5. TEST MANAGEMENT

A. Test Organisation

i. Roles within a project team:

- Project Managers
- Quality Assurance Managers
- Developers
- Business and Domain Experts (SME, business analyst)
- Infrastructure Personnel (database designers etc)
- IT Operations (Help desk, merge functionality etc)
- **Tester**
- **Test Manager**

ii. Test Manager vs tester:

The Tester does the tests, the Test Lead manages and reports:

Test Manager	Tester
May take many forms such as Project Manager, Development Manager, QA Manager, Manager of Test Group (depending on the business)	May take many forms such as Developer, Business Analyst, User, SME, Specialists (depending on the business)
Develop or review a test policy and test strategy for the organization. Manage costs and time	Analyse, review, and assess requirements, user stories and acceptance criteria, specifications, and models for testability (i.e., the test basis)
Plan the test activities and understand test objectives	Analyse, review, and assess requirements, user stories and acceptance criteria, specifications, and models for testability (i.e., the test basis)
Write and update Test Plans	Review and contribute to test plans
Initiate the analysis, design, implementation, and execution of tests, monitor test progress and results, and check the status of exit criteria	Design and implement test cases and test procedures and create test data
Create Test Progress reports for Stakeholders	Create test execution schedule
Introduce suitable metrics for measuring test progress and evaluating the quality of the testing and the product	Execute tests, evaluate the results, and document deviations from expected results
Decide about the implementation of test environment/s	Design, set up, and verify test environment/s
Support the selection and implementation of tools to support the test process	Use appropriate tools to facilitate the test process

iv. Independence

The more removed you are from the code the less bias you will have when testing the code. There are several **levels of independent testing** within an organisation (see right)

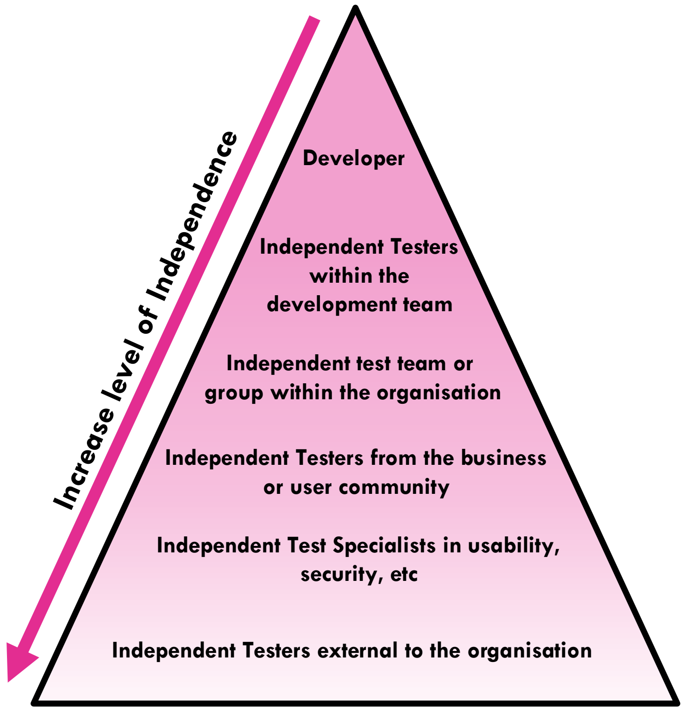
The **advantages** and **disadvantages** of having independent testers:

Advantages

- See defects that others who are closer to the project may not. They should be unbiased
- Can verify assumptions made during the specification and implementation phases

Disadvantages

- Can be seen as not part of the project and suffer from isolation
- Can be easily blamed for delays and targeted as delivery bottlenecks
- Developers may no longer feel responsible for their mistakes


Increase level of Independence
Developer
Independent Testers within the development team
Independent test team or group within the organisation
Independent Testers from the business or user community
Independent Test Specialists in usability, security, etc
Independent Testers external to the organisation

B. Test planning & estimation

i. Purpose of a Test Plan

- Determine objective, scope, who is doing what.
- Timescale and budget
- Selecting metrics for test monitoring and control

ii. Test Strategies

ANALYTICAL: Based on some type of analysis e.g. Risk or requirements based

MODEL BASED: Tests are designed based on some model of some required aspect of the product, such as a function, a business process, an internal structure, or a non-functional characteristic (e.g., reliability) i.e. stochastic testing

METHODOLICAL: Checklist & failure-based

PROCESS COMPLIANT: Involves analysing, designing, and implementing tests based on external rules and standards, such as those specified by industry-specific standards

REACTIVE: Heuristic, exploratory testing. Exploratory techniques usually used.

DIRECTED/CONSULTATIVE: Technology and/or business domain experts outside or within the test team

REGRESSION AVERSE: Highly automated. Avoids regression of existing capabilities.

iii. Entry & Exit Criteria

Entry and exit criteria should be defined for each test level and test type and will differ based on the test objectives.

Entry Criteria: Specific conditions or on-going activities that must be present before a process can begin (e.g. availability of test environment).

Exit Criteria: What conditions must be achieved in order to declare a test level or a set of tests completed. (e.g. Planned tests have been executed)

iv. Test Execution Schedule

A schedule for the execution of test suites within a test cycle. test cases would be ordered to run based on their priority levels, usually by executing the test cases with the highest priority first. However, this practice may not work if the test cases have dependencies or the features being tested have **dependencies**.

v. Factors influencing Test Effort

- Product Characteristics (e.g. test base quality, size of product, product risks)
- Development Process Characteristics (e.g. tools used, test approach, time)
- People Characteristics (e.g. skills)
- Test Results (e.g. no. and severity of defects)

C. Test Monitoring & Control

i. Purpose, of Test reports

The purpose of test reporting is to summarize and communicate test activity information, both during and at the end of a test activity (e.g., a test level)

ii. Test Progress Reports:

- Summary of testing performed
- What occurred during a test period
- Deviations from plan
- Status of testing and product quality with respect to the exit criteria or definition of done
- Factors that have blocked or continue to block progress
- Metrics of defects, test cases, test coverage, activity progress, and resource consumption
- Residual risks
- Reusable test work products produced

E. Defect Management

Defect Management is the process of recognising, investigating, taking action and disposing of defects. A defect is anything when the **actual result** is different from the **expected result**. An example of the contents of a defect report can be found in ISO standard (ISO/IEC/IEEE 29119-3) called incident reports.

Defect report include:

1. **Test incident report identifier & Date**
2. **Title & Short Summary (include screenshot)**
3. **Incident description (Expected results, Actual results, Anomalies, Date and Time, Procedure step, attempts to repeat, Testers comments, observers comments)**
4. **Impact**
5. **Urgency to fix**
6. **State of the defect report (e.g., open, deferred, duplicate, waiting to be fixed, awaiting confirmation testing, re-opened, closed)**
7. **Conclusions, recommendations and approvals**

D. Configuration Management

The purpose of configuration management is to establish and maintain the integrity of the products (components, data and documentation) of the software or system through the project and product life cycle. During the defect management process, some of the reports may turn out to describe false positives, not actual failures due to defects.

For testing, configuration management may involve ensuring the following:

- All items of testware are identified, **version controlled**, tracked for changes, related to each other and related to development items (test objects) so that **traceability** can be maintained throughout the test process
- All identified documents and software items are referenced unambiguously in test documentation

For the tester, configuration management helps to uniquely identify (and to reproduce) the tested item, test documents, the tests and the test harness(es).

During test planning, the configuration management procedures and infrastructure (tools) should be chosen, documented and implemented.

F. Risks Management

Risk – A factor that could result in future negative consequences; usually expressed as **impact** and **likelihood**.

Risk Analysis – The process of assessing identified risks to estimate their impact and probability of occurrence (likelihood).

Product Risk – the possibility that the system or software might fail to satisfy or fulfil some reasonable expectation of the customer, user, or stakeholder. (Some authors also call ‘Product risks’ ‘Quality risks’). Affects **Quality** of product. i.e. software not performing intended function

Project Risk – A risk related to management and control of the (test) project, e.g. lack of staffing, strict deadlines, changing requirements, political issues, supplier contracts etc.

Risk-based 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 (i.e. mitigate risk) when the system is deployed (e.g. tailor test approaches, test coverage required, prioritise testing).