## **資安事件通報系統 - 期末分組報告**

#### **系統概述**

本作品為一個完整的資安事件通報系統，基於 Flask 框架開發，旨在為普通民眾、管理人員與資安人員提供一個安全、高效的平台，用於通報、處理與管理資安事件。系統採用基於角色的存取控制（RBAC）模型，確保不同使用者僅能存取其職責範圍內的功能。

#### **設計目標**

* **權限控制系統設計：**實作基於角色的存取控制（RBAC），學習多層級權限管理
* **資安系統開發：**確保系統安全性，防止未授權存取或資料洩露
* **全端整合開發：**實現前後端與資料庫的完整整合
* **使用者體驗：**透過賽博龐克風格的 UI 設計提升現代感改善使用者體驗

#### **預期效益**

* 提升資安事件處理效率，建立標準化通報流程
* 確保資料隔離與權限控制，維護系統安全性
* 透過角色設計學習資安系統的權限控制實務
* 建立完整的事件追蹤與處理記錄

**背景資料**

角色與權限相關知識

**RBAC 模型設計**

本系統採用三層 RBAC 架構：

* **一般使用者（User）：**僅能查看和管理自己提交的事件
* **資安人員（SECURITY\_STAFF）：**負責處理指派的資安事件，更新處理狀態
* **管理員（Admin）：**擁有系統最高權限，負責使用者管理和事件指派

### **權限矩陣**

|  |  |  |  |
| --- | --- | --- | --- |
| **功能 / 角色** | **普通民眾** | **管理人員** | **資安人員** |
| 提交資安通報 | ✅ | ❌ | ❌ |
| 查詢自己通報紀錄 | ✅ | ❌ | ❌ |
| 查看所有通報紀錄 | ❌ | ✅ | ✅（僅限指派案件） |
| 指派通報事件 | ❌ | ✅ | ❌ |
| 處理事件並紀錄結果 | ❌ | ❌ | ✅ |
| 修改使用者帳號或角色 | ❌ | ✅ | ❌ |
| 系統參數/通報類別設定 | ❌ | ✅ | ❌ |
| 匯出統計報表 | ❌ | ✅ | ❌ |
| 補充附件與說明（案件內） | ✅ | ❌ | ✅ |

#### **技術架構**

**前端技術**

* **HTML5/CSS3**：結構化網頁內容與響應式設計
* **Jinja2 模板引擎**：動態內容渲染與模板繼承

**後端技術**

* **Flask 框架**：輕量級 Python Web 框架
* **SQLite 資料庫**：嵌入式關聯式資料庫
* **bcrypt 加密**：密碼雜湊與驗證機制
* **Session 管理**：使用者狀態維護

**資料庫設計**

* **使用者表格（users）**：儲存帳號資訊與角色
* **資安事件表格（security\_incidents）**：事件詳細資訊
* **事件記錄表格（incident\_logs）**：處理歷程追蹤
* **事件標籤表格（incident\_tags）**：威脅分類標籤
* **密碼重設表格（password\_reset\_codes）**：重設驗證碼

#### **環境介紹**

**開發環境**

* **作業系統**：Windows 11 Pro for Workstations
* **Python 版本**：3.10
* **開發工具**：Visual Studio Code
* **版本控制**：GitHub & GitHub Destop

#### **專案結構**

security\_system/

├── app.py # 主應用程式

├── requirements.txt

├── README.md

├── LICENSE

├── templates/

│ ├── base.html # 基礎模板

│ ├── login.html # 登入頁面

│ ├── register.html # 註冊頁面

│ ├── dashboard.html # 儀表板

│ ├── report\_incident.html # 通報事件

│ ├── incident\_detail.html # 事件詳情

│ ├── forgot\_password.html # 忘記密碼

│ ├── reset\_password.html # 重設密碼

│ └── about.html # 關於系統

└── static/

└── style.css

#### **實作方法**

**環境檢查與安裝**

1. **環境準備**

git clone https://github.com/AmanoShizukikun/security\_system.git

cd security\_system

pip install -r requirements.txt

1. **啟動應用程式**

flask --debug run

#### **資料庫設計**

**1. users（使用者表）**

CREATE TABLE users (

user\_id TEXT PRIMARY KEY, -- UUID 主鍵

username TEXT UNIQUE NOT NULL, -- 使用者名稱

email TEXT UNIQUE NOT NULL, -- 電子郵件

password\_hash TEXT NOT NULL, -- bcrypt 加密密碼

role TEXT NOT NULL, -- 使用者角色

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

is\_active BOOLEAN DEFAULT 1 -- 帳號啟用狀態

);

**2. security\_incidents（資安事件表）**

CREATE TABLE security\_incidents (

incident\_id TEXT PRIMARY KEY, -- 事件 UUID

title TEXT NOT NULL, -- 事件標題

description TEXT NOT NULL, -- 詳細描述

reporter\_id TEXT NOT NULL, -- 通報者 ID

severity TEXT NOT NULL, -- 嚴重程度

status TEXT NOT NULL, -- 處理狀態

assigned\_to TEXT, -- 指派人員

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

updated\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

resolved\_at TIMESTAMP -- 解決時間

);

**3. incident\_logs（事件記錄表）**

CREATE TABLE incident\_logs (

log\_id TEXT PRIMARY KEY,

incident\_id TEXT NOT NULL, -- 關聯事件

user\_id TEXT NOT NULL, -- 操作人員

log\_type TEXT NOT NULL, -- 記錄類型

action TEXT NOT NULL, -- 執行動作

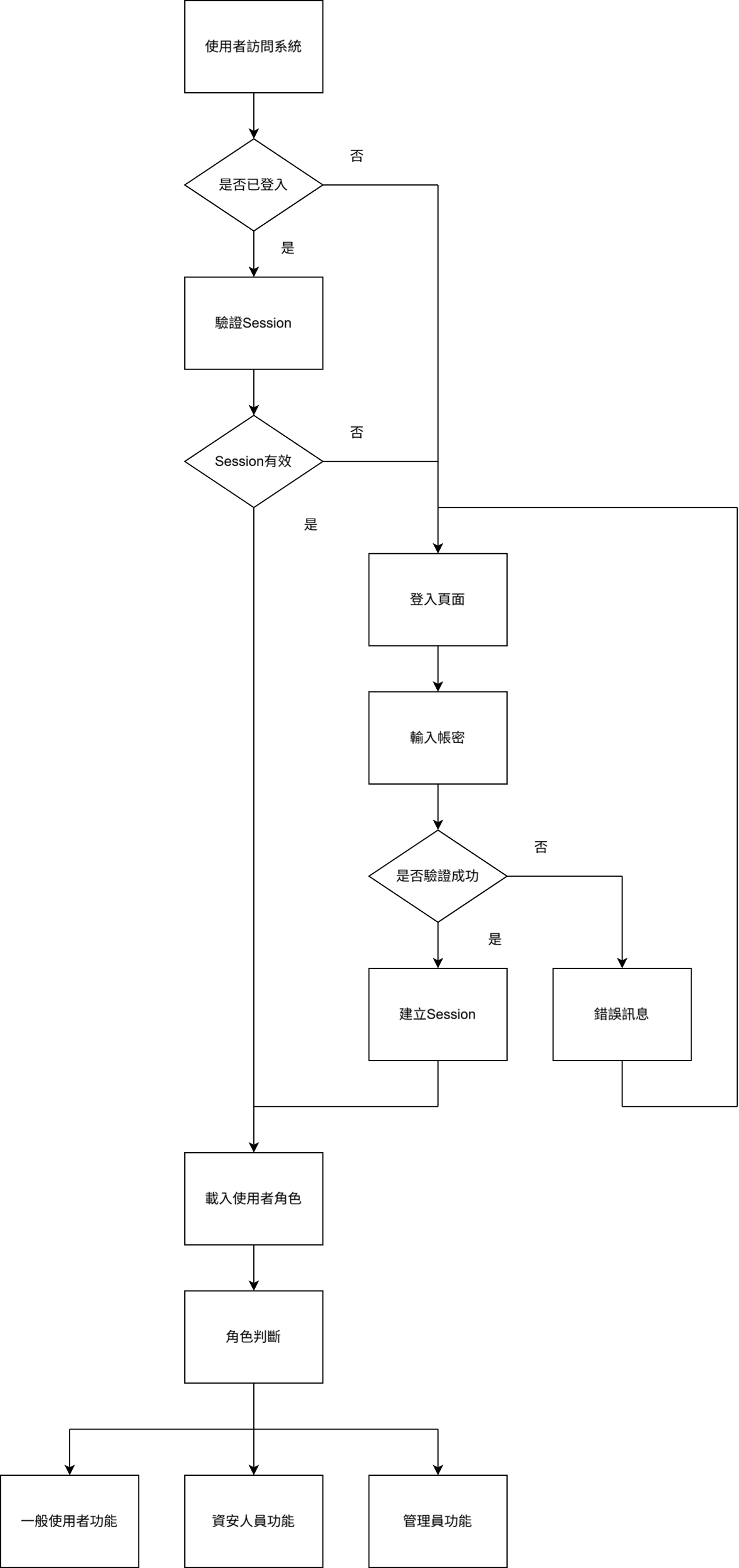
comment TEXT, -- 備註說明

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP

);

#### **系統流程圖**

**主要系統流程**

****

**事件處理詳細流程**

**權限驗證流程**

#### **安裝與設定**

1. **資料庫設計**：
2. **後端設定**：
3. **前端設定**：

#### **運用技巧**

* **權限檢查**：
* **資料隔離**：
* **附件安全**：
* **通知系統**：

### **實作心得**

* **想法**：透過 RBAC 實現權限控制，讓我們更理解資安系統中角色分工的重要性。
* **心得**：
* **困難**：

#### **工作分配**

**李萌家（組長）：**

* 專案企劃與架構設計
* AI 工具基礎框架建構
* 資料庫結構規劃與後期 SQL 語法新功能添加
* 撰寫書面報告
* 安排小組工作

**陳右承：**

* 後端系統核心功能開發
* 前端 UI/UX設計
* 基礎資料庫建置與資料操作
* 基礎框架測試與除錯優化
* 使用者體驗測試與改善
* 技術文件撰寫與專案管理
* 書面報告內容修正
* 期末報告上台負責人
* 團隊協調與反饋需修改的缺失內容

**陳柏勁：**

* 修正前端跟後端缺失內容
* 製作PPT
* 書面報告內容修正

#### **參考文獻**

* Flask 文檔：<https://flask.palletsprojects.com/>
* RBAC 介紹：<https://csrc.nist.gov/projects/role-based-access-control>
* SQLite 文檔 - <https://www.sqlite.org/docs.html>

#### **專案連結**

* **GitHub 儲存庫：**[**https://github.com/AmanoShizukikun/security\_system**](https://github.com/AmanoShizukikun/security_system)
* **專案展示影片：**[**https://youtu.be/FcvB6M-iEws?si=QUaJgNEo6I91XWcz**](https://youtu.be/FcvB6M-iEws?si=QUaJgNEo6I91XWcz)