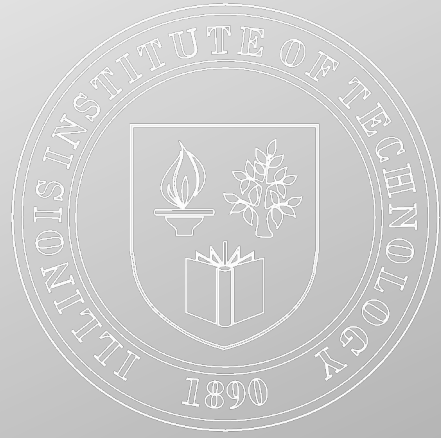
5.3 Test Progress and Monitoring

- **Test Progress Report:** A document summarizing testing activities and results, produced at regular intervals, to report progress of testing activities against a baseline (such as the original test plan) and to communicate risks and alternatives requiring a decision to management.
- Monitoring test preparation and execution.
- Interpretation of test metrics for reporting, controlling and analyzing the test effort, including those based on defects and those based on test data.



Test Plan

1. Introduction
2. Scope
3. Quality Risks
4. Schedule and Resources
5. Test Phase Transition Criteria
6. Test Environment and Configurations
7. Test Plan Execution
8. Test Phase Deliverables and Metrics
9. Risks and Contingencies



5.3.1 Monitoring the test progress of test activities

- Test monitoring can serve various purposes during the project, including following:
- Give the test team and the test manager feedback on how the testing work is going, allowing opportunities to guide and improve the testing and the project.
- ***Provide*** the project team with visibility about the test results.
- ***Measure*** the status of the testing, test coverage and the test items against the exit criteria to determine whether the test work is done.
- ***Gather data*** for use in estimating future test efforts.

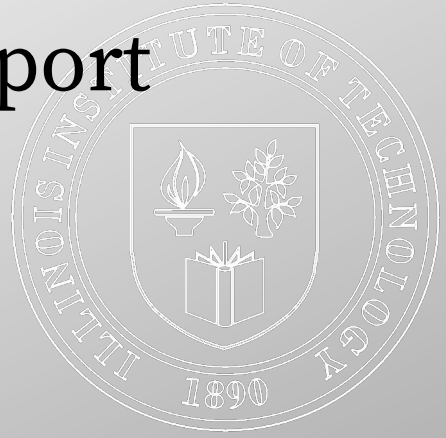
Test Case Summary Worksheet

	A	B	C	D	E	F	G	H	I	J	K	L	M
1				System Test Case Summary									
2				Cycle One									
3													
4	Test			System	Bug	Bug	Run	Plan	Act	Plan	Actual	Test	
5	ID	Test Suite/Case	Status	Config	ID	RPN	By	Date	Date	Effort	Effort	Duration	Comment
6													
7	1.000	*Functionality*											
8	1.001	File	Fail	A	701	1	LTW	1/8	1/8	4	6	6	
9	1.002	Edit	Fail	A	709	1	LTW	1/9	1/10	4	8	8	
10					710	5							
11					718	3							
12					722	4							
13	1.003	Font	Pass	B			IHB	1/10	1/10	4	4	4	
14	1.004	Tables	Warn	B	708	15	IHB	1/8	1/9	4	5	5	
15	1.005	Printing	Skip					1/10		4			Out of runway
16		**Suite Summary**						**1/10**	**1/10**	**20**	**23**	**23**	
17													
18	2.000	*Performance/Stress*											
19	2.001	Solaris Server	Warn	A,B,C	701	1	EM	1/10	1/13	4	8	24	Replan 1/11
20	2.002	NT Server	Fail	A,B,C	724	2	EM	1/11	1/14	4	4	24	Replan 1/12
21					713	2							
22					725	1							
23	2.003	Linux Server	Skip					1/12		4			Out of runway
24		**Suite Summary**						**1/12**	**1/14**	**12**	**12**	**48**	
25													
26	3.000	*Error Handling/Recovery*											
27	3.001	Corrupt File	Fail	A	701	1	LTW	1/8	1/9	4	8	8	
28					706	2							
29					707	4							
30					709	1							
31					710	5							
32					713	2							
33	3.002	Server Crash	Fail	A	712	6	LTW	1/9	1/10	4	6	6	
34					715	2							
35					717	1							
36		**Suite Summary**						**1/9**	**1/10**	**8**	**14**	**14**	
37													
38	4.000	*Localization*											
39	4.001	Spanish	Skip										
40	4.002	French	Skip										
41	4.003	Japanese	Skip										
42	4.004	Chinese	Skip										
43		**Suite Summary**						**1/8**	**1/8**	**0**	**0**	**0**	

FIGURE 5.1 Test case summary worksheet

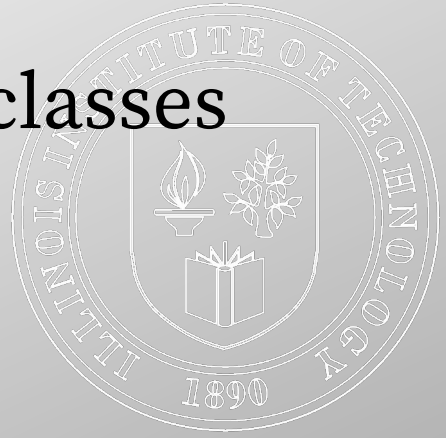
IEEE 829 Standard: Test Log Template

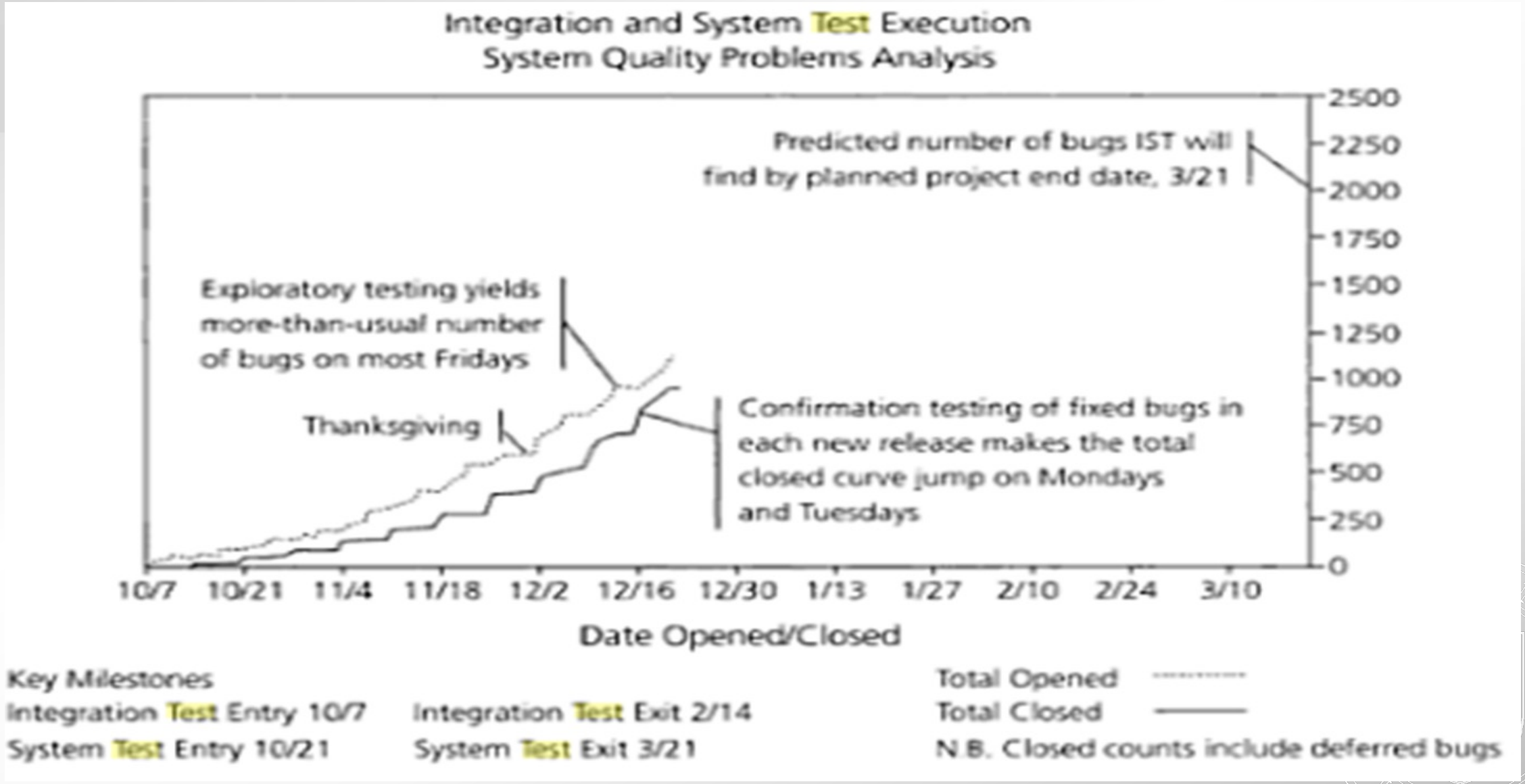
- Test log identifier
- Description (items being tested, environment in which the testing is conducted)
- Activity and event entries (execution description, procedure results, environmental information, anomalous events, incident report identifiers)



5.3.1 Monitoring the progress of test activities

- **Failure Rate:** The ratio of the number of failure of a given category to a given unit of measure, e.g. failures per unit of time, failures per number of transactions, failures per number of computer runs.
- **Defect Density:** The number of defects identified in a component or system divided by the size of the component or system (expressed in standard measurement terms, e.g. lines-of-code, number of classes or function points).




Integration and System Test Execution
System Quality Problems Analysis
Predicted number of bugs IST will find by planned project end date, 3/21
Exploratory testing yields more-than-usual number of bugs on most Fridays
Thanksgiving
Confirmation testing of fixed bugs in each new release makes the total closed curve jump on Mondays and Tuesdays
2500
2250
2000
1750
1500
1250
1000
750
500
250
0
10/7
10/21
11/4
11/18
12/2
12/16
12/30
1/13
1/27
2/10
2/24
3/10
Date Opened/Closed
Key Milestones
Integration Test Entry 10/7
System Test Entry 10/21
Integration Test Exit 2/14
System Test Exit 3/21
Total Opened
Total Closed
N.B. Closed counts include deferred bugs

5.3.1 Monitoring the progress of test activities

- The planned test period end date and the planned number of defects that will be found is listed in this template.
- All found defects are resolved and fixed prior to the release.

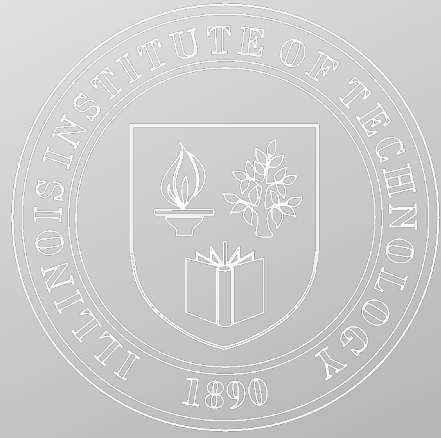
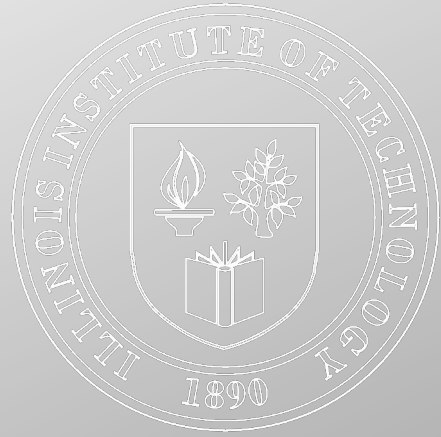


TABLE 5.1 Risk coverage by defects and tests

	Unresolved defects		Test cases to be run		
Test targets (product risk areas)	**#**	**%**	**Planned**	**Actual**	**%**
Performance, load, reliability	304	27	3843	1512	39
Robustness, operations, security	234	21	1032	432	42
Functionality, data, dates	224	20	4744	2043	43
Use cases, user interfaces, localization	160	14	498	318	64
Interfaces	93	8	193	153	79
Compatibility	71	6	1787	939	53
Other	21	2	0	0	0
	1107	***100***	***12857***	***5703***	***44***

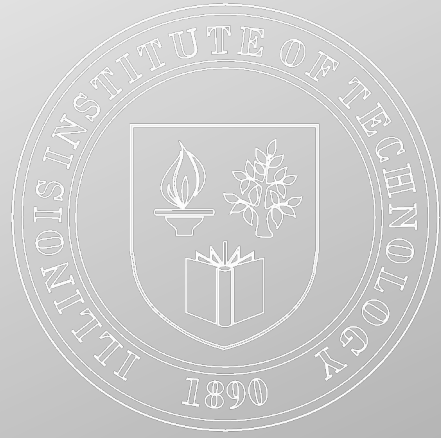
5.3.2 Reporting Test Status

- Test progress monitoring is about gathering detailed test data; reporting test status is about effectively communicating our findings to other project stake-holders.



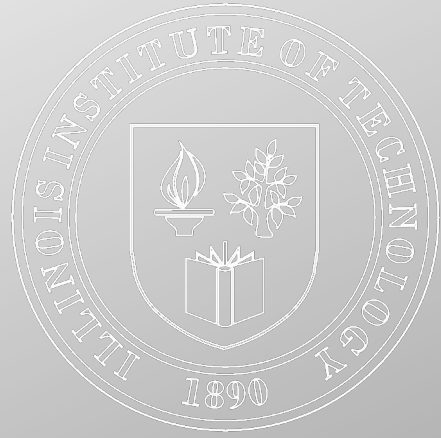
IEEE 829 Standard: Test Summary Report Template

- Test Summary report identifier
- Summary
- Variances
- Comprehensive assessment
- Summary of results
- Evaluation
- Summary of activates
- Approvals



5.3.3 Test Control

- Test control is about guiding and corrective actions to try to achieve the best possible outcome for the project.
- Software under test will be delivered late.
- Test control might involve re-prioritizing the tests.
- For cost reasons, performance testing is normally run on weekday evenings during off-hours in the production environment.
- At times due to production hours, performance test is rescheduled on weekends to have better test control.

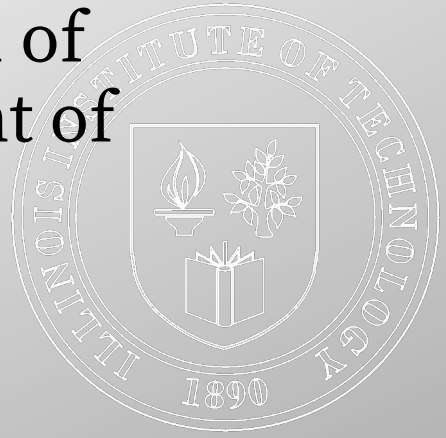


5.4 Configuration Management

- **Configuration Management:** A discipline applying technical and administrative direction and surveillance to: identify and document the functional and physical characteristics of a configuration item, control changes to those characteristics, record and report the change processing and implementation status, and verify compliance with specified requirements.

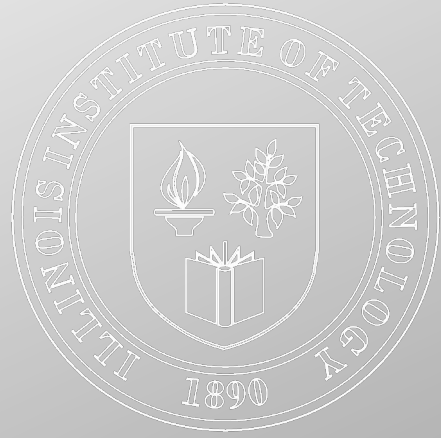
5.4 Configuration Management

- Configuration management is in part about determining clearly what the items are that make up the software a system. These items include source code, test scripts, third-party software (including tools, that support testing), hardware, data, and both development and test documentation.
- **Configuration Control (Version Control):** An element of configuration management, consisting of the evaluation, co-ordination, approval or disapproval, and implementation of changes to configuration items after formal establishment of their configuration identification.



IEEE 829 Standard: Test Items Transmittal Report Template

- Transmittal report identifier
- Transmitted items
- Location
- Status
- Approvals

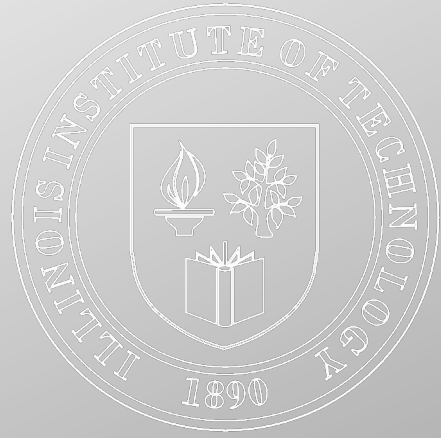


5.5 Risk and Testing

- **Risk:** A factor that could result in future negative consequence; usually expressed as impact and likelihood.
- **Testing:** Testing is a process that checks to see whether the solutions meets the design specifications and its free from errors.

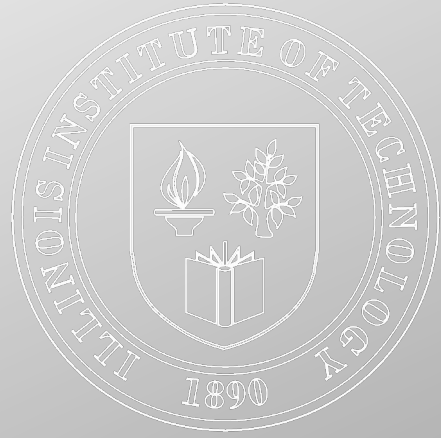
5.5 Risk and Testing

- Risk-based testing is the idea that we can organize our testing efforts in a way that reduces the residual level of product risk when the system ships. Risk-based testing use risk to prioritize and emphasize the appropriate tests during test execution.



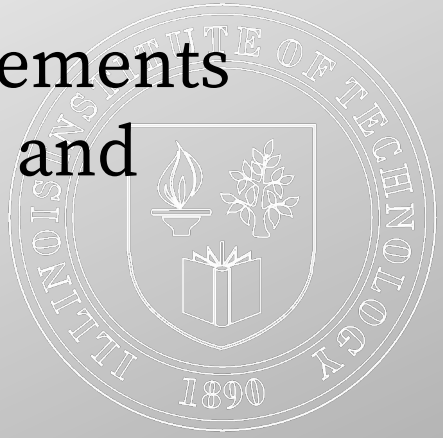
5.5.1 Risk and Levels of Risk

- Risk is a possibility not a certainty. It could vary from 0% to 100%.
- There are two risks:
- Product risk and
- Project risk



5.5.2 Product Risk

- **Product Risk:** A risk directly related to the test object.
- **Risk-based testing:** An approach to testing to reduce the level of product risks and inform stakeholders of their status, starting in the initial stages of a project. It involves the identification of product risks and the use of risk levels to guide the test process.
- Risk-based testing starts with product risk analysis. One technique for risk analysis is a close reading of the requirements specifications, design specifications, user documentation and other items.



5.5.2 Product Risk

- First you need to consider likelihood and the impact.
- Second risk analyzes, especially ones, are educated guesses. Follow the V-model you might perform the initial analysis during the requirements phase, then review and revise during the implementation phases, start with unit test, integration test, and system test.

5.5.2 Product Risk

Product risk	Likelihood	Impact	Risk priority #	Mitigation
Risk Category 1				
Risk 1				
Risk 2				
Risk n				

5.5.3 Project Risks

- **Project Risks:** A risk related to management and control of the (test) project, e.g. lack of staffing, strict deadlines, changing requirements, etc.
- Project risk is the possibility of a negative outcome.

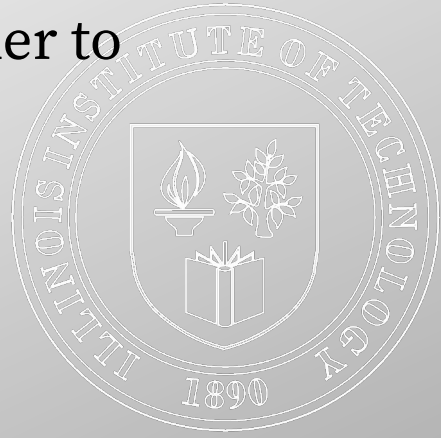
There are ***four options*** for any product or project risk:

Mitigate: Take steps in advance to reduce the likelihood of the risk.

Contingency: Have a plan in place to reduce the impact should the risk become an outcome.

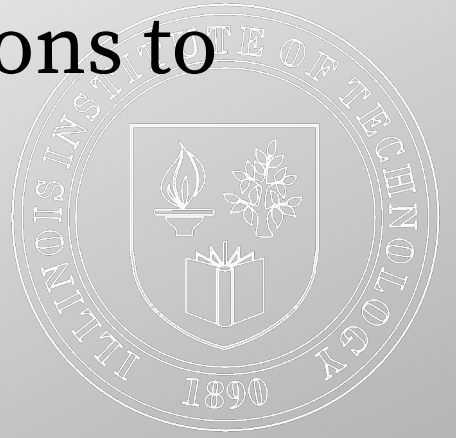
Transfer: Convince some other member of the team or project stakeholder to reduce the likelihood or accept the impact of the risk.

Ignore: Do nothing about the risk.



5.5.4 Trying it all together risk management

- The first step is to assess or analyze risks early in the project.
- Risk analyzes are educated guesses.
- Plan to re-assess and adjust risks at regular intervals in the project and make appropriate curse corrections to the testing or the project itself.

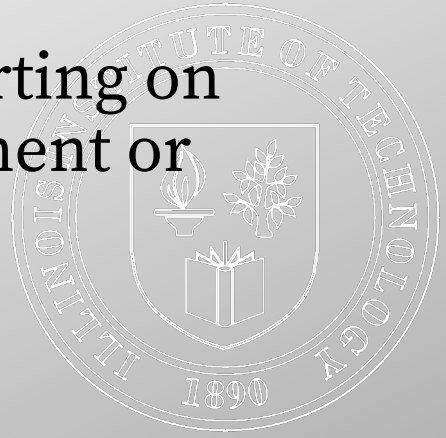


5.6 Incident Management

- **Incident Management:** The process of recognizing, investigating, taking actin and disposing of incidents. It involves logging incidents, classifying them and identifying the impact.
- One of the objectives of testing is to find defects, which reveals themselves as discrepancies between actual and expected results.

5.6.1 What are incident reports for and how do I write good ones?

- Major goals of testing is to find problems.
- They are also called incidents, bugs, defects, problems or issues.
- Other causes of incidents are misconfiguration or failure of test environment, corrupted test data, bed tests, invalid expected results and tester mistakes.
- **Incident Logging:** Recording the details of any incident that occurred, e.g. during testing.
- Defect report (bug report, problem report). A document reporting on any flaw in a component or system that can cause the component or system to fail to perform its required function.



5.6.1 What are incident reports for and how do I write good ones?

- **Defect Detection Percentage (DDP):** The number of defects found by a test phase, divided by the number found by that test phase and any other means afterwards.

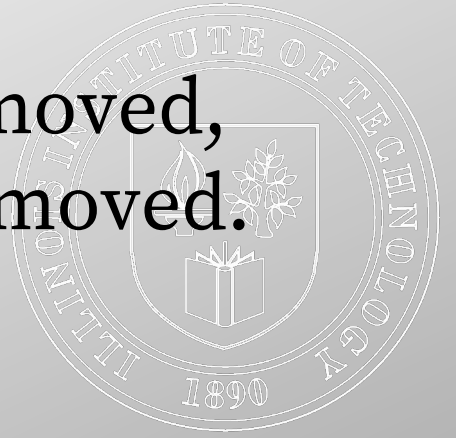
$$DDP = \frac{\text{Defects (testers)}}{\text{Defects (testers)} + \text{defects (field)}}$$

5.6.1 What are incident reports for and how do I write good ones?

- **Incident Report:** A document reporting on any event that occurred, e.g. during the testing, which requires investigation.
- Incident report contains a description of the misbehavior that was observed and classification of that misbehavior.
- **Priority:** The level of (business) importance assigned to an item e.g. defect.
- **Severity:** The degree of impact that a defect has on the development or operation of a component or system.

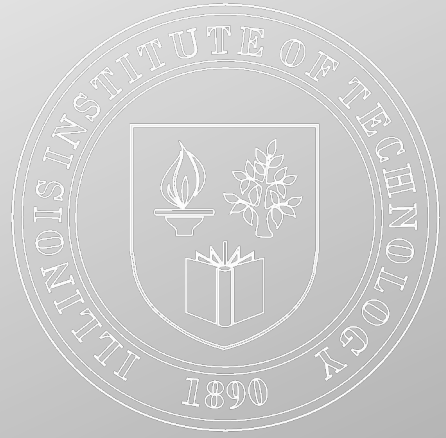
5.6.2 What goes on in an incident report?

- An incident report describes some situation, behavior or event that occurred during testing that requires further investigation.
- The programmer, when fixing the defect, can capture the root cause, the phase of introduction and the phase of removal.
- **Root cause:** A source of a defect such that if it is removed, the occurrence of the defect type is decreased or removed.



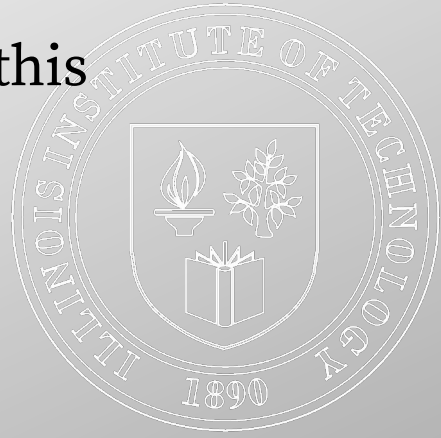
IEEE 829 Standard: Test Incident Report Template

- Test incident report identifier
- Summary
- Incident description (inputs, expected results, actual results, anomalies).
- Date and time, produce step, environment, attempts, to repeat, testers and observers)
- Impact



5.6.3 What happens to the incident reports after you file them?

- All incident reports move through a series of clearly identified states after being reported.
- Reported incident goes to developer and gets it fixed
- Fixed incident comes back to the tester for retesting
- Fixed incident will be closed
- If the incident fail again during retesting goes back to the programmer
- Until the incident is fixed the it goes back to the programmer
- If the management and stakeholders decides to postpone or not to fix this incident and they want to move on with the project.
- The incident is not fixed but it is documented for the future release.



Incident Report Life Cycle

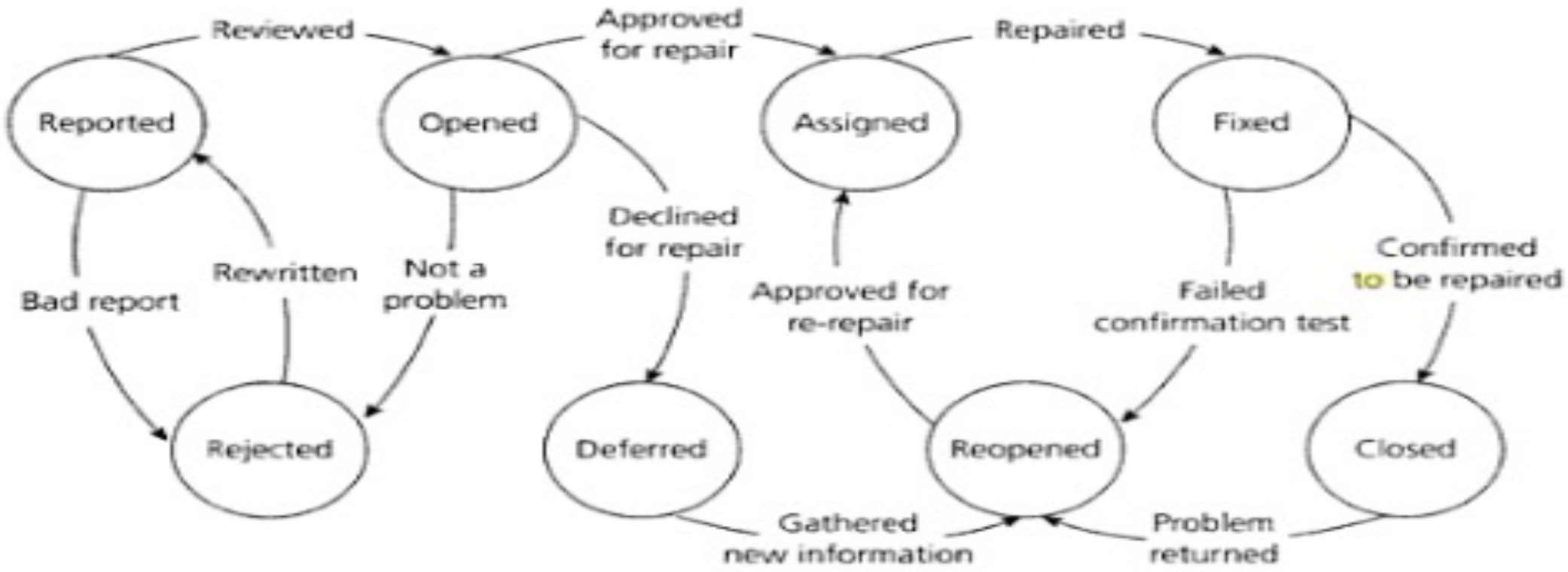


FIGURE 5.3 Incident report life cycle

5. Test Management – Question 1

1. Why is independent testing important?

a. Independent testing is usually cheaper than testing your own work.

b. Independent testing is more effective at finding defects.

c. Independent testers should determine the processes and methodologies used.

d. Independent testers are dispassionate about whether the project succeeds or fails.

5. Test Management – Question 2

2. Which of the following is among the typical tasks of a test leader?

a. Develop system requirements, design specifications and usage models.

b. Handle all test automation duties.

c. Keep tests and test coverage hidden from programmers.

d. Gather and report test progress metrics.

5. Test Management – Question 3

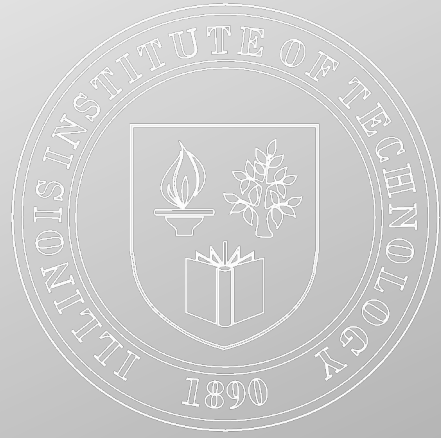
3. According to the ISTQB Glossary, what do we mean when we call someone a test manager

a. A test manager manages a collection of test leaders.

b. A test manager is the leader of a test team or teams.

c. A test manager gets paid more than a test leader.

d. A test manager reports to a test leader.



5. Test Management – Question 4

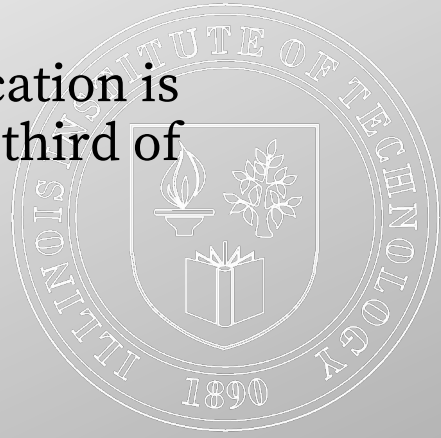
4. What is primary difference between the test plan, and the test design specification, procedure specification?

a. The test plan describes one or more levels of testing, the test design specification identifies the associated high-level test cases and a test procedures specification describes the actions for executing a test.

b. The test plan is for managers, the test design specification is for programmers and the test procedure specification is for testers who are automating tests.

c. The test plan is the least through, the test procedure specification is the most through and the test design specification is midway between the two.

d. The test plan is finished in the first third of the project, the test design specification is finished in the project and the test procedure specification is finished in the last third of the project.



5. Test Management – Question 5

5. Which of the following factors is an influence on the test effort involved in most projects?
a. Geographical separation of tester and programmers.
b. The departure of the test manager during the project.
c. The quality of the information used to develop the tests.
d. Unexpected long-term illness by a member of the project team.

5. Test Management – Question 6

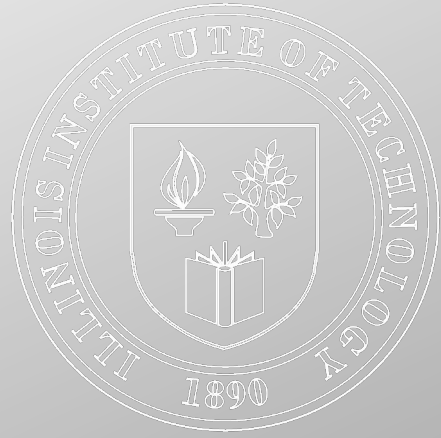
6. The ISTQB Foundation Syllabus establishes a fundamental test process where test planning occurs early in the project, while test execution occurs later. Which of the following elements of the test plan, while specified during test planning, are assessed during test execution?

a. Test tasks

b. Environmental needs

c. Exit criteria

d. Test team training



5. Test Management – Question 7

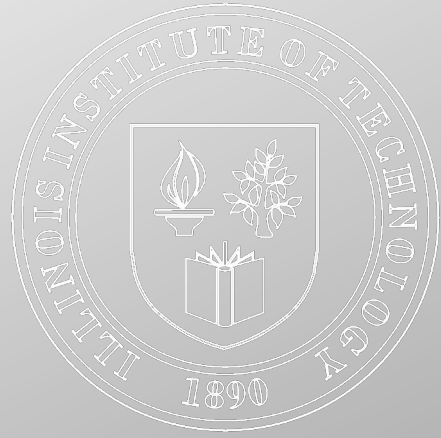
7. Consider the following exit criteria which might be found in a test plan:
I. No known customer-critical defects.
II. All interfaces between components tested.
III. 100% code coverage of all units.
IV. All specified requirements satisfied.
V. System functionality matches legacy system for all business rules.
Which of the following statements is true about whether these exit criteria belong in an acceptance test plan?

a. All statements belong in an acceptance test plan.

b. Only statement I belongs in an acceptance test plan.

c. Only statements I, II, and V belong in an acceptance test plan.

d. Only statements I, IV, and V belong in an acceptance test plan.



5. Test Management – Question 8

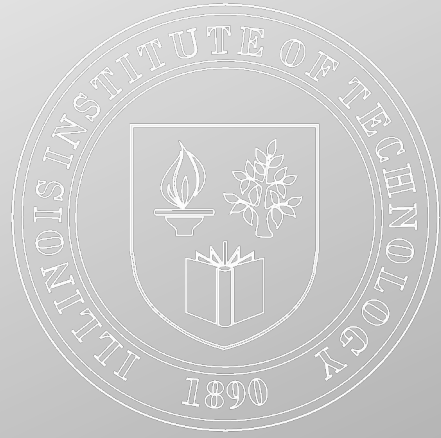
8. According to the ISTQB Glossary, what is a test level?

a. A group of test activities that are organized together.

b. One or more test design specification documents.

c. A test type.

d. As ISTQB certification.



5. Test Management – Question 9

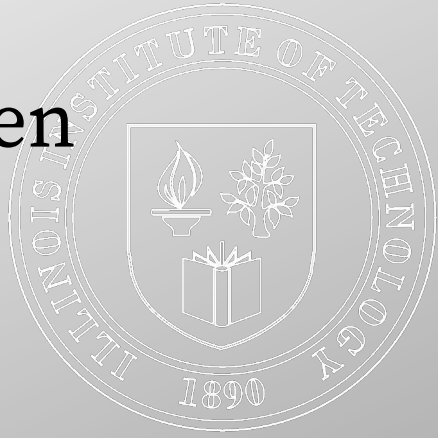
9. Which of the following metrics would be most useful to monitor during test execution?

a. Percentage of test cases written.

b. Number of test environments remaining to be configured.

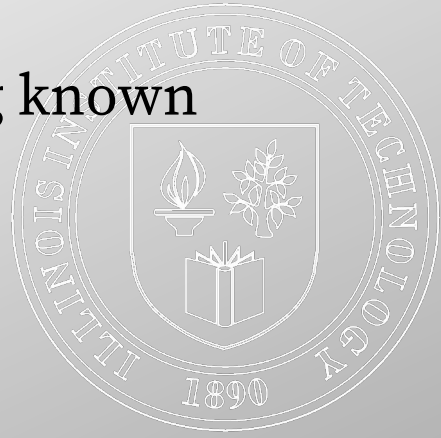
c. Number or defects found and fixed.

d. Percentage of requirements for which a test has been written.



5. Test Management – Question 10

10. During test execution, the test manager describes the following situation to the project team: '90% of the test cases have been run. 20% of the test cases have identified defects. 127 defects have been found. 112 defects have been fixed and have passed confirmation testing. Of the remaining 15 defects, project management has decided that they do not need to be fixed prior to release.' Which of the following is the most responsible interpretation of this test status report?
a. The remaining 15 defects should be confirmation tested prior to release.
b. The remaining 10% test cases should be run prior to release.
c. The system is now ready for release with no further testing or development effort.
d. The programmers should focus their attention on fixing the remaining known defects prior to release.



5. Test Management – Question 11

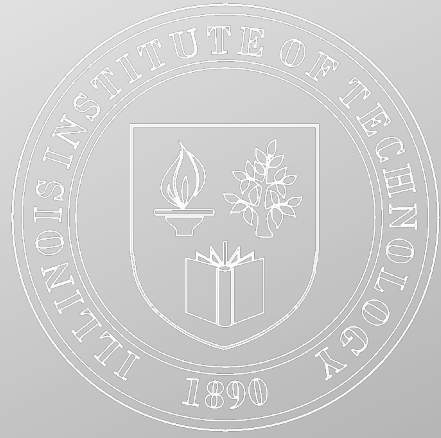
11. In a test summary report, the project's test leader makes the following statement, 'The payment from American Express cardholders, which is considered a must-work feature for this release'. The statement is likely to be found in which of the following sections?

a. Evaluation

b. Summary of activities

c. Variances

d. Incident description



5. Test Management – Question 12

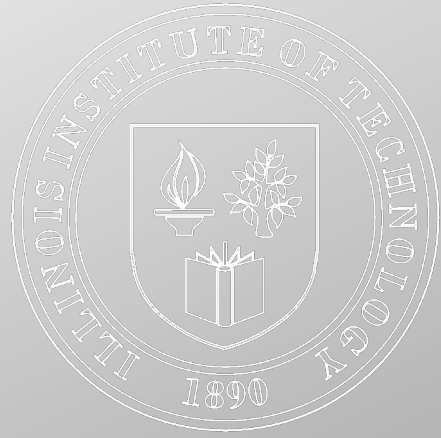
12. During an early period of test execution, a defect is located, resolved and confirmed as resolved by re-testing, but is seen again later during subsequent test execution. Which of the following is a testing-related aspect of configuration management that is most likely to have broken down?

a. Traceability

b. Confirmation testing

c. Confirmation control

d. Test documentation management



5. Test Management – Question 13

13. You are working as a tester on a project to develop a point-of-sales system for grocery stores and other similar retail outlets. Which of the following is a product risk for such a project?

a. The arrival of a more-reliable competing product on the market.

b. Delivery of an incomplete test release to the first cycle of system test.

c. An excessively high number of defect fixes fail during re-testing.

d. Failure to accept allowed credit cards.

5. Test Management – Question 14

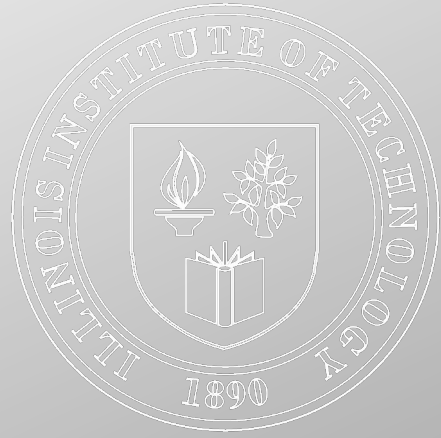
14. A product risk analysis meeting is held during the project planning period. Which of the following determines the level of risk?

a. Difficulty of fixing related problems in code.

b. the harm that might result to the user.

c. The price for which the software is sold.

d. The technical staff in the meeting.



5. Test Management – Question 15

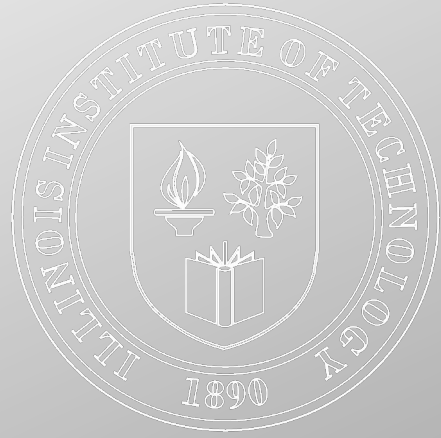
15. You are writing a test plan using the IEEE 829 template and are currently completing the Risks and Contingencies section. Which of the following is most likely to be listed as a project risk?

a. Unexpected illness of a key team member.

b. Excessively slow transaction-processing time.

c. Data corruption under network congestion.

d. Failure to handle a key use case.



5. Test Management – Question 16

16. You and the project stakeholders develop a list of product risks and project risks during the planning stage of a project. What else should you do with those lists of risks during test planning?

a. Determine the extent of testing required for the product risks and the mitigation and contingency actions required for the project risks.

b. Obtain the resources needed to completely cover each product risk with tests and transfer responsibility for the project risks to the project manager.

c. Execute sufficient tests for the product risks, based on the likelihood and impact of each product risk and execute mitigation actions for all project risks.

d. No further risk management action is required at the test planning stage.

5. Test Management – Question 17

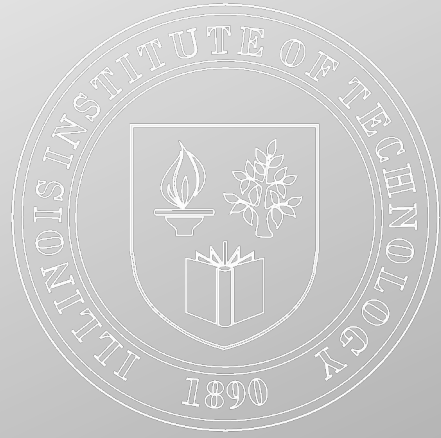
17. According to the ISTQB Glossary, a product risk is related to which of the following?

a. Control of the test project

b. The test object

c. A single test item

d. A potential negative outcome



5. Test Management – Question 18

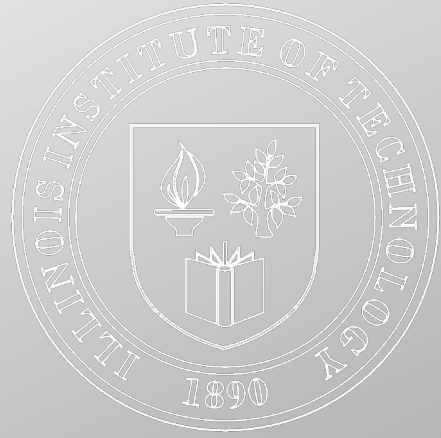
18. In an incident report, the tester makes the following statement, ‘At this point, I expect to receive and error message explaining the rejection of this invalid input and asking me to enter a valid input. Instead the system accepts the input, displays an hourglass for between one and five seconds and terminates abnormally, giving the message, “Unexpected data type: 15 Click to continue”.’ This statement is likely to be found in which of the following sections of an IEEE 829 standard incident report?

a. Summary

b. Impact

c. Item pass/fail criteria

d. Incident description



5. Test Management – Question 19

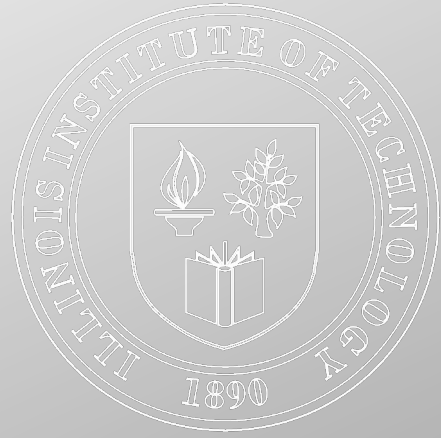
19. According to the ISTQB Glossary, what do we call a document that describes any event that occurred during testing which requires further investigation?

a. Bug report

b. A defect report

c. An incident report

d. A test summary report



5. Test Management – Question 20

20. A product risk analysis is performed during the planning stage of the test process. During the execution stage of the test process, the test manager directs the testers to classify each defect report by the known product risk it relates to (or to 'other'). Once a week, the test manager runs a report that shows the percentage of defects related to each known product risk and to unknown risks. What is one possible use of such a report?

a. To identify new risks to system quality.

b. To locate defect clusters in product subsystems.

c. To check risk coverage by tests.

d. To measure exploratory testing.