

軟體測試工程人員職能基準

職能基準代碼		ISD2519-002v1			
職能基準名稱 (擇一填寫)		職類			
		職業	軟體測試工程人員		
所屬 類別	職類別	資訊科技 / 軟體開發及程式設計		職類別代碼	ISD
	職業別	其他軟體、應用程式開發人員及分析師		職業別代碼	2519
	行業別	出版、影音製作、傳播及資通訊服務業 / 電腦程式設計、諮詢及相關服務業		行業別代碼	J6201
工作描述		從事軟體測試規劃、執行、問題回報及追蹤之工作。			
基準級別		4			

主要職責	工作任務	工作產出	行為指標	職能級別	職能內涵 (K=knowledge 知識)	職能內涵 (S=skills 技能)
T1 測試規劃、分析與設計	T1.1 測試規劃	O1.1.1 測試規劃書	P1.1.1 確認測試範圍、風險與目標。 P1.1.2 依循軟體發展生命週期 (SDLC) 測試政策策略。 P1.1.3 決定可行方案。 P1.1.4 規劃測試資源 (人力、環境、設備等)。 P1.1.5 安排測試分析、設計、實作、執行與評估作業。 P1.1.6 決定測試輸出標準 (Exit Criteria)。 P1.1.7 撰寫測試規劃書。	4	K01 職業安全衛生相關規範 K02 電腦系統層級化知識【註1】 K03 軟體開發生命週期的測試活動【註2】 K04 測試技術 (黑箱測試【註3】、白箱測試【註4】) K05 軟體測試支援工具【註5】 K06 測試管理【註6】	S01 溝通協調能力 S02 規劃和組織能力 S03 技術文件閱讀與撰寫技能 S04 問題解決能力 S05 自我管理能力的 S06 團隊協作能力 S07 外部環境認知與評估 S08 分析與解讀能力 S09 測試工具與技術使用能力 S10 數值計算技能
	T1.2 測試分析與設計	O1.2.1 測試計畫	P1.2.1 檢視分析軟體開發需求、規格、架構與介面，確認測試條件 (test conditions)。 P1.2.2 確認測試情境、標準與方法。	4	K03 軟體開發生命週期的測試活動 K04 測試技術	S03 技術文件閱讀與撰寫技能 S04 問題解決能力 S05 自我管理能力的

主要職責	工作任務	工作產出	行為指標	職能級別	職能內涵 (K=knowledge 知識)	職能內涵 (S=skills 技能)
			P1.2.3 確認測試種類、工具與輸入資料。 P1.2.4 設計測試環境與測試案例。 P1.2.5 撰寫測試計畫。		K05 軟體測試支援工具 K07 <u>測試基礎</u> ^{【註7】} K08 <u>測試設計技術</u> ^{【註8】} K09 <u>測試層級</u> ^{【註9】}	S06 團隊協作能力 S07 外部環境認知與評估 S08 分析與解讀能力 S09 測試工具與技術使用能力 S10 數值計算技能 S11 資訊安全防禦能力 S12 資料備份和還原能力
T2 測試實作與執行	T2.1 測試實作	O2.1.1 測試案例資料	P2.1.1 依據測試計畫書，發展測試案例資料（測試項目之各項輸入資料）並決定優先順序。 P2.1.2 同儕檢視測試案例與回饋、最佳化。 P2.1.3 產出測試套件以便測試有效執行。 P2.1.4 實作與確認測試環境。	4	K08 測試設計技術 K09 測試層級	S03 技術文件閱讀與撰寫技能 S04 問題解決能力 S05 自我管理能力的 S06 團隊協作能力 S07 外部環境認知與評估 S08 分析與解讀能力 S09 測試工具與技術使用能力 S10 數值計算技能 S11 資訊安全防禦能力 S12 資料備份和還原能力 S13 測試分析與設計技能

主要職責	工作任務	工作產出	行為指標	職能級別	職能內涵 (K=knowledge 知識)	職能內涵 (S=skills 技能)
	T2.2 測試執行	O2.2.1 測試事件報告	<p>P2.2.1 依測試程序執行測試套件和個別測試案例。</p> <p>P2.2.2 紀錄測試結果及測試項目的版本與識別碼 (ID)。</p> <p>P2.2.3 比較測試結果、分析異常原因與事件紀錄、並產出事件報告。</p>	4	<p>K08 測試設計技術</p> <p>K09 測試層級</p>	<p>S03 技術文件閱讀與撰寫技能</p> <p>S04 問題解決能力</p> <p>S05 自我管理能力的</p> <p>S06 團隊協作能力</p> <p>S09 測試工具與技術使用能力</p> <p>S10 數值計算技能</p> <p>S11 資訊安全防禦能力</p> <p>S12 資料備份和還原能力</p> <p>S14 測試實作與執行技能</p>
T3 測試管理	T3.1 測試評估與報告結果	O3.1.1 測試報告書	<p>P3.1.1 依測試計畫書中的輸出標準，檢查測試結果與記錄檔。</p> <p>P3.1.2 評估是否需要更多測試，或改變測試輸出標準。</p> <p>P3.1.3 撰寫測試報告。</p>	3	<p>K10 功能性測試^{【註10】}</p> <p>K11 非功能性測試^{【註11】}</p>	<p>S03 技術文件閱讀與撰寫技能</p> <p>S04 問題解決能力</p> <p>S05 自我管理能力的</p> <p>S11 資訊安全防禦能力</p> <p>S12 資料備份和還原能力</p> <p>S15 測試評估技能</p>
	T3.2 測試監控	O3.2.1 測試總結報告書	<p>P3.2.1 建立追蹤監控機制並協同修復問題。</p> <p>P3.2.2 檢查測試計畫中的交付項目，並確保測試事件均已被解決。</p> <p>P3.2.3 管理測試帳號與測試資料。</p> <p>P3.2.4 記錄所有測試腳本 (script) 及測試環境等，以利重複使用。</p> <p>P3.2.5 評估測試經驗，以利未來發行與計畫參考。</p>	3	K06 測試管理	<p>S03 技術文件閱讀與撰寫技能</p> <p>S04 問題解決能力</p> <p>S05 自我管理能力的</p> <p>S08 分析與解讀能力</p> <p>S10 數值計算技能</p> <p>S11 資訊安全防禦能力</p> <p>S12 資料備份和還原能力</p> <p>S16 測試管理技能</p>

職能內涵 (A=attitude 態度)

A01 主動積極：不需他人指示或要求能自動自發做事，面臨問題立即採取行動加以解決，且為達目標願意主動承擔額外責任。

A02 持續學習：能夠展現自我提升的企圖心，利用且積極參與各種機會，學習任務所需的新知識與技能，並能有效應用在特定任務。

A03 謹慎細心：對於任務的執行過程，能謹慎考量及處理所有細節，精確地檢視每個程序，並持續對其保持高度關注。

A04 壓力容忍：冷靜且有效地應對及處理高度緊張的情況或壓力，如緊迫的時間、不友善的人、各類突發事件及危急狀況，並能以適當的方式紓解自身壓力。

A05 自我管理：設立定義明確且實際可行的個人目標；對於及時完成任務展現高度進取、努力、承諾及負責任的行為。

A06 團隊意識：積極參與並支持團隊，能彼此鼓勵共同達成團隊目標。

A07 應對不確定性：當狀況不明或問題不夠具體的情況下，能在必要時採取行動，以有效釐清模糊不清的態勢。

說明與補充事項

● 建議擔任此職類 / 職業之學歷 / 經驗 / 或能力條件：

- 大專以上，具備任一種程式語言基礎，並有3年以上之資訊領域工作經驗。

● 其他補充說明：

- 【註1】電腦系統層級化知識：包括硬體、網路、作業系統、應用程式及資料庫等。
- 【註2】軟體開發生命週期的測試活動：包括各軟體開發生命週期的測試層級、測試類型與影響分析等。
- 【註3】黑箱測試：也稱黑盒測試 (black-box testing)，是軟體測試方法，測試應用程式的功能，而不是其內部結構或運作。測試者不需具備應用程式的程式碼、內部結構和程式語言的專門知識。測試者只需知道什麼是系統應該做的事，即當鍵入一個特定的輸入，可得到一定的輸出。測試案例是依應用系統應該做的功能，照規範、規格或要求等設計。測試者選擇有效輸入和無效輸入來驗證是否正確的輸出。此測試方法可適合大部分的軟體測試，例如整合測試 (integration testing) 以及系統測試 (system testing)。
- 【註4】白箱測試：也稱白盒測試 (white-box testing，又稱透明盒測試 glass box testing、結構測試 structural testing 等) 是一個測試軟體的方法，測試應用程式的內部結構或運作，而不是測試應用程式的功能 (即黑箱測試)。在白盒測試時，以程式語言的角度來設計測試案例。測試者輸入資料驗證資料流在程式中的流動路徑，並確定適當的輸出，類似測試電路中的節點。白箱測試可以應用於單元測試 (unit testing)、整合測試 (integration testing) 和系統的軟體測試流程，可測試在整合過程中每一單元之間的路徑，或者主系統跟子系統中的測試。儘管這種測試的方法可以發現許多的錯誤或問題，它可能無法檢測未使用部分的規範。
- 【註5】軟體測試支援工具：包括測試工具類型、工具導入注意事項、自動化的利弊等。

說明與補充事項

- 【註6】測試管理：包括測試流程細部工作、測試主管與人員的職責、測試報告與風險分析等。
- 【註7】測試基礎：包括測試的價值、測試術語、七個測試準則與測試活動等。
- 【註8】測試設計技術：包括靜態技術及動態技術。
- 【註9】測試層級：包括功能性測試及非功能性測試。
- 【註10】功能性測試：包括單元測試、整合測試、系統測試、回歸測試、接受測試、Alpha 測試^{【註12】}、Beta 測試^{【註13】}等。
- 【註11】非功能性測試：包括效能測試、負載測試、壓力測試、可使用性測試、安全測試、可攜性測試等。
- 【註12】Alpha 測試：Alpha 測試通常是階段性的開發完成後所開始進行，一直持續到進入 Beta 測試階段前的階段。Alpha 測試是一種驗證測試，在類比的環境中以類比的資料來執行。在這個階段中，通常是在開發單位由開發人員與測試的測試人員，以類比或實際操作性的方式進行驗證測試。
- 【註13】Beta 測試：在系統測試中通常先進行 Alpha 測試以驗證資訊系統符合使用者以及設計需求所期望的功能。當 Alpha 階段完成後，開發過程進入到 Beta 階段，由公眾參與的測試的階段。Beta 測試可稱為確認測試，在一個真實的環境中以實際的資料來執行測試，以確認效能，系統執行有效率，系統復原與備份作業正常，透過測試讓資訊系統日後可以更趨完善。