Metaage GitHub Enterprise × Entra ID14 天企業現代化導入方案：

打造標準化、安全與自動化的研發治理平台

一、 現況痛點與企業治理風險

1. 身分管理斷裂（Identity Silo）

   企業研發團隊成員異動頻繁，但 GitHub 帳號權限仍多半仰賴人工維護。

   離職或角色異動後，權限平均存在 48 小時以上的回收空窗期，形成重大資安風險。

   同時，企業無法強制所有開發者啟用 MFA，身份盜用事件難以防範。

2. 憑證管理失控（Secret Sprawl）

   Service Principal、API Key、Token 等長效密鑰散落於各個 Repository、Pipeline 與設定檔中。

   缺乏集中管理與到期可視性，一旦外洩，難以快速盤點影響範圍並全面重置。

3. 合規稽核黑盒（Audit Black Hole）

   程式碼的 Push、PR、Merge 行為無法即時對應企業員工身分。

   缺乏統一的品質與資安門檻（Quality Gates），導致安全與程式碼品質高度仰賴個人自律。

二、方案核心目標與設計原則

本方案以「不處理舊資料、專注新標準建立」為核心思維，協助企業快速建立可長期運作的研發治理基準。

核心原則

- 規範先行：新專案一開始即符合企業標準
- 治理內建：身份、權限與安全策略由平台自動落實
- 去密鑰化：全面移除長效密鑰，降低外洩風險
- 標準化複用：讓最佳實務可被不斷複製與延伸

三、 14 天實施時程與教育訓練規劃

天數	階段重點	關鍵任務與交付物
Day 01–03	平台治理藍圖設計	**任務：** 定義 Enterprise/Org 階層；設定 **SCIM 帳號自動化同步**。 **交付：** 權限模型設計書、企業組織治理手冊。
Day 04–06	DevSecOps 環境建置	**任務：** 佈署 **Self-hosted Runners**；建立 OIDC 無密鑰雲端連線。 **交付：** 安全運算架構圖、Actions 網路配置說明。
Day 07	【教育訓練 Day 1】	**基礎操作與團隊協作**（強調新標準規範落地）。
Day 08	【教育訓練 Day 2】	**進階功能與實戰演練**（強調自動化與安全門檻）。
Day 09–12	標準化模板實作	**任務：** 建立 **Reusable Workflows** 與 **Repo Templates**。 **交付：** 企業共用 CI/CD 模板庫、安全掃描政策定義。
Day 13–14	驗收與首批專案導入	**任務：** 協助 1-3 個新專案套用標準化模板上線。 **交付：** 開發規範懶人包、平台維運 Runbook。

四、 核心課程：GitHub 實戰工作坊 (Day 07–08)

【Day 07 | 基礎操作與新世代協作規範】

- **上午：現代化 Git 協作心法**
  - **標準化起手式**：使用企業級 **Repository Templates** 快速建立新專案。
  - **命名美學**：嚴格執行 Branch、Tag、Commit 的命名範例（如 Conventional Commits）。
  - **README 驅動開發**：如何撰寫標準化文件與 .gitignore。
- **下午：Pull Request 協作與品質防線**
  - **標準協作流**：GitHub Flow 分支模型實作（main / feature）。
  - **PR 高效評論**：如何進行有建設性的 Code Review 與使用建議 (Suggestions)。

- **合併管控**：了解 Squash Merge 的好處與分支保護規則 (Branch Protection Rules)。

【Day 08 | 進階功能與 DevSecOps 安全實戰】

- **上午：GitHub Actions 自動化加速**
  - **CI/CD 基礎實務**：自動化編譯、單元測試、品質檢查。
  - **秘密防護**：如何在不洩露任何 Secret 的情況下進行自動化部署。
  - **快取優化**：利用 Cache 縮短開發者等待時間。
- **下午：GitHub Advanced Security (GAS) 安全內建**
  - **靜態掃描 (CodeQL)**：在程式碼合併前自動抓出潛在漏洞。
  - **密鑰防線**：體驗 **Push Protection** 阻擋敏感資訊入庫的機制。
  - **軟體供應鏈安全**：自動產生 SBOM 並監控第三方依賴套件風險 (Dependabot)。

五、 方案價值：為何這是企業導入的首選？

1. **解決「各做各的」混亂**：

透過 **Repository Templates**，工程師開啟新專案時，CI/CD、安全掃描與目錄結構都已預設完成，不必從零寫 YAML。

2. **極致的帳號生命週期管理**：

整合 SCIM，當 HR 系統停權同仁時，其 GitHub 存取權秒級回收，徹底解決「離職員工還能看程式碼」的資安黑洞。

3. **將安全轉化為「開發者體驗」**：

安全檢查直接顯示在 PR 介面，開發者在撰寫程式時就能即時修正，不再需要等待資安部門的厚重報告。

4. **Policy as Code 的治理高度**：

所有的保護規則（如 PR 必須 2 人簽核、必須通過弱掃）都是全組織統一設定，確保合規性。

六、 預期專案效益 (KPI)

- **專案啟動效率**：新專案從建立到符合 CI/CD 規範的時間縮短 **80%**。
- **安全合規達成率**：100% 的新進程式碼異動皆自動通過資安與品質門檻。
- **管理負擔降低**：透過 IdP 與 SCIM 整合，減少 **90%** 的手動權限調整作業。