ISO about CTFL

- ISO29119 -> SW Test (Dynamic)
- ISO25010 -> Quality
- ISO20246 -> Static Test

Seven Testing Principles

- Testing shows the presence, not the absence of defects
- Exhaustive testing is impossible
- Early testing saves time and money
- Defects cluster together -> 缺陷通常群聚在某一小塊功能
- Tests wear out / Pesticide paradox -> 反覆執行相同的測試會越來越無效
- Testing is context dependent -> 測試須依情境而異
- Absence-of-defects fallacy -> 無failure的軟體也可能無法滿足user需求

Test Activities

Test planning		test plan / schedule / criteria / risk
Test monitoring and control	度量指標	test progress report / risk
Test analysis	Static Test	test condition / bug report
Test design		test cases / testware
Test implementation		test env. / testsuites / script / test data
Test execution		test log / bug report
Test completion		test complete report

Test Levels

1. Components Test (Dev) -> Auto
2. Integration Test (Dev) -> CI/CD
3. System Test (Tester) -> Manua & Auto
4. System Integration Test (Tester) -> Manual & Auto
5. Acceptance Test (User) -> Manual & Tool

Quality Aspect

- Function
- Reliability
- Usability
- Perfoemance
- Maintainability
- Portability
- Security
- Compatibility

SLDC

- Sequential -> Waterfall, V model
- Iterative / Incremental -> Rational Unified Process, Prototyping, Scrum & Kanban

Refactor

降低複雜度，屬於Test Driven Development的過程之一

Acceptance testing

- User aspect
- Validation
- 增進信心
- 4種類型
    - user acceptance testing (UAT)
    - operational acceptance testing
    - contractual and regulatory acceptance testing,
    - alpha testing and beta testing
        - Alpha -> test in dev. env.
        - Beta -> test in user env.

Test Type

- Functional testing (function test in quality aspect)
- Non-functional testing (other tests in quality aspect)
- Black-box testing -> easy to build CI/CD
- White-box testing

Confirmation & Regression

- Confirmation (retest) -> 驗證bug是否被修復
- Regression -> 驗證其他地方是否被這版fix給影響 -> 需透過impact analysis確認範圍

Review Process Activities (靜態測試中的一種子項)

1. Planning
2. Review initiation
3. Individual review
4. Communication and analysis
5. Fixing and reporting

Review Types	Roles	Document	Process Activities
Informal review	Author Reviewer		Planning Review initiation
Walkthrough	Author Reviewer Scribe(recorder, should be objective)		Planning Review initiation
Techmical review	Author Reviewer Scribe Moderator(facilitator)	Issue log Bug report	Planning Review initiation Individual review Communication and analysis Fixing and reporting
Inspection	Author Reviewer Scribe Moderator Review leader Manager	Issue log Bug report Checklist Matrix	Planning Review initiation Individual review Communication and analysis Fixing and reporting

Success Factors for Reviews

- Defining clear objectives and measurable exit criteria. Evaluation of participants should never be an objective
- Choosing the appropriate review type to achieve the given objectives, and to suit the type of work product, the review participants, the project needs and context
- Conducting reviews on small chunks, so that reviewers do not lose concentration during an individual review and/or the review meeting (when held)
- Providing feedback from reviews to stakeholders and authors so they can improve the product and their activities
- Providing adequate time to participants to prepare for the review
- Support from management for the review process
- Making reviews part of the organization’s culture, to promote learning and process improvement
- Providing adequate training for all participants so they know how to fulfill their role
- Facilitating meetings

Test Techniques

Test Type		Coverage items (通常是覆蓋率的分母)
Black Test	Equivalence Partitioning (EP)	Equivalence Type
	Boundary Value Analysis (BVA)	Boundaries ● 2-value ● 3-value
	Decision Table Testing	Decision Rules
	State Transition Testing	● All States ● Valid Transitions ● All Transitions
White Test	Statement testing	All Statements (每行程式碼)
	Branch testing	All Branchs
Experience-based	Error Guessing	Fault Attack
	Exploratory Testing	designed executed evaluated unscripted test session-based test -> follow test charter
	Checklist-Based Testing	

Collaboration-based Test

- Collaborative User Story -> by customer, dev. & tester
    - Independent
    - Negotiable
    - Valuable
    - Estimable
    - Small
    - Testable
- Acceptance
    - Define the scope of the user story
    - Reach consensus among the stakeholders
    - Describe both positive and negative scenarios of test
    - As a basis for the user story acceptance testing
    - Accurate planning and estimation
    - Two formats to write acceptance criteria for a user story
        - Scenario-oriented -> BDD, ATDD
        - Rule-oriented -> build rules list

Test Plan Column

- Context of testing (e.g., scope, test objectives, constraints, test basis)
- Assumptions and constraints of the test project
- Stakeholders (e.g., roles, responsibilities, relevance to testing, hiring and training needs)
- Communication (e.g., forms and frequency of communication, documentation templates)
- Risk register (e.g., product risks, project risks)
- Test approach (e.g., test levels, test types, test techniques, test deliverables, entry criteria and exit criteria, independence of testing, metrics to be collected, test data requirements, test environment requirements, deviations from the organizational test policy and test strategy)
- Budget and schedule

Test Progress Report Column

- Test period
- Test progress (e.g., ahead or behind schedule), including any notable deviations
- Impediments for testing, and their workarounds
- Test metrics (see section 5.3.1 for examples)
- New and changed risks within testing period
- Testing planned for the next period

Test Completion Report Column

- Test summary
- Testing and product quality evaluation based on the original test plan (i.e., test objectives and exit criteria)
- Deviations from the test plan (e.g., differences from the planned schedule, duration, and effort).
- Testing impediments and workarounds
- Test metrics based on test progress reports
- Unmitigated risks, defects not fixed
- Lessons learned that are relevant to the testing
- Iteration and Release Planning

Release Planning & Iteration Planning

	發佈規畫 (release planning)	迭代規畫 (iteration planning)
範疇	• 專注於更廣泛範圍(通常跨越多個迭代或衝刺) • 涉及高層級規劃和製定產品部署的總體目標和功能	• 專注於較小範圍(通常是一次迭代或衝刺) • 涉及更詳細的規劃
時間框	涵蓋較長時間範圍(通常是幾個月到一年)，具體取決於專案的規模和複雜性	涵蓋較短的時間範圍(並在每次迭代/衝刺開始時重複)，Sprint通常持續1~4個星期
目的	定義發佈的高階目標(涉及到確定在版本結束前需要交付的關鍵特性、使用者故事和功能)	使用者故事細化(澄清)並對其拆分以便在迭代內開發
優先性	團隊根據商業價值、客戶回饋和其他策略考量來決定功能和使用者故事的優先順序	團隊根據優先順序、複雜性和可行性從產品待辦事項中選擇使用者故事或任務的子集
產品代辦清單	定義和重新定義產品待辦事項清單	與迭代待辦事項有關
團隊規模/資源配置	設定整體團隊能力，並考慮可能影響交付時間表的任何限制	根據團隊即將進行的迭代的能力分配資源

Test Quadrants

<table>
<tr><td></td><td colspan="2">User facing</td><td></td></tr>
<tr><td rowspan="2">Support / Dev.</td><td>functional test
user story test
user experience prototype
API test</td><td>exploratory testing
usability testing
acceptance test
user-oriented
often manual</td><td rowspan="2">Product / Critique</td></tr>
<tr><td>component test
component integration test
CI process
often automated</td><td>smoke test
non-functional test
often automated</td></tr>
<tr><td></td><td colspan="2">Technology facing</td><td></td></tr>
</table>