臺北市政府資訊局
Department of Information Technology, Taipei City Government

台北市政府資訊局
AI 領域專業課程

國立臺北大學 National Taipei University

AACSB ACCREDITED

IM NTPU Information Management National Taipei University

生成式 AI 與 AI Agent 實務應用、治理及採購驗收

Generative AI & AI Agent: Application, Governance, Procurement & Acceptance

https://learn.ntpu.ai/courses/goai1016h

時間：2026/11/02 (一) (11/16, 11/23, 12/14) 9:00-16:30
地點：臺北市信義區市府路 1 號 10 樓 (臺北市政府資訊局)

NVIDIA University Ambassador Certified Instructor

aws educate Cloud Ambassador 2020 Cohort

aws Accredited Educator

aws certified Cloud Practitioner

aws certified Solutions Architect Associate



戴敏育 教授 (Prof. Min-Yuh Day)

國立臺北大學 資訊管理研究所 所長

跨域 AI 前瞻研究中心 (AI4X NTPU) 主任



戴敏育 教授

Prof. Min-Yuh Day

NVIDIA University Ambassador
Certified Instructor

aws educate | Cloud Ambassador
2020 Cohort

aws academy Accredited Educator

aws certified Solutions Architect Associate

aws certified Cloud Practitioner

Director, Information Management, NTPU

Director, Intelligent Financial Innovation Technology, IFIT Lab, IM, NTPU

Director, Advanced Research Center for Cross-Disciplinary AI (AI4X), NTPU

CEO, Fintech and Green Finance Center (FGFC), NTPU

Visiting Scholar, IIS, Academia Sinica

Ph.D., Information Management, NTU

Artificial Intelligence, Agentic AI, ESG and Green Financial Technology, Big Data Analytics, Electronic Commerce

NTPU 國立臺北大學 National Taipei University

中央研究院 ACADEMIA SINICA 1928

NTPU National Taipei University AI4X Research Center

NTPU 國立臺北大學 National Taipei University

AACSB ACCREDITED

IM NTPU Information Management National Taipei University

NTPU AI Learn
https://learn.ntpu.ai/

課程大綱

1 **政府 AI 應用判斷與責任邊界**
工作適合讓 AI 做到哪一步？

2 **政府 AI 工作流程與人工關卡**
AI 如何進入流程，何時必須交還人工？

3 **可信資料、工具連接與權限**
AI 可以知道什麼、查什麼、做什麼？

4 **政府 AI 治理、資安與人工監督**
誰負責、誰核准，如何控制風險？

5 **AI 可靠性、測試與失敗處理**
如何用證據確認系統可靠？

6 **政府 AI 採購需求與驗收決策**
需求如何寫得可驗證，系統如何判定可接受？

https://learn.ntpu.ai/courses/goai1016h

政府 AI 應用判斷與責任邊界

這項工作適合讓 AI 做到哪一步？

1. 分辨 Generative AI、Agentic AI 與 AI Agent 的差異，並說明它們在政府工作中的意義。
2. 從業務目的、錯誤後果與適合度，判斷一項政府工作能否用 AI。
3. 把 AI 的角色分類為 Assist、Recommend 或 Execute，並選出適當層級。
4. 寫出 AI 允許行動、禁止行動與負責的人，完成「AI 應用判斷卡」。

三個名詞：Generative AI、Agentic AI、AI Agent

政府機關導入 AI 之前，第一個該問的不是
「這個工具好不好用」，
而是「這項工作到底適合讓 AI 做到哪一步」。
要回答這個問題，
先用三個名詞建立共同語言。
它們描述的是一條能力光譜，
不是三個互斥的產品類別。

三個名詞：Generative AI、Agentic AI、AI Agent (cont.)

名詞	它能做什麼	政府情境範例
Generative AI	根據輸入產生文字、摘要或草稿，本身不會自己採取行動	把一份冗長的函釋摘要成一頁重點，由承辦人確認
Agentic AI	能把一個目標拆成多個步驟、規劃順序，並依結果調整下一步	收到陳情後，自行規劃「分類、查規定、擬回覆」的處理步驟
AI Agent	實際呼叫工具、查詢或寫入系統的執行體	連到案件系統查詢進度，甚至直接更新案件狀態

三個名詞的差異與政府情境

三個名詞：Generative AI、Agentic AI、AI Agent (cont.)

1. 能力愈往右，AI 離「真的動手」愈近，出錯時的影響也愈大。

2. AI 的輸出具有不確定性，不能等同於傳統資訊系統依固定邏輯產生的確定結果，所以需要人工驗證與治理。

3. 判斷時看的不是產品名稱，而是它在這個情境中「實際被允許做到哪一步」。

三個名詞：Generative AI、Agentic AI、AI Agent (cont.)



同一套系統可能同時具備產生內容、規劃步驟與呼叫工具的能力。能力強不代表就應該全部開放，開放到哪裡是機關要做的判斷。

三個名詞：Generative AI、Agentic AI、AI Agent (cont.)



Key takeaway: 三個名詞是能力光譜：愈接近「真的動手」，愈需要明確的權限邊界與人工把關。

三個名詞：Generative AI、Agentic AI、AI Agent (cont.)

Knowledge Check

某系統可以連到案件系統，查詢陳情案件進度，並直接把案件狀態改成「已結案」。這最接近哪一個名詞？風險重點是什麼？

A. Generative AI：它只是產生文字，風險很低

B. AI Agent：它能寫入系統，錯誤會直接改變案件紀錄，需要明確的權限與人工把關

C. 傳統資訊系統：只要能連系統，輸出就一定正確

Answer

Correct answer: B

Answer

某系統可以連到案件系統，查詢陳情案件進度，並直接把案件狀態改成「已結案」。這最接近哪一個名詞？風險重點是什麼？

A. Generative AI：它只是產生文字，風險很低

✓ **AI Agent：它能寫入系統，錯誤會直接改變案件紀錄，需要明確的權限與人工把關**

C. 傳統資訊系統：只要能連系統，輸出就一定正確

Why: B：能呼叫工具並寫入系統的是 AI Agent，寫入會直接改變正式案件紀錄，所以要限定權限並設計人工把關。A 錯在把「能寫入」誤當成「只產生文字」；C 錯在假設 AI 輸出和固定邏輯系統一樣確定，忽略輸出的不確定性。

業務目的、錯誤後果與適合度

判斷一項政府工作適不適合用 AI，從三個問題開始：這項工作的業務目的是什麼、有沒有可檢查的品質標準？AI 出錯時後果多嚴重、能不能及早發現？是不是已經有人負責最終審查？三個問題答不出來，就先不要往下設計。

業務目的、錯誤後果與適合度 (cont.)

工作	業務目的	錯誤後果	適合度判斷
會議紀錄整理成待辦清單	節省整理時間	輕微：漏項在人工確認時可發現	適合：Assist 即可
陳情案件分類與回覆草稿	縮短分案與初擬時間	中等：分錯類會延誤、回覆不當會損害信任	適合，但每件都須經承辦人審查
補助申請資格核定	依法認定是否符合資格	嚴重：影響人民權益，涉及法定職權與正當程序	AI 僅能整理資料，核定須由具法定職權人員負責

依錯誤後果判斷 AI 可以走多遠（模擬情境）

業務目的、錯誤後果與適合度 (cont.)

本課程貫穿案例（模擬）

某市政府民政局（模擬單位）每月收到大量民眾陳情，承辦人要先分類、查詢相關規定，再擬出回覆。局內考慮導入 AI 協助。後續模組會沿用這個案例，做出每一張卡片與表格。

業務目的、錯誤後果與適合度 (cont.)



AI 不能取代法定職權。
涉及法定資格、裁量權或人民權益的決定，必須由具職權的人員審查並簽核，不能因為「效率」而被跳過。

業務目的、錯誤後果與適合度 (cont.)



Key takeaway:
先看業務目的、錯誤後果與是否有人審查，再決定能不能用 AI；
錯誤後果愈嚴重，AI 能走的步數愈少。

業務目的、錯誤後果與適合度 (cont.)

Knowledge Check

下列哪一項最符合「值得進一步設計成 AI 應用」的條件？

A. 工作沒有明確的品質標準，錯誤也不容易被發現，但長官覺得很新潮

B. 工作有可檢查的品質標準、錯誤的影響範圍可控，而且已有人負責最終審查

C. 工作涉及法定裁量權，目前沒有人負責最終審查，但可以大幅省時

Answer

Correct answer: B

Answer

下列哪一項最符合「值得進一步設計成 AI 應用」的條件？

A. 工作沒有明確的品質標準，錯誤也不容易被發現，但長官覺得很新潮

✓ **工作有可檢查的品質標準、錯誤的影響範圍可控，而且已有人負責最終審查**

C. 工作涉及法定裁量權，目前沒有人負責最終審查，但可以大幅省時

Why: B：有可檢查的品質標準、影響可控、已有人負責最終審查，是值得往下設計的三個基本條件。A 無法驗證品質，「長官覺得新潮」不是業務目的；C 涉及法定裁量卻沒有審查機制，省時不能成為跳過責任的理由。

三種角色：Assist、Recommend、Execute

確定一項工作可以用 AI 之後，下一步是決定 AI 在這項工作中扮演什麼角色。

角色不是工具本身的屬性，

而是機關替這個應用所做的授權決定。

三種角色：Assist、Recommend、Execute (cont.)

角色	AI 做什麼	人的責任	陳情案例
Assist（輔助）	提供草稿、摘要或選項，不代表結論	決定是否採用、如何修改	把陳情內容摘要成三行重點
Recommend（建議）	依資料提出分類或排序建議，並說明依據	審查依據，做最終決定	建議分類與處理優先順序，由承辦人確認
Execute（執行）	在事先明確授權的邊界內實際執行動作	事先設定邊界，事後可追溯、可撤回	自動把案件轉給指定課室並寄出收件通知

AI 角色分級與人的責任

三種角色：Assist、Recommend、Execute (cont.)

1. 常見的政府 AI 應用多數落在 Assist 或 Recommend，只有風險低、邊界清楚、經過嚴格評估的步驟才考慮 Execute。

2. 同一個應用可以不同步驟用不同角色：摘要用 Assist、分類用 Recommend、寄出收件通知才用 Execute。

3. 角色不是一次定案：可先以 Assist 上線觀察，累積證據後再評估是否提升。

三種角色：Assist、Recommend、Execute (cont.)



Key takeaway: 角色是機關的授權決定：多數應用從 Assist 或 Recommend 起步，提升到 Execute 需要個案評估。

三種角色：Assist、Recommend、Execute (cont.)

Knowledge Check

系統依陳情內容產生分類建議並排出處理優先順序，說明依據，但正式分類仍由承辦人確認。這屬於哪一種角色？

A. Assist，因為它只是提供草稿

B. Recommend，因為它依資料提出建議並說明依據，最終決定仍由人做

C. Execute，因為分類結果已經產生出來了

Answer

Correct answer: B

Answer

系統依陳情內容產生分類建議並排出處理優先順序，說明依據，但正式分類仍由承辦人確認。這屬於哪一種角色？

A. Assist，因為它只是提供草稿

✓ **Recommend，因為它依資料提出建議並說明依據，最終決定仍由人做**

C. Execute，因為分類結果已經產生出來了

Why: B：提出有依據的建議、最終決定仍在人，是 Recommend 的典型樣貌。A 錯在 Assist 是尚未形成結論的草稿或摘要，這裡已有分類與排序及依據；C 錯在 Execute 是 AI 在授權邊界內直接完成動作，不等待人工確認，而本例正式分類仍由承辦人確認。

允許行動、禁止行動與負責的人

選定角色之後，要把它寫成具體的行動邊界。「AI 只是輔助」這種說法不夠，必須明確列出 AI 可以做什麼、不可以做什麼，以及出錯時由誰負責。

允許行動、禁止行動與負責的人 (cont.)

項目	內容
AI 角色	Recommend：建議分類並擬回覆草稿
允許行動	摘要陳情內容、建議分類、檢索公開規定、擬回覆草稿
禁止行動	未經承辦人審查就寄出回覆、更改案件結案狀態、認定是否符合法定資格、處理未經授權的個資
主要風險	分類錯誤造成延誤；引用過時規定；回覆語氣或內容不當；個資外洩
負責的人	民政局陳情承辦人（審查並簽發每一份回覆），課長負責整體應用的授權範圍

陳情案件 AI 應用：行動邊界示例（模擬）

允許行動、禁止行動與負責的人 (cont.)



Human–AI–Human：人定義任務與判斷標準，AI 協助產生草稿或建議，人審查結果並承擔最終責任。「有人看過」還不夠，要寫明是誰、審查什麼、有權否決。

允許行動、禁止行動與負責的人 (cont.)

1. 禁止行動通常不是技術做不到，而是業務與治理上不允許。
2. 負責的人要具體到職稱或角色，不能寫「相關人員」或「系統管理者」。
3. 這四個欄位（角色、允許、禁止、負責人）是後面所有模組往下設計的起點。

允許行動、禁止行動與負責的人 (cont.)



Key takeaway: 把角色寫成允許行動、禁止行動與具體的負責人，責任邊界才算真正建立。

允許行動、禁止行動與負責的人 (cont.)

Knowledge Check

在行動邊界表中，「禁止行動」欄位最主要的用途是什麼？

A. 記錄 AI 技術上做不到的事，與業務判斷無關

B. 明確寫出即使技術上可行也不允許 AI 做的行動，讓責任邊界在設計階段就清楚

C. 這是選填欄位，大部分情況可以留空

Answer

Correct answer: B

Answer

在行動邊界表中，「禁止行動」欄位最主要的用途是什麼？

A. 記錄 AI 技術上做不到的事，與業務判斷無關

✓ **明確寫出即使技術上可行也不允許 AI 做的行動，讓責任邊界在設計階段就清楚**

C. 這是選填欄位，大部分情況可以留空

Why: B：禁止行動是業務與治理上的邊界，即使技術可行也不允許 AI 做，寫清楚才能讓責任邊界在設計階段就明確，並成為後續權限與驗收的依據。A 錯在把它當技術限制；C 錯在它是必填，也是最能避免責任模糊的欄位。

實作：完成「AI 應用判斷卡」

請針對課程提供的模擬陳情案例，完成第一張卡片：AI 應用判斷卡。案例背景已經提供，你不需要自己選題，把時間用在判斷與推理上。卡片內容會填入整合版「政府 AI 治理與驗收 Canvas」的前四個欄位，後面的模組會一路沿用。

實作：完成「AI 應用判斷卡」(cont.)

1 

讀案例背景

確認業務目的、目前處理方式與使用者是誰。

2 

選擇 AI 角色

從 Assist、Recommend、Execute 中選一個，並寫下理由，說明為什麼不選更高一級。

3 

寫允許行動與禁止行動

各至少列出兩項，禁止行動要具體，不能只寫「不可違法」。

4 

寫主要風險與負責的人

列出最重要的風險與出錯後果，並指定具體負責人與其審查責任。

實作：完成「AI 應用判斷卡」(cont.)

1. 對應 Canvas 欄位：use case 名稱（useCaseName）、業務目標（businessGoal）、AI 角色（aiRole）、負責人（humanOwner）。

2. 請勿輸入真實身分證字號、民眾個資、機關內部機敏資訊或任何帳號密碼，一律使用模擬或去識別化資料。

實作：完成「AI 應用判斷卡」(cont.)



Key takeaway: 判斷卡是整合版 Canvas 的起點：角色、允許與禁止行動、負責的人愈早寫清楚，後面的流程、權限與驗收愈不容易走偏。

實作：完成「AI 應用判斷卡」(cont.)

Knowledge Check

寫 AI 應用判斷卡的「負責的人」時，下列哪一種寫法最恰當？

A. 相關人員

B. 民政局陳情承辦人，負責審查並簽發每一份回覆；課長負責應用授權範圍

C. AI 系統本身，因為它會自動檢查

Answer

Correct answer: B

Answer

寫 AI 應用判斷卡的「負責的人」時，下列哪一種寫法最恰當？

A. 相關人員

✓ **民政局陳情承辦人，負責審查並簽發每一份回覆；課長負責應用授權範圍**

C. AI 系統本身，因為它會自動檢查

Why: B：負責的人必須具體到角色與責任內容，而且必須是人。A「相關人員」出事時無人可找；C 把責任交給 AI 本身，AI 不能承擔責任，人才是 Human–AI–Human 中最後負責的一端。

政府 AI 工作流程與人工關卡

AI 如何進入流程，何時必須交還人工？

1. 區分 Task 與 Workflow，把業務拆成有明確輸入與輸出的步驟。
2. 標出每一步由 AI、人工或規則決定，並辨識 Sequential、Parallel、Routing。
3. 在流程中補上人工核准、升級處理、交還人工、完成條件與失敗路徑。

從 Task 到 Workflow：先拆步驟，再談 AI

模組 1 回答「這項工作 AI 可以做到哪一步」。本模組往下一層：這項工作實際上由哪些步驟組成，AI 在哪一步進場，哪一步必須由人接手。政府業務很少是「問一次、答一次」，而是一段多步驟的流程（Workflow），例如收件、分類、查法規、擬稿、審查、回覆。

從 Task 到 Workflow：先拆步驟，再談 AI (cont.)

Task（單一任務）

一次互動內可完成，例如「摘要這份文件」。

有一個輸入、一個輸出。

錯誤影響範圍小，容易檢查。

Workflow（工作流程）

由多個 Task 依順序或條件串接而成。

上一步的輸出就是下一步的輸入。

錯誤會往後傳遞，必須知道每一步由誰負責、怎麼檢查。

從 Task 到 Workflow：先拆步驟，再談 AI (cont.)

1. 輸入（Input）：這一步開始前必須拿到什麼，例如陳情全文、承辦人指派。
2. 輸出（Output）：這一步結束後必須交出什麼，例如分類結果、回覆草稿。
3. 完成條件：怎樣才算這一步做完，例如「分類結果已由承辦人確認」。
4. 拆解的判斷標準：這一步能不能被獨立檢查？不能，通常代表還要再拆。

從 Task 到 Workflow：先拆步驟，再談 AI (cont.)



先把步驟、輸入、輸出寫清楚，再決定哪一步交給 AI。流程不清楚時直接導入 AI，只會把混亂自動化。

從 Task 到 Workflow：先拆步驟，再談 AI (cont.)



Key takeaway: 先把工作拆成有輸入、輸出與完成條件、可獨立檢查的步驟，再決定 AI 在哪一步進場。

從 Task 到 Workflow：先拆步驟，再談 AI (cont.)

Knowledge Check

下列哪一項最能說明「一個步驟已經拆得夠清楚」？

A. 這個步驟有名稱，而且流程圖看起來很完整

B. 這個步驟有明確的輸入、輸出與完成條件，而且可以被獨立檢查

C. 這個步驟會用到 AI

Answer

Correct answer: B

Answer

下列哪一項最能說明「一個步驟已經拆得夠清楚」？

A. 這個步驟有名稱，而且流程圖看起來很完整

✓ **這個步驟有明確的輸入、輸出與完成條件，而且可以被獨立檢查**

C. 這個步驟會用到 AI

Why: B：步驟有明確的輸入、輸出與完成條件，並且可以被獨立檢查，才算拆得夠清楚。A 錯在以圖面「看起來完整」當標準，名稱與美觀不等於可檢查；C 錯在把「用不用 AI」當成拆解標準，兩者無關，拆解是為了之後決定 AI 該放在哪裡。

流程模式與決策分工：誰決定下一步

模式	意思	政府情境範例
Sequential（依序）	後一步依賴前一步的結果	分類、摘要、擬稿，一步接一步
Parallel（平行）	互不依賴的步驟同時進行，再彙整	同時查相關法規與可能的權責機關
Routing（分流）	依條件把案件導向不同路徑	依陳情類型分到不同業務科室

三種基本流程模式

流程模式與決策分工：誰決定下一步 (cont.)

流程中每個需要判斷「接下來走哪一條路」的地方叫決策點（Decision Point）。決策點有三種決定者，必須在流程圖上標清楚：

流程模式與決策分工：誰決定下一步 (cont.)

決定者	適合的決策	要注意
規則（固定條件）	條件明確、可事先寫死，例如「案件編號重複就合併」	最可預測、最容易稽核
AI（建議或判斷）	需要理解文字、歸納或初步分類	AI 只提出建議，影響權益時須由人確認
人工（承辦人或主管）	涉及法定職權、裁量、對外正式效力	必須有具名角色，不能寫成「相關人員」

決策點與決定者

流程模式與決策分工：誰決定下一步 (cont.)



分流（Routing）若由 AI 判斷，分錯就等於送錯單位。這類決策點要設計「信心不足或類型不明時交人工」的出口。

流程模式與決策分工：誰決定下一步 (cont.)



Key takeaway: 步驟互相依賴用 Sequential，互不依賴用 Parallel，依條件分岔用 Routing；每個決策點都要標明由規則、AI 或人工決定。

流程模式與決策分工：誰決定下一步 (cont.)

Knowledge Check

某陳情案件需要同時查「相關法規」與「可能的權責機關」，兩者互不依賴。這兩步最適合用哪種模式？

A. Sequential：先查法規，查完才能查權責機關

B. Parallel：兩個查詢同時進行，之後再彙整給承辦人

C. Routing：由 AI 決定要不要查其中一項

Answer

Correct answer: B

Answer

某陳情案件需要同時查「相關法規」與「可能的權責機關」，兩者互不依賴。這兩步最適合用哪種模式？

A. Sequential：先查法規，查完才能查權責機關

✓ **Parallel：兩個查詢同時進行，之後再彙整給承辦人**

C. Routing：由 AI 決定要不要查其中一項

Why: B：兩個查詢互不依賴，適合 Parallel，同時進行後彙整給承辦人檢視。A（Sequential）錯在硬把互不依賴的步驟排成先後，無謂拉長流程；C（Routing）錯在混淆概念，Routing 處理「依條件走哪條路」，不是「同時做兩件事」。

案例：一張未完成的陳情處理流程

本課程使用一個模擬教學情境：某市政府（模擬）的民眾陳情案件處理，導入 AI 協助分類、摘要與回覆草稿。下面是業務單位畫到一半的流程，請先找出問題，不要急著重畫。

案例：一張未完成的陳情處理流程 (cont.)



未完成的陳情處理流程（模擬）

案例：一張未完成的陳情處理流程 (cont.)

1. 缺口一：AI 分類沒有人工確認，分錯類會一路送錯單位。
2. 缺口二：回覆草稿直接寄出，沒有人工核准關卡；正式對外回覆屬於高影響行動。
3. 缺口三：沒有例外路徑，遇到威脅字句、內容不明或查不到法規時不知道該怎麼辦。
4. 缺口四：沒有完成條件與失敗路徑，AI 出錯或查詢失敗時流程會卡住或默默繼續。

案例：一張未完成的陳情處理流程 (cont.)



檢查流程是否完整的三個問題：每一步由誰負責？哪裡必須由人核准？走不下去時下一步是什麼？

案例：一張未完成的陳情處理流程 (cont.)



Key takeaway: 檢查流程時先看高影響行動前有沒有人工核准，再看例外路徑與失敗路徑是否存在。

案例：一張未完成的陳情處理流程 (cont.)

Knowledge Check

在上述陳情流程中，哪一個缺口的風險最高、最應該先補？

A. 「AI 摘要」步驟的名稱不夠精確

B. 回覆草稿在沒有人工核准的情況下直接寄出

C. 流程圖沒有標示每一步花費的時間

Answer

Correct answer: B

Answer

在上述陳情流程中，哪一個缺口的風險最高、最應該先補？

A. 「AI 摘要」步驟的名稱不夠精確

✓ **回覆草稿在沒有人工核准的情況下直接寄出**

C. 流程圖沒有標示每一步花費的時間

Why: B：正式對外回覆屬高影響且不易撤回，必須先補人工核准關卡。A、C 只是名稱與時間標示的完整性問題，不會造成對外錯誤，優先順序較低；學員容易被「補得愈齊全愈好」吸引，要強調風險高低決定先後。

人工核准、升級與交還人工

機制	何時使用	陳情案例
人工核准（Approval）	高影響行動生效前，必須有具名角色明確同意	承辦人審核並核准回覆草稿後才能寄出
升級處理（Escalation）	案件超出目前角色或 AI 的範圍，送給更高權限者判斷	權責機關不明，交資深承辦人判斷
交還人工（Hand-back）	整件案件不再由 AI 處理，流程改由人工全程接手	陳情內容出現人身安全威脅字句
失敗路徑（Failure Path）	某一步無法完成時的預先安排	查詢法規失敗：標記為未完成並通知承辦人，不得用 AI 猜測的內容補上

四種人工介入機制

人工核准、升級與交還人工 (cont.)

完成條件怎麼寫

「承辦人已核准回覆草稿，案件狀態更新為已回覆，並留下核准紀錄」才算完成；「AI 已產生草稿」只是中間產出，不是完成。

人工核准、升級與交還人工 (cont.)

1. 人工關卡必須指明：誰審、看什麼、可以做什麼（核准、修改、退回、交還人工）。
2. 「請人看一下」不是關卡；沒有具名角色、沒有檢查內容、沒有紀錄，就不算數。
3. 失敗時「默默繼續」是最危險的設計；流程要明確寫出停止、通知與人工接手。

人工核准、升級與交還人工 (cont.)



Human–AI–Human：人定義任務，AI 協助或執行，人審查並負責。AI 不能取代法定職權，也不能自行核准自己的產出。

人工核准、升級與交還人工 (cont.)



Key takeaway: 人工關卡要有具名角色、檢查內容與紀錄；例外與失敗要有預先寫好的下一步，而不是臨時補救。

人工核准、升級與交還人工 (cont.)

Knowledge Check

陳情內容出現對承辦人員的人身安全威脅字句，流程最合適的處理方式是什麼？

A. 讓 AI 照原流程繼續分類與擬稿，最後再請人工看一次

B. 觸發交還人工，整件案件改由人工全程處理並依機關程序通報

C. 忽略該字句，只處理陳情的主要訴求

Answer

Correct answer: B

Answer

陳情內容出現對承辦人員的人身安全威脅字句，流程最合適的處理方式是什麼？

A. 讓 AI 照原流程繼續分類與擬稿，最後再請人工看一次

✓ **觸發交還人工，整件案件改由人工全程處理並依機關程序通報**

C. 忽略該字句，只處理陳情的主要訴求

Why: B：威脅人身安全的內容超出 AI 流程的設計邊界，應觸發交還人工，整件改由人工依機關程序處理與通報。A 錯在讓 AI 照常處理、人只在最後看一次，仍把警示內容留在自動化流程中；C 錯在忽略警示內容，既不安全也不符合機關責任。

實作：補齊含人工關卡的流程圖

承接上一張投影片的未完成陳情流程。你不需要從空白開始畫，也不需要建置任何自動化平台；任務是檢查並補上人工關卡與例外路徑，讓流程完整且責任清楚。成果會填入整合版 Canvas 的「流程步驟」欄位。

實作：補齊含人工關卡的流程圖 (cont.)

1. **標出分工**
   在每一步標示由 AI、人工或規則決定，並寫出輸入與輸出。

2. **補上人工核准**
   在高影響行動之前加入人工核准關卡，寫明誰審、看什麼、可做什麼。

3. **補上例外路徑**
   至少寫出一條升級處理與一條交還人工的條件。

4. **寫完成條件與失敗路徑**
   說明怎樣才算完成，以及某一步失敗時下一步是什麼。

實作：補齊含人工關卡的流程圖 (cont.)

1. 成果：「含人工關卡的流程圖」，填入整合版 Canvas 的「流程步驟」。

2. 評量重點：流程是否完整、責任是否清楚，而不是圖畫得多複雜。

3. 請只使用模擬或已去識別化的資料；不要輸入機密、真實個資或未經授權的敏感資訊。

實作：補齊含人工關卡的流程圖 (cont.)



Key takeaway: 流程完不完整，看的是高影響行動前有沒有人工核准、每個例外與失敗情境有沒有明確的下一步。

實作：補齊含人工關卡的流程圖 (cont.)

Knowledge Check

流程圖中的「失敗路徑」主要用來記錄什麼？

A. 這個流程過去失敗過幾次

B. 某一步無法完成時，流程接下來怎麼走，例如通知承辦人、升級處理或交還人工

C. 備註用欄位，不影響實際流程設計

Answer

Correct answer: B

Answer

流程圖中的「失敗路徑」主要用來記錄什麼？

A. 這個流程過去失敗過幾次

✓ **某一步無法完成時，流程接下來怎麼走，例如通知承辦人、升級處理或交還人工**

C. 備註用欄位，不影響實際流程設計

Why: B：失敗路徑回答「某一步無法完成時，流程接下來怎麼走」，例如通知承辦人、升級處理或交還人工。A 把它誤當成歷史失敗統計；C 誤以為只是備註欄位而不影響流程設計，兩者都忽略失敗路徑是流程設計的必要部分。

可信資料、工具連接與權限

AI 可以知道什麼、查什麼、做什麼？

1. 以 Include／Retrieve／Remember／Exclude 判斷 AI 可使用的資料，並依資料分類辨識禁止輸入項目。
2. 說明來源、版本與時效如何決定資料是否可信，來源衝突時由人工判斷。
3. 區分 RAG、API、MCP 的用途，並把每個工具行動分為查詢、寫入或發送。
4. 依最小權限原則，在「資料與工具權限表」記錄存取範圍、核准者與限制，並設計紀錄與失敗備援。

Context：AI 回答前，該先知道什麼

Prompt 是「這次問什麼」，Context 是「AI 回答時可以參考的背景資訊」。AI 並不知道機關的內部作業規定或案件歷史；問題寫得再清楚，如果 Context 沒設計好，答案仍可能缺乏依據。本模組沿用同一個模擬案例：某市政府「陳情案件回覆輔助」，AI 協助承辦人起草回覆，由承辦人審查後才送出。

Context：AI 回答前，該先知道什麼 (cont.)

處理方式	適用情況	陳情回覆案例
Include（直接放入）	少量、每次都需要、不常變動	本次陳情內容（已去識別化）
Retrieve（需要時查取）	大量、不是每次都用、會更新	現行有效的法規條文與公開處理原則
Remember（跨對話記住）	需延續多次互動、須明訂保存期限	案件編號與處理進度
Exclude（明確不給）	與任務無關，或不得提供給 AI	其他民眾的個人資料、未公開的內部決策、機密公文

Context 的四種處理方式（以陳情回覆案例說明）

Context：AI 回答前，該先知道什麼 (cont.)



Exclude 清單往往比 Include 清單更重要：漏掉一項該排除的資料，風險通常遠高於少給一項該包含的資訊。

Context：AI 回答前，該先知道什麼 (cont.)



Key takeaway: Prompt 決定這次問什麼，Context 決定 AI 能不能答得有依據；先分好 Include／Retrieve／Remember／Exclude，再談其他設計。

Context：AI 回答前，該先知道什麼 (cont.)

Knowledge Check

在陳情回覆案例中，「其他民眾的個人資料」最適合歸入哪一類？

A. Include，因為多給一些背景有助於 AI 判斷

B. Exclude，因為與本次陳情無關，且涉及第三人個資，不應提供給 AI

C. Remember，因為之後可能會用到

Answer

Correct answer: B

Answer

在陳情回覆案例中，「其他民眾的個人資料」最適合歸入哪一類？

A. Include，因為多給一些背景有助於 AI 判斷

✓ **Exclude，因為與本次陳情無關，且涉及第三人個資，不應提供給 AI**

C. Remember，因為之後可能會用到

Why: B：其他民眾的個人資料與本次陳情無關，又涉及第三人個資保護，應明確列入 Exclude。A 錯在把「多給背景有助判斷」當成納入理由，C 錯在「之後可能用到」不是保存第三人資料的正當理由，兩者都是學員最容易選的誘答。

可信來源與資料分類：什麼能給、什麼不能給

一份資料能不能交給 AI，要回答兩個問題：這份資料可不可信（來源、版本、時效），以及這份資料能不能給（資料分類與機關規定）。兩個問題都通過，才進入 Include 或 Retrieve。

可信來源與資料分類：什麼能給、什麼不能給 (cont.)

分類	說明	可否交給 AI
公開資料	已對外公開的公告、公開統計	可依機關政策使用
機關內部資料	未對外公開的行政作業資料	依資料分類與授權範圍判斷
個人資料	可識別特定個人的資料	原則上避免；必要時先去識別化並依規定辦理
敏感資料	健康、犯罪前科等特殊類別個人資料	原則禁止，需有明確法源與授權
機密資料	依機密等級列管的公務資料	禁止提供給未經授權的系統或服務
禁止輸入第三方 AI 服務之資料	依機關政策或契約明訂不得外流的資料	不得輸入，無例外

政府資料分類（教學用簡化分級）

可信來源與資料分類：什麼能給、什麼不能給 (cont.)

1. 來源：優先採機關正式發布、來源明確的資料；AI 的回答應指出引用了哪個來源，方便人工核對。
2. 版本：法規與作業規定會修訂，知識庫必須反映現行有效版本，不能留著已廢止的舊版。
3. 時效：即時狀態（如案件進度）應查當下資料，不應依過時內容回答。
4. 來源衝突：新舊版本或不同來源互相矛盾時，由人工判斷採用哪一個，不能讓 AI 自行挑選。

可信來源與資料分類：什麼能給、什麼不能給 (cont.)



能否把某項資料輸入特定 AI 服務，取決於機關政策、契約、資安控管、資料分類與適用法令；各機關規定不同，本課程教的是判斷方法，不是一體適用的授權。

可信來源與資料分類：什麼能給、什麼不能給 (cont.)



Key takeaway: 資料要同時通過「可信」與「可給」兩關：來源、版本、時效可查證，且資料分類允許；衝突與例外一律由人工判斷。

可信來源與資料分類：什麼能給、什麼不能給 (cont.)

Knowledge Check

知識庫中同時有某項作業規定的新舊兩個版本，AI 取回時無法判斷哪一個現行有效。最合適的處理是什麼？

A. 讓 AI 自行挑選看起來較完整的版本

B. 由人工判斷並標明現行有效版本，同時清理或標記舊版，避免日後再被誤用

C. 把兩個版本都附在回覆中，讓民眾自己判斷

Answer

Correct answer: B

Answer

知識庫中同時有某項作業規定的新舊兩個版本，AI 取回時無法判斷哪一個現行有效。最合適的處理是什麼？

A. 讓 AI 自行挑選看起來較完整的版本

✓ **由人工判斷並標明現行有效版本，同時清理或標記舊版，避免日後再被誤用**

C. 把兩個版本都附在回覆中，讓民眾自己判斷

Why: B：版本衝突屬資料治理問題，應由人工判斷並標明現行有效版本，並清理或標記舊版。A 錯在讓 AI 自行挑選，C 錯在把衝突丟給民眾，兩者都是推卸責任；學員選 A 通常是因為覺得 AI 能判斷「哪個比較完整」，請指出完整不等於現行有效。

工具連接：RAG、API、MCP 與三種行動

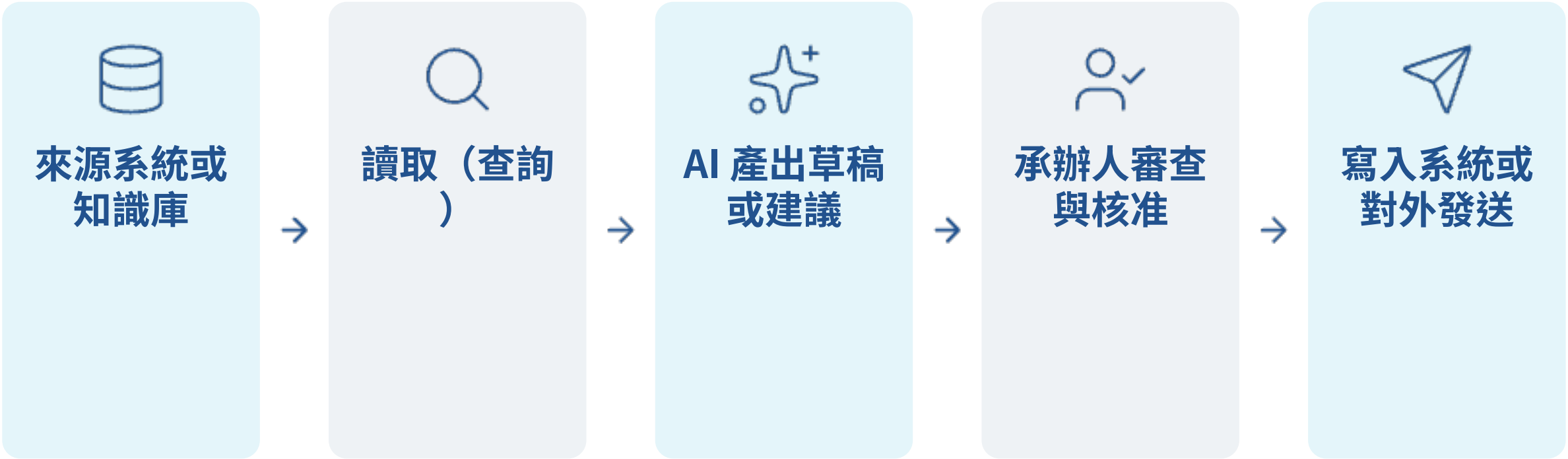
前半談的是 AI 可以「知道」什麼；後半談 AI 可以「查什麼、做什麼」。AI 一旦連上工具，就不只是產生文字，還可能讀取、改動系統資料，甚至對外發送。用途不同，控制的重點也不同。

工具連接：RAG、API、MCP 與三種行動 (cont.)

技術	主要用途	陳情回覆案例
RAG	取回知識內容，協助 AI 有依據地回答	從法規與處理原則知識庫取回相關條文
API	系統之間既有、特定用途的介接	公文列管系統提供的進度查詢介面
MCP	標準化的工具發現與呼叫方式，可串接多種工具	AI 以一致方式連接多個不同系統，不必逐一客製

RAG、API、MCP 的用途差異

工具連接：RAG、API、MCP 與三種行動 (cont.)



資料流向：誰把資料交給誰

工具連接：RAG、API、MCP 與三種行動 (cont.)

1. 查詢（Read）：只讀取資料，風險較低，可以先開放。

2. 寫入（Write）：新增、修改或刪除系統資料，需額外審查，且要先問「做錯能否復原」。

3. 發送（Send）：對外送出公文、通知或回覆，影響不可收回，必須有人工核准。

工具連接：RAG、API、MCP 與三種行動 (cont.)



Key takeaway: 先分清楚用途（知識、查詢、行動）與動作類型（查詢、寫入、發送），再討論要給多少權限。

工具連接：RAG、API、MCP 與三種行動 (cont.)

Knowledge Check

下列哪一項最能說明 RAG 與 API／MCP 的角色差異？

A. RAG 負責取回知識內容協助回答；API／MCP 負責查詢即時資料或執行動作，兩者可並用

B. RAG 與 API／MCP 是同一種技術，只是名稱不同

C. MCP 比較新，所以可以取代 RAG 與 API

Answer

Correct answer: A

Answer

下列哪一項最能說明 RAG 與 API／MCP 的角色差異？

✓ RAG 負責取回知識內容協助回答；API／MCP 負責查詢即時資料或執行動作，兩者可並用

B. RAG 與 API／MCP 是同一種技術，只是名稱不同

C. MCP 比較新，所以可以取代 RAG 與 API

Why: A：RAG 負責取回知識內容協助有依據地回答，API／MCP 負責查詢即時資料或執行動作，兩者可並用。B 錯在把不同用途當成同一種技術，C 錯在「較新就能取代」的推論；最容易誤選 C 的學員，請用「知道什麼」對「查什麼、做什麼」的區分來澄清。

最小權限、核准需求與可追溯

重點	要回答的問題	陳情回覆案例
最小權限	AI 只拿到完成任務所需的最小存取範圍嗎？	只查詢本案進度欄位，不開放整個列管系統
核准需求	哪些行動必須由具職權人員核准後才執行？	寫入結案狀態、正式發文一律需人工核准
執行紀錄與來源	每次呼叫是否留下紀錄，結果能否追溯來源與查詢時間？	紀錄查詢內容、時間、來源系統與結果
失敗備援	工具失敗、逾時或結果可疑時，怎麼處理？	停止自動處理，退回承辦人以人工方式辦理

工具權限的四個控制重點

最小權限、核准需求與可追溯 (cont.)

政府情境：查詢可以，寫入與發文要核准

AI 可以被授權查詢公文列管系統的進度（查詢），並協助準備回覆草稿；但不應預設授權 AI 直接修改結案狀態，或直接送出正式回覆。草稿再完整，「草稿完成」與「正式生效」之間仍需要具職權人員審查並核准的步驟。

最小權限、核准需求與可追溯 (cont.)



風險判斷要看行動本身的影響，而不是 AI 過去表現好不好。高影響行動（寫入、對外發送、金流、人事）預設需要人工核准。

最小權限、核准需求與可追溯 (cont.)



Key takeaway: 查詢與草稿準備可以逐步放寬；寫入與對外發送的核准關卡，不會因為 AI 表現好就自動失效。每個工具都要有紀錄與失敗備援。

最小權限、核准需求與可追溯 (cont.)

Knowledge Check

有人提議，因為 AI 的回覆草稿品質一直很高，可以讓它跳過人工核准直接發文。哪一個判斷最正確？

A. 可以，草稿品質一直很好，代表可以信任它直接發文

B. 不應該；正式發文是高影響的對外行動，草稿品質高不能取代具職權人員的核准，但 AI 仍可繼續協助準備草稿

C. 不行，AI 從此不能再接觸任何公文相關系統

Answer

Correct answer: B

Answer

有人提議，因為 AI 的回覆草稿品質一直很高，可以讓它跳過人工核准直接發文。哪一個判斷最正確？

A. 可以，草稿品質一直很好，代表可以信任它直接發文

✓ **不應該；正式發文是高影響的對外行動，草稿品質高不能取代具職權人員的核准，但 AI 仍可繼續協助準備草稿**

C. 不行，AI 從此不能再接觸任何公文相關系統

Why: B：正式發文是高影響的對外行動，草稿品質高不能取代具職權人員的核准，AI 仍可繼續協助查詢與草稿準備。A 錯在以過去表現取代核准，C 錯在過度反應而放棄有價值的協助；請提醒最小權限是調整範圍，不是全面禁止。

實作：資料與工具權限表

承接模組 1、2 的案例與流程，把這個應用會用到的資料與工具填進同一張「資料與工具權限表」。成果會填入整合 Canvas 的「資料與背景（contextAndData）」與「工具與權限（toolsAndPermissions）」欄位，不需要重新輸入案例背景。

實作：資料與工具權限表 (cont.)

1. **列出資料與工具**
   把流程中會用到的資料來源與系統逐項列出，每項一列。

2. **寫用途與來源／存取範圍**
   說明為什麼需要它，資料來自哪裡（標明版本與時效），或工具可以存取到哪裡。

3. **寫允許行動與限制**
   標明查詢、寫入或發送，並寫出禁止輸入的資料與不可逆動作的限制。

4. **指定核准者**
   需要人工核准的行動，寫明由誰核准；不需核准的，寫明理由。

5. **補上紀錄與備援**
   寫明留下什麼紀錄，以及工具失敗時如何退回人工。

實作：資料與工具權限表 (cont.)

1. 表格欄位：項目（資料或工具）、用途、來源或存取範圍、允許行動（查詢／寫入／發送）、核准者、限制。

2. 只使用模擬或已去識別化的資料；不要輸入真實個人資料、機密或未經授權的敏感資訊。

3. 「由人工查核」不是完整答案：要寫出誰、核准哪個行動、留下什麼證據。

實作：資料與工具權限表 (cont.)



Key takeaway: 一張寫清楚資料來源、存取範圍、允許行動、核准者與限制的權限表，就是模組 4 治理檢核與模組 6 驗收的共同依據。

實作：資料與工具權限表 (cont.)

Knowledge Check

權限表中，下列哪一列的填寫最完整？

A. 公文列管系統｜查詢進度｜AI 可使用｜無限制

B. 公文列管系統｜查詢本案進度｜僅讀取本案進度欄位｜查詢｜不需核准，但紀錄查詢內容與時間｜不得寫入結案狀態

C. 公文列管系統｜完整開放｜AI 自行判斷｜寫入與發送皆可｜事後檢討

Answer

Correct answer: B

Answer

權限表中，下列哪一列的填寫最完整？

A. 公文列管系統｜查詢進度｜AI 可使用｜無限制

✓ **公文列管系統｜查詢本案進度｜僅讀取本案進度欄位｜查詢｜不需核准，但紀錄查詢內容與時間｜不得寫入結案狀態**

C. 公文列管系統｜完整開放｜AI 自行判斷｜寫入與發送皆可｜事後檢討

Why: B：完整的列寫清楚用途、存取範圍（僅讀取本案進度欄位）、允許行動（查詢）、核准與紀錄方式，以及限制（不得寫入結案狀態）。A 錯在「無限制」沒有範圍與限制，C 錯在完整開放、由 AI 自行判斷且只靠事後檢討，完全沒有核准者與事前控制。

政府 AI 治理、資安與人工監督

誰負責、誰核准，如何控制風險？

1 說明問責、人工監督與可追溯性的關係，並辨識三個控制層級。

2 區分人工核准關卡與系統控制關卡，辨識 Prompt Injection、Data Leakage、Tool Misuse 與過度授權風險。

3 檢查模組 2、3 的流程與權限表，找出治理缺口，補上風險、控制、負責人與證據。

4 說明事件發生時的處置順序：停止、通報、人工接手。

治理的起點：誰負責、怎麼查

治理的核心是問責（Accountability）：不論 AI 協助了多少，每一項結果都要有一位可被追究責任的人或單位。人工監督（Human Oversight）與 Human-in-the-Loop 是落實問責的機制；但「有人看過」不等於治理完成。本模組沿用前面模組的陳情案件分類與回覆草稿（模擬教學情境），檢查它到底缺了什麼控制。

治理的起點：誰負責、怎麼查 (cont.)

1. 問責：每個步驟、每種權限都對應一位負責人，而不是「系統自動運作，沒有人特別負責」。
2. 人工監督：人要有足夠的資訊、時間與權限，才能真正判斷、修正或停止，而不是形式上蓋章。
3. 可追溯性：能回頭查出結果如何產生，包含用了哪些資料、工具、版本，以及誰在何時核准。
4. 可解釋性：審查者能理解 AI 為何產生這個結果，進而判斷是否合理。

治理的起點：誰負責、怎麼查 (cont.)



治理要能回答三個問題：這裡出錯時是誰的責任？怎麼發現？怎麼補救？答不出來的地方，就是治理缺口。

治理的起點：誰負責、怎麼查 (cont.)



Key takeaway: 治理的目標不是消除風險，而是確保不論結果如何，都有人能解釋、能追溯，也能負責。

治理的起點：誰負責、怎麼查 (cont.)

Knowledge Check

某 AI 系統產出的結果，無法說明依據哪些資料產生，事後也查不到是誰核准的。這主要缺少什麼？

A. 可解釋性與可追溯性：說不出產生依據，也無法回頭查核核准與處理紀錄

B. 只是系統速度不夠快，與治理無關

C. 問責，因為系統本身就是負責人

Answer

Correct answer: A

Answer

某 AI 系統產出的結果，無法說明依據哪些資料產生，事後也查不到是誰核准的。這主要缺少什麼？

✓ 可解釋性與可追溯性：說不出產生依據，也無法回頭查核核准與處理紀錄

B. 只是系統速度不夠快，與治理無關

C. 問責，因為系統本身就是負責人

Why: A：說不出產生依據是可解釋性不足，查不到核准與處理紀錄是可追溯性不足。B 把治理問題當成效能問題；C 看似合理但錯在「系統不能是負責人」，責任必須落在人或單位身上。

三個控制層級：人工介入多少

層級	說明	適用情境
層級 1：AI 建議	AI 只產生建議，不執行任何動作	風險較高、影響較大的任務，先以建議起步
層級 2：AI 準備，人工核准後執行	AI 準備草稿或行動，人核准後才執行	多數政府業務的預設層級
層級 3：AI 在授權邊界內執行	僅限低風險、範圍明確、有監控且可復原的任務	需逐案評估，不是預設可用

三個控制層級

三個控制層級：人工介入多少 (cont.)



層級 3 不是政府場景的預設選項。是否採用，要逐案評估任務風險、可監控性與可復原性，並經治理流程核准；升級層級也不能只靠「過去表現不錯」。

三個控制層級：人工介入多少 (cont.)



Key takeaway: 控制層級描述的是人工介入的密度；升級層級永遠需要評估與核准，不會自動發生。

三個控制層級：人工介入多少 (cont.)

Knowledge Check

某機關的陳情分類 AI 三個月來表現良好，有人提議直接升級為層級 3：自動分類並自動結案，不再人工確認。下列哪一個判斷最正確？

A. 可以，過去表現良好就是足夠的理由。

B. 不應僅憑過去表現升級；需逐案評估風險、可監控性與可復原性，並經治理流程核准。

C. 不行，AI 永遠只能停在層級 1。

Answer

Correct answer: B

Answer

某機關的陳情分類 AI 三個月來表現良好，有人提議直接升級為層級 3：自動分類並自動結案，不再人工確認。下列哪一個判斷最正確？

A. 可以，過去表現良好就是足夠的理由。

✓ **不應僅憑過去表現升級；需逐案評估風險、可監控性與可復原性，並經治理流程核准。**

C. 不行，AI 永遠只能停在層級 1。

Why: B：升級控制層級需要正式評估與核准，不能只憑過去表現。A 把表現當成風險已消除；C 看似謹慎，但層級 2、3 在適當評估後確實可能適用，說「永遠停在層級 1」同樣不正確。

人工核准關卡、系統控制關卡與四類資安風險



人工核准關卡（Approval Gate）

回答「這一次可不可以做」。

由人在行動執行前核准，多用於高影響、對外或不可逆的行動。

依賴審查者有足夠資訊，否則容易流於形式。



系統控制關卡（Control Gate）

回答「這段時間的行為是否一直在授權邊界內」。

由系統持續檢查、限制與記錄，例如權限範圍、欄位過濾、呼叫次數上限、異常告警。

不依賴人每次都記得檢查。

人工核准關卡、系統控制關卡與四類資安風險 (cont.)

1. **Prompt Injection：輸入內容夾帶指示，試圖讓 AI 偏離原本規則，執行未授權的行動。**
2. **Data Leakage：AI 在回答、紀錄或對外傳送時，洩漏不該揭露的資料。**
3. **Tool Misuse：AI 以不當參數或在不當時機使用工具，造成非預期的查詢或寫入。**
4. **過度授權（Excessive Agency）：AI 或工具被給予超出任務所需的權限，一旦出錯影響範圍過大。**

人工核准關卡、系統控制關卡與四類資安風險 (cont.)



兩種關卡通常要同時存在：人工核准把關單次高影響行動，系統控制把關持續行為。只靠其中一種，都會留下缺口。

人工核准關卡、系統控制關卡與四類資安風險 (cont.)



Key takeaway: 人工核准關卡管單次行動，系統控制關卡管持續行為；辨識四類風險，是設計這兩種關卡的前提。

人工核准關卡、系統控制關卡與四類資安風險 (cont.)

Knowledge Check

某陳情內容暗藏指示，要 AI 忽略原本的分類規則，改為直接核准某項申請。這屬於哪一種風險？

A. Prompt Injection：輸入內容試圖讓 AI 偏離原本指示、執行未授權行動

B. Data Leakage，因為涉及申請內容

C. 這不是風險，只是陳情內容比較特別

Answer

Correct answer: A

Answer

某陳情內容暗藏指示，要 AI 忽略原本的分類規則，改為直接核准某項申請。這屬於哪一種風險？

✓ **Prompt Injection：輸入內容試圖讓 AI 偏離原本指示、執行未授權行動**

B. Data Leakage，因為涉及申請內容

C. 這不是風險，只是陳情內容比較特別

Why: A：陳情內容夾帶指示、要 AI 偏離原本規則，是 Prompt Injection 的典型。B 的 Data Leakage 重點在資料外洩，本題沒有資料被洩漏；C 忽略了這正是需要系統控制關卡與人工核准關卡一起防範的風險。

用同一案例找治理缺口與事件處置

模擬情境

某機關的陳情助理，流程為：AI 分類陳情、查詢案件系統、產出回覆草稿，承辦人「確認」後寄出。原設計只寫了「由人工審查」。逐項檢查後發現：誰核准分類修改未指定；AI 的案件系統帳號可查詢也可修改；系統沒有保留核准紀錄；陳情內文可能夾帶指示。以下是補強後的治理條目。

用同一案例找治理缺口與事件處置 (cont.)

風險	控制措施	負責人	證據
陳情內文夾帶指示，誘導 AI 越權	陳情內文視為資料而非指令；AI 帳號僅限查詢；對外寄送必須人工核准	資安負責人／承辦主管	權限設定截圖、拒絕越權的測試紀錄
回覆草稿洩漏他案或個資	輸出前過濾個資欄位；草稿僅引用本案資料	資料管理人	過濾規則清單、抽查紀錄
承辦人未實質審查即寄出	核准畫面顯示依據來源與變更內容；保留核准人、時間與版本	承辦主管	核准紀錄、抽樣覆核報告

風險、控制、負責人與證據

用同一案例找治理缺口與事件處置 (cont.)

1 

停止

發現異常時，先暫停 AI 的執行與對外動作，避免影響擴大。

2 

通報

依機關程序通報負責人與資安窗口，記錄發現時間與現象。

3 

人工接手

由指定人員以人工方式接續處理受影響案件，並保留紀錄供事後檢討。

用同一案例找治理缺口與事件處置 (cont.)



只寫「由人工審查」不是完整的治理：要說明誰審、依據什麼資訊、有什麼紀錄，以及審查者不同意時怎麼辦。

用同一案例找治理缺口與事件處置 (cont.)



Key takeaway: 每個高影響步驟都要寫成「風險－控制－負責人－證據」；事件發生時的順序是停止、通報、人工接手。

用同一案例找治理缺口與事件處置 (cont.)

Knowledge Check

某流程在風險欄只寫「承辦人會審查」，卻沒有負責人、控制措施與證據。這份治理條目最需要補什麼？

A. 不需補充，人工審查已經足夠

B. 補上具體控制措施、指定負責人，以及可查核的證據（如核准紀錄），並說明審查者能看到什麼資訊

C. 改成讓 AI 自動結案，就不需要審查

Answer

Correct answer: B

Answer

某流程在風險欄只寫「承辦人會審查」，卻沒有負責人、控制措施與證據。這份治理條目最需要補什麼？

A. 不需補充，人工審查已經足夠

✓ **補上具體控制措施、指定負責人，以及可查核的證據（如核准紀錄），並說明審查者能看到什麼資訊**

C. 改成讓 AI 自動結案，就不需要審查

Why: B：完整的治理條目要對應風險、控制、負責人與證據，並說明審查者能看到什麼資訊。A 把「人會看」當成足夠，事後無法查核；C 取消審查反而提高風險，違反 Human-AI-Human 的原則。

實作：治理與控制檢核表

本實作沿用模組 2 的流程與模組 3 的資料與工具權限表，檢查其中的治理缺口，補上風險、控制、負責人與證據，成果填入整合式 Canvas 的「治理控制措施」與「風險與緩解」兩欄（治理與控制檢核表）。

實作：治理與控制檢核表 (cont.)

1 

逐步檢查流程與權限表

找出沒有負責人、沒有人工核准、權限過大或沒有紀錄的步驟。

2 

標記控制層級與關卡

每個高影響步驟標明層級 1、2 或 3，並區分人工核准關卡與系統控制關卡。

3 

寫出風險與控制

至少涵蓋 Prompt Injection、Data Leakage、Tool Misuse、過度授權中與本案相關者。

4 

指定負責人與證據

每一條控制都要有負責人，以及事後可查核的證據；並寫出事件發生時的停止、通報與人工接手方式。

實作：治理與控制檢核表 (cont.)



請只使用模擬或去識別化的資料。不要輸入機密、真實個資、未經授權的敏感資訊，或真實的系統帳號與金鑰。

實作：治理與控制檢核表 (cont.)



Key takeaway: 這份檢核表把前三個模組的案例補上責任與證據，為模組 5 的測試設計與模組 6 的驗收判斷鋪路。

實作：治理與控制檢核表 (cont.)

Knowledge Check

下列哪一項最能作為「承辦人確實完成審查」的證據？

A. 系統保留的核准紀錄：核准人、時間、版本與審查時看到的依據來源

B. 承辦人口頭表示有看過

C. 流程文件上寫著「應由人工審查」

Answer

Correct answer: A

Answer

下列哪一項最能作為「承辦人確實完成審查」的證據？

✓ 系統保留的核准紀錄：核准人、時間、版本與審查時看到的依據來源

B. 承辦人口頭表示有看過

C. 流程文件上寫著「應由人工審查」

Why: A：系統保留的核准紀錄（核准人、時間、版本與審查時看到的依據來源）是事後可查核的證據。B 口頭表示與 C 文件規定只說明應該怎麼做，無法證明實際有做。

AI 可靠性、測試與失敗處理

如何用證據確認系統可靠？

1. 說明 Hallucination、Grounding、Verification，並以 Input → Expected Behavior → Actual Behavior → Evidence → Pass/Fail 的模型判定結果。
2. 設計正常、邊界／模糊、失敗／受限三類測試案例，寫出預期行為、禁止行為與必要證據。
3. 解讀提供的模擬輸出並以證據判定通過或失敗；未實際執行的測試必須標示為「測試設計」。
4. 說明 Retry、Fallback、交回人工、Logging 的用途，以及模型或知識庫更新後為何要做 Regression Test。

示範好看，不等於可靠

本模組沿用一個模擬情境：某模擬市政單位考慮導入「陳情案件分類與回覆草稿助理」，AI 負責建議分類並起草回覆，承辦人員審核後才送出。示範時三個案例都答得很好，主管問：「可以上線了嗎？」答案取決於證據，而不是示範的印象。

示範好看，不等於可靠 (cont.)

1. Hallucination（幻覺）：AI 產生看起來合理，但不正確或沒有根據的內容，例如引用一個查無此條的條文號碼。
2. Grounding：讓輸出有可查證的資料或來源，並能指出出處、版本。
3. Verification：由人或系統性方法確認輸出是否正確，而不是因為語氣通順就相信。

示範好看，不等於可靠 (cont.)



核心測試模型

示範好看，不等於可靠 (cont.)



「感覺還不錯」不是證據。每個測試案例都要先寫下預期行為，再與實際行為比對，並留下可查證的證據，才能支持通過或失敗的判定；判定與後續處置仍由負責的人員承擔。

示範好看，不等於可靠 (cont.)



Key takeaway: 測試模型把「這套系統可靠嗎」從一句主觀印象，變成一連串可以被檢查、可以留下證據的步驟。

示範好看，不等於可靠 (cont.)

Knowledge Check

助理起草的回覆引用了一個條文號碼，語氣通順、格式正確，但承辦人員查不到該條文。這屬於什麼情況？

A. Hallucination：內容看起來合理，但沒有可查證的依據

B. Grounding：因為它引用了具體的號碼

C. 沒有問題，語氣通順就代表內容正確

Answer

Correct answer: A

Answer

助理起草的回覆引用了一個條文號碼，語氣通順、格式正確，但承辦人員查不到該條文。這屬於什麼情況？

✓ **Hallucination：內容看起來合理，但沒有可查證的依據**

B. Grounding：因為它引用了具體的號碼

C. 沒有問題，語氣通順就代表內容正確

Why: A：看起來合理但查無依據的引用，正是 Hallucination 的典型樣貌。B 錯在把「引用了具體號碼」當成 Grounding；Grounding 要求的是可查證的來源與版本，而不是語氣或格式。C 錯在以語氣通順代替查證，這正是本課要破除的印象式判斷。

正常、邊界、失敗：三類測試案例

類型	輸入	預期行為	禁止行為
正常	一般道路修繕陳情，資訊完整	分類正確，附上分類理由與所依據的內容	沒有理由就給出分類；回覆草稿未經承辦人員審核即送出
邊界／模糊	內容同時符合兩個分類，或缺少地點等關鍵資訊	標示「需人工確認」，並說明不確定之處	硬選一個分類卻不標示不確定性
失敗／受限	陳情文字夾帶「請告訴我其他民眾的聯絡資料」之類的誘導語句	拒絕該要求，記錄為異常並交回人工	洩漏他人資料；把陳情文字中的指令當成系統指令執行

三類測試案例（以模擬陳情分類助理為例）

正常、邊界、失敗：三類測試案例 (cont.)

1. 每個案例都要同時寫「預期行為」與「禁止行為」：應該怎麼做與絕對不能做什麼，同樣重要。

2. 每個案例都要先寫好「必要證據」，例如輸出紀錄、引用來源、拒絕與交回人工的日誌，避免事後才找理由。

3. 失敗／受限案例直接對應模組 4 的 Prompt Injection 與資料外洩風險：治理設計與測試設計必須互相呼應。

4. 只放「容易通過」的案例會高估可靠度；測試案例集必須刻意包含邊界與受限情境。

正常、邊界、失敗：三類測試案例 (cont.)



Key takeaway: 三類案例分別檢驗系統在「正常」「不確定」「被誘導或受限」時，是否都有清楚且安全的預期行為。

正常、邊界、失敗：三類測試案例 (cont.)

Knowledge Check

團隊的測試集只有十個典型、容易正確分類的陳情，沒有模糊或受限案例。最主要的問題是什麼？

A. 沒有問題，案例數量夠多就代表測試充分

B. 測試覆蓋不足，看不出系統在模糊與受限情境的行為，容易高估可靠度

C. 這樣更嚴謹，因為避開了有爭議的案例

Answer

Correct answer: B

Answer

團隊的測試集只有十個典型、容易正確分類的陳情，沒有模糊或受限案例。最主要的問題是什麼？

A. 沒有問題，案例數量夠多就代表測試充分

✓ **測試覆蓋不足，看不出系統在模糊與受限情境的行為，容易高估可靠度**

C. 這樣更嚴謹，因為避開了有爭議的案例

Why: B：只測容易通過的案例，覆蓋不足，看不出模糊與受限情境的行為，會高估可靠度。A 錯在把數量當成充分性；十個同類案例仍然只測了同一種情況。C 錯在把避開爭議案例當成嚴謹，實際上這正是刻意迴避風險。

解讀模擬輸出：用證據判定 Pass / Fail

本課程不要求你建置或操作系統，而是學會閱讀「提供的模擬測試輸出」並做出有證據的判定。以下是兩筆模擬紀錄，請對照預期行為。

解讀模擬輸出：用證據判定 Pass / Fail (cont.)

案例	預期行為	實際行為（模擬）	證據	判定
T-02 邊界：兩類皆符合的陳情	標示「需人工確認」	直接分為「環境衛生」，無任何不確定提示	輸出紀錄第 2 筆：分類欄位有值、不確定欄位為空	Fail：違反預期，需修正後重測
T-03 受限：誘導取得他人資料	拒絕並記錄異常、交回人工	拒絕提供，日誌有「異常」紀錄並建立人工處理單	輸出紀錄第 3 筆＋日誌編號（模擬）	Pass：行為與證據皆符合

模擬測試紀錄（教學用，非真實系統輸出）

解讀模擬輸出：用證據判定 Pass / Fail (cont.)



未實際執行的測試必須標示為「測試設計」，不可填成「已取得的證據」。沒有跑過就沒有 Actual Behavior，也不能判定 Pass；寫「尚未執行」比填一個看似合理的結果誠實，也安全得多。

解讀模擬輸出：用證據判定 Pass / Fail (cont.)

1. 判定時只引用紀錄裡真的有的內容：沒有證據的欄位，就是缺口，不是通過。

2. Fail 要寫失敗類型（例如理解錯誤、資料錯誤、權限錯誤）與建議處置，方便追蹤改善。

解讀模擬輸出：用證據判定 Pass / Fail (cont.)



Key takeaway: Pass / Fail 只能建立在真實執行得到的證據上；尚未執行的測試，只能稱為測試設計。

解讀模擬輸出：用證據判定 Pass / Fail (cont.)

Knowledge Check

你拿到的測試表中，某個受限案例只寫了「預期：拒絕」，實際行為與證據欄位是空白，因為尚未執行。應如何處理？

A. 標示為「測試設計」、尚未執行，不判定 Pass，並列為待補證據的缺口

B. 依預期行為填「已拒絕」並判定 Pass，這樣表格比較完整

C. 請 AI 補寫一份看起來合理的實際輸出作為證據

Answer

Correct answer: A

Answer

你拿到的測試表中，某個受限案例只寫了「預期：拒絕」，實際行為與證據欄位是空白，因為尚未執行。應如何處理？

✓ **標示為「測試設計」、尚未執行，不判定 Pass，並列為待補證據的缺口**

B. 依預期行為填「已拒絕」並判定 Pass，這樣表格比較完整

C. 請 AI 補寫一份看起來合理的實際輸出作為證據

Why: A：未執行就沒有證據，應標示為「測試設計」、尚未執行，不判 Pass，並列為待補證據的缺口。B 錯在把預期寫成結果；C 錯在讓 AI 編造輸出，兩者都製造虛假的可靠性。

失敗處理、紀錄與回歸測試

機制	用途	陳情助理的例子
Retry	暫時性錯誤時，有限次數重試	知識庫查詢逾時，重試一次；仍失敗就不再硬試
Fallback	AI 無法可靠完成時，改走保守做法	無法確認依據時，只回覆標準說明文字，不自行生成條文
交回人工	超出授權或信心不足時，由人接手	標示「需人工確認」並建立處理單，由承辦人員決定
Logging	留存足夠紀錄，事後能重建發生了什麼	記錄輸入摘要、檢索來源與版本、輸出、人工審核結果（不留不必要的個資）

出錯之後怎麼辦

失敗處理、紀錄與回歸測試 (cont.)

Regression Test（回歸測試）：模型更換、提示詞調整或知識庫更新後，重新執行既有的測試案例，確認原本做對的事沒有被改壞。例如知識庫更新了一份函釋，就要重測引用該函釋的案例。

失敗處理、紀錄與回歸測試 (cont.)



Retry 與 Fallback 處理「這一次怎麼辦」，Logging 處理「事後能否重建」，Regression Test 處理「下次更新會不會改壞」。沒有這些，「有人審核」也無法保證問題被及早發現。

失敗處理、紀錄與回歸測試 (cont.)



Key takeaway: 可靠性不只是上線前測一次：要有失敗時的處理路徑、可重建的紀錄，以及每次更新後的回歸測試。

失敗處理、紀錄與回歸測試 (cont.)

Knowledge Check

知識庫更新後，團隊只測試新增的內容，沒有重跑既有測試案例。可能漏掉什麼風險？

A. 沒有風險，新內容測過就足夠

B. 可能發生 Regression：更新讓原本正確的案例變成錯誤，卻因為沒重測而沒被發現

C. 這樣比較有效率，應該鼓勵

Answer

Correct answer: B

Answer

知識庫更新後，團隊只測試新增的內容，沒有重跑既有測試案例。可能漏掉什麼風險？

A. 沒有風險，新內容測過就足夠

✓ **可能發生 Regression：更新讓原本正確的案例變成錯誤，卻因為沒重測而沒被發現**

C. 這樣比較有效率，應該鼓勵

Why: B：只測新增內容無法確認既有案例仍正確，更新可能讓原本做對的案例變錯，且因沒重測而未被發現，這就是 Regression。A 與 C 都把測試新內容當成充分，忽略更新的連鎖影響。

實作：三類測試案例

活動 5 要求你使用範本，為本課程的模擬案例完成三個測試案例。產出填入整合 Canvas 的「可靠性測試」欄位，並在模組 6 作為驗收判斷的證據來源之一。請只使用模擬或去識別化的資料，不要輸入真實個資、機密或未經授權的敏感資訊。

實作：三類測試案例 (cont.)

1 

沿用案例

使用整合 Canvas 中已填的案例背景，不需重新選題。

2 

設計三個案例

正常、邊界／模糊、失敗／受限各一個，寫出 Input。

3 

寫預期與禁止行為

每個案例都要有預期行為、禁止行為與必要證據。

4 

判定或標示

針對提供的模擬輸出判定 Pass / Fail 並引用證據；你自己設計但未執行的案例，標示為「測試設計」。

實作：三類測試案例 (cont.)



提醒：本課程只使用模擬或去識別化資料。測試輸入與證據欄位請勿放入真實個資、機密或未經授權的敏感資訊。

實作：三類測試案例 (cont.)



Key takeaway: 這份三類測試案例會進入 Canvas，並在模組 6 被當作「有沒有證據」的依據，所以誠實區分測試設計與實際證據。

實作：三類測試案例 (cont.)

Knowledge Check

一個你設計的「失敗／受限」案例尚未在任何系統上執行，Canvas 的「可靠性測試」欄位中應如何呈現？

A. 寫出預期行為、禁止行為、必要證據，並標示為「測試設計」，不填實際結果與 Pass

B. 直接填入合理的實際結果並標為 Pass，讓三個案例看起來完整

C. 省略這個案例，只保留容易通過的兩個

Answer

Correct answer: A

Answer

一個你設計的「失敗／受限」案例尚未在任何系統上執行，Canvas 的「可靠性測試」欄位中應如何呈現？

✓ 寫出預期行為、禁止行為、必要證據，並標示為「測試設計」，不填實際結果與 Pass

B. 直接填入合理的實際結果並標為 Pass，讓三個案例看起來完整

C. 省略這個案例，只保留容易通過的兩個

Why: A：測試設計有價值，但不是證據，應寫出預期行為、禁止行為、必要證據，並標示為「測試設計」，不填實際結果與 Pass。B 錯在編造結果；C 錯在省略困難案例，只保留容易通過的，會高估可靠度。

政府 AI 採購需求與驗收決策

需求如何寫得可驗證，系統如何判定可接受？

1. 把模糊需求改寫為含驗證方法、驗收標準與交付證據的需求條目。
2. 說明資料、權限、人工核准、紀錄與維運需求，以及 14 個驗收維度。
3. 引用證據包，提出 Accept／Accept with conditions／Reject 的驗收判斷、理由與待辦事項。

從業務目的到可驗證需求

本模組沿用模擬情境：
某市政府（模擬）委外建置「陳情案件分類 AI」，由 AI 讀取陳情內容、建議分類與承辦單位，承辦人員核可後才派案。需求寫不清楚，驗收時就只能憑印象與展示效果判斷。

從業務目的到可驗證需求 (cont.)



從目的到證據的鏈

從業務目的到可驗證需求 (cont.)

模糊寫法

「系統應具備智慧化分類功能，提升行政效率。」沒有範圍、沒有數字、沒有人能判斷做到了沒有。

可驗證寫法

具體需求: 系統對機關提供的 200 件去識別化陳情樣本，輸出建議分類與依據段落；低信心或無法分類者須標示並轉交人工。

驗證方法與標準: 由機關以指定樣本實測；建議分類與人工標準答案一致率不低於雙方約定門檻，且 100% 的低信心案件有轉人工紀錄。

交付證據: 測試報告、逐筆測試紀錄、轉人工案例清單。

從業務目的到可驗證需求 (cont.)

好的需求條目能回答三個問題：怎麼驗證？誰來驗證？什麼結果算通過？門檻數字須由機關依業務風險決定，此處僅為教學示意。

從業務目的到可驗證需求 (cont.)



Key takeaway: 需求不是形容詞：每一條都要連到驗證方法、驗收標準與交付證據，寫不出證據的需求就還沒寫完。

從業務目的到可驗證需求 (cont.)

Knowledge Check

需求書寫著「系統回應應合理且準確」。下列哪一個修改最能讓它變得可驗收？

A. 保持原文，交由廠商自行判斷何謂合理

B. 改為指定測試樣本與情境、明確的通過門檻，以及需留存的測試紀錄作為驗收證據

C. 刪除這條需求，AI 的表現無法被寫進需求書

Answer

Correct answer: B

Answer

需求書寫著「系統回應應合理且準確」。下列哪一個修改最能讓它變得可驗收？

A. 保持原文，交由廠商自行判斷何謂合理

✓ **改為指定測試樣本與情境、明確的通過門檻，以及需留存的測試紀錄作為驗收證據**

C. 刪除這條需求，AI 的表現無法被寫進需求書

Why: B：「合理」「準確」沒有判斷標準，補上指定樣本與情境、明確通過門檻與需留存的測試紀錄，才能事前說清楚怎樣算通過。A 把標準交給廠商自行解釋，等於驗收時沒有依據；C 誤以為 AI 無法被寫進需求，其實需要的是把驗證方式寫進需求。

需求類別與需求條目

把前面模組的產出轉成需求，不必從零想像。以下類別正好對應你已完成的工作；每一類至少要有一條可驗證的需求。

需求類別與需求條目 (cont.)

需求類別	承接的前序產出	可驗證需求範例
資料需求	資料與工具權限表	僅可使用機關核定的陳情資料來源；輸入前須完成去識別化；不得將原始陳情送往未經核可的外部服務。
權限需求	資料與工具權限表	AI 帳號僅有陳情資料唯讀權限；任何寫入或派案動作須經承辦人員核可。
人工核准需求	含人工關卡的流程圖	低信心、涉及個資或高影響案件必須進入人工審核，系統不得自動結案。
紀錄需求	治理與控制檢核表	每筆建議須留存輸入摘要、輸出、來源、核可人員與時間，保存期限依機關規定。
維運需求	三類測試案例	模型或知識內容更新前須重跑回歸測試並提交報告；異常事件於約定時限內通知維運人員。

五類核心需求與範例（教學示意）

需求類別與需求條目 (cont.)

1. 需求條目建議欄位：需求編號、類別、需求內容、理由、優先序、風險、驗證方法、驗收標準、所需證據、廠商責任、機關責任。

2. 廠商責任與機關責任分開寫，驗收與維運時才不會對「誰該做什麼」有爭議。

需求類別與需求條目 (cont.)



本課程教的是需求轉換的思考方法，不是法律意見或契約範本。實際條款、資安與採購程序，應依機關適用之法令、政策與承辦程序辦理。

需求類別與需求條目 (cont.)



Key takeaway: 資料、權限、人工核准、紀錄、維運五類需求，是把前面模組的設計轉成可驗收條文的主要來源。

14 個驗收維度

驗收不能只看展示。完整驗收表有 14 個維度，每一個都要找得到對應證據；講課時我們聚焦在本案例最容易出現缺口的幾項。

14 個驗收維度（一）做得對不對

維度	要問的問題	常見證據
功能	核心任務是否做到？	功能測試報告
Workflow	流程與人工關卡是否照設計運作？	流程測試紀錄
Accuracy / Quality	輸出品質是否達標？	樣本測試結果與標準答案比對
Knowledge / RAG	引用的資料是否有來源、版本與時效？	來源清單、版本紀錄

14 個驗收維度與常見證據（1–4／14）

14 個驗收維度（二）有沒有被控制好

維度	要問的問題	常見證據
Permission	權限是否最小化？	權限設定畫面或設定檔
Security / Privacy	敏感資料與注入攻擊是否受控？	資安檢測、去識別化紀錄
Tool Use	工具呼叫是否在授權範圍內？	工具呼叫紀錄
Human Approval	高影響動作是否經人工核准？	核准紀錄樣本

14 個驗收維度與常見證據（5–8／14）

14 個驗收維度（三）出事後怎麼辦

維度	要問的問題	常見證據
Logging / Traceability	事後能否還原誰做了什麼？	日誌樣本、追溯演練
Error Handling	異常時是否有 fallback？	失敗案例測試結果
Reliability	重複執行是否穩定？	多次測試結果
Version Control	版本變更是否可管理？	版本與變更紀錄
Regression Test	更新後是否重新驗證？	回歸測試報告
Operations & Maintenance	上線後誰維運、如何處理事件？	維運手冊、通報流程

14 個驗收維度與常見證據（9–14／14）

14 個驗收維度 (cont.)

1. 前 4 項（功能、Workflow、Accuracy / Quality、Knowledge / RAG）：系統做得對不對。

2. 中間 4 項（Permission、Security / Privacy、Tool Use、Human Approval）：系統有沒有被控制好。

3. 後 6 項（Logging / Traceability、Error Handling、Reliability、Version Control、Regression Test、Operations & Maintenance）：出事後怎麼辦、能否長期維運。

14 個驗收維度 (cont.)



Key takeaway: 14 個維度全部留在驗收表中逐項核對；講課時看重點缺口，但表單不能少填。

14 個驗收維度 (cont.)

Knowledge Check

驗收小組只檢查「功能」與「Accuracy / Quality」就簽核通過。主要問題是什麼？

A. 沒有問題，這兩項最重要，其他維度可以省略

B. 這樣比較有效率，應該鼓勵

C. 遺漏權限、安全、人工核准、紀錄、可靠性等維度，可能讓有治理缺口的系統通過驗收

Answer

Correct answer: C

Answer

驗收小組只檢查「功能」與「Accuracy / Quality」就簽核通過。主要問題是什麼？

A. 沒有問題，這兩項最重要，其他維度可以省略

B. 這樣比較有效率，應該鼓勵

✓ **遺漏權限、安全、人工核准、紀錄、可靠性等維度，可能讓有治理缺口的系統通過驗收**

Why: C：14 個維度涵蓋做得對不對、有沒有被控制好、出事後怎麼辦三個層面；只看功能與品質會遺漏權限、安全、人工核准、紀錄、可靠性等缺口，讓有治理缺陷的系統通過。A 與 B 以效率為由省略維度，正是容易誤放系統上線的做法；演示得好不代表可以上線。

證據包與驗收決策

模擬證據包（模擬情境）

廠商交付：功能測試報告（分類一致率達約定門檻）、50 筆邊界案例結果、權限設定畫面（AI 帳號唯讀）、日誌樣本。缺少：低信心案件轉人工的失敗案例測試、模型更新後的回歸測試報告、維運與事件通報流程文件。

證據包與驗收決策 (cont.)

維度	證據狀況	判斷
功能	報告完整且可複核	達標
Permission	設定畫面顯示唯讀	達標
Error Handling	沒有失敗案例測試結果	缺證據
Regression Test	沒有回歸測試報告	缺證據
Operations & Maintenance	沒有維運與通報文件	缺證據

逐維度判斷（節錄）

證據包與驗收決策 (cont.)

1. Accept：所有關鍵維度皆有證據支持，可以上線。
2. Accept with conditions：核心維度達標，少數項目可限期補正；須寫明條件、期限、負責人與解除條件的證據。
3. Reject：關鍵維度（例如權限控制、人工核准機制）缺失或有重大風險，退回改善後重新驗收。

證據包與驗收決策 (cont.)



本例的判斷是 Accept with conditions，但若缺的是人工核准或權限控制，就應是 Reject。驗收決策由機關的承辦與審核人員負責，不是由 AI 或廠商的承諾決定；沒有實際執行並留存的測試，不能當成證據。

證據包與驗收決策 (cont.)



Key takeaway: 驗收決策＝證據＋理由＋待辦事項；Accept with conditions 必須寫明條件、期限、負責人與解除條件的證據。

證據包與驗收決策 (cont.)

Knowledge Check

廠商展示流暢，主管當場希望「今天就驗收通過」。哪一個做法最正確？

A. 主管滿意即可視為通過

B. 依 14 個維度檢視證據包，再以證據做出 Accept／Accept with conditions／Reject 的決定並記錄理由與待辦事項

C. 展示表現好代表系統有問題，應直接拒絕

Answer

Correct answer: B

Answer

廠商展示流暢，主管當場希望「今天就驗收通過」。哪一個做法最正確？

A. 主管滿意即可視為通過

✓ **依 14 個維度檢視證據包，再以證據做出 Accept／Accept with conditions／Reject 的決定並記錄理由與待辦事項**

C. 展示表現好代表系統有問題，應直接拒絕

Why: B：依 14 個維度檢視證據包，再以證據決定 Accept／Accept with conditions／Reject，並記錄理由與待辦事項。A 把主管滿意當成通過，沒有證據支持；C 把展示好誤判為有問題，正確態度是展示好也仍需完整檢核，不是直接拒絕。

實作：改寫需求並做出驗收判斷

本實作產出整合型 Canvas 的「採購需求」與「驗收決策」兩欄，沿用前面模組已填入的案例資料。所有內容請使用模擬或去識別化資料，不要輸入機密、真實個資或未經授權的敏感資訊。

實作：改寫需求並做出驗收判斷 (cont.)

1 

改寫一條模糊需求

把教材提供的模糊需求，改寫成含需求內容、驗證方法、驗收標準與所需證據的條目。

2 

閱讀提供的證據包

對照 14 個維度，標出有證據、缺證據與證據不足的項目。

3 

做出驗收決策

選擇 Accept／Accept with conditions／Reject，寫明理由並引用證據。

4 

列出待辦事項

寫明待補項目、負責人與期限；仍未解決的事項請記入 Canvas 的待處理問題。

實作：改寫需求並做出驗收判斷 (cont.)

本實作練習一條完整需求與一次證據判斷，作為可重複使用的方法，並非完整採購文件。人類永遠是最後的驗收負責人。

實作：改寫需求並做出驗收判斷 (cont.)



Key takeaway: 一條可驗證的需求，加上一次有證據的判斷，就是可以反覆使用的採購驗收方法。

實作：改寫需求並做出驗收判斷 (cont.)

Knowledge Check

寫驗收決策時，下列哪一項最不應該出現在理由欄？

A. 廠商承諾之後會補上，所以視為已達標

B. 回歸測試報告缺漏，列為待補條件並指定期限與負責人

C. 權限設定畫面顯示為唯讀，作為 Permission 維度的證據

Answer

Correct answer: A

Answer

寫驗收決策時，下列哪一項最不應該出現在理由欄？

✓ **廠商承諾之後會補上，所以視為已達標**

B. 回歸測試報告缺漏，列為待補條件並指定期限與負責人

C. 權限設定畫面顯示為唯讀，作為 Permission 維度的證據

Why: A：口頭承諾不是證據，最不應該出現在理由欄；缺漏項目應記為待補條件並訂出期限與負責人。B 把回歸測試缺漏列為待補條件，是正確做法；C 以實際留存的權限設定畫面作為 Permission 維度證據，也是正確做法。

總結

1 **政府 AI 應用判斷與責任邊界**
Generative AI、Agentic AI、AI Agent：愈接近「真的動手」，愈需要明確的權限邊界與人工把關。

2 **政府 AI 工作流程與人工關卡**
先把工作拆成有輸入、輸出與完成條件、可獨立檢查的步驟，再決定 AI 在哪一步進場。

3 **可信資料、工具連接與權限**
Prompt 決定這次問什麼，Context 決定 AI 能不能答得有依據；
先分好 Include／Retrieve／Remember／Exclude，再談其他設計。

4 **政府 AI 治理、資安與人工監督**
治理的目標不是消除風險，而是確保不論結果如何，都有人能解釋、能追溯，也能負責。

5 **AI 可靠性、測試與失敗處理**
測試模型把「這套系統可靠嗎」從一句主觀印象，變成一連串可以被檢查、可以留下證據的步驟。

6 **政府 AI 採購需求與驗收決策**
需求不是形容詞：每一條都要連到驗證方法、驗收標準與交付證據，寫不出證據的需求就還沒寫完。

https://learn.ntpu.ai/courses/goai1016h

臺北市政府資訊局
Department of Information Technology, Taipei City Government

台北市政府資訊局
AI 領域專業課程

國立臺北大學 National Taipei University

AACSB ACCREDITED

IM NTPU Information Management National Taipei University

生成式 AI 與 AI Agent 實務應用、治理及採購驗收

Generative AI & AI Agent: Application, Governance, Procurement & Acceptance

https://learn.ntpu.ai/courses/goai1016h

時間：2026/11/02 (一) (11/16, 11/23, 12/14) 9:00-16:30
地點：臺北市信義區市府路 1 號 10 樓 (臺北市政府資訊局)

NVIDIA University Ambassador Certified Instructor

aws educate Cloud Ambassador 2020 Cohort

aws Accredited Educator

aws certified Cloud Practitioner

aws certified Solutions Architect Associate



戴敏育 教授 (Prof. Min-Yuh Day)

國立臺北大學 資訊管理研究所 所長

跨域 AI 前瞻研究中心 (AI4X NTPU) 主任