

# Toastlabplus 專案實作計畫

**專案概述:** Flutter App (宮崎駿風 UI + Chat) + Azure 雲端服務

## 1. 系統架構與資料流

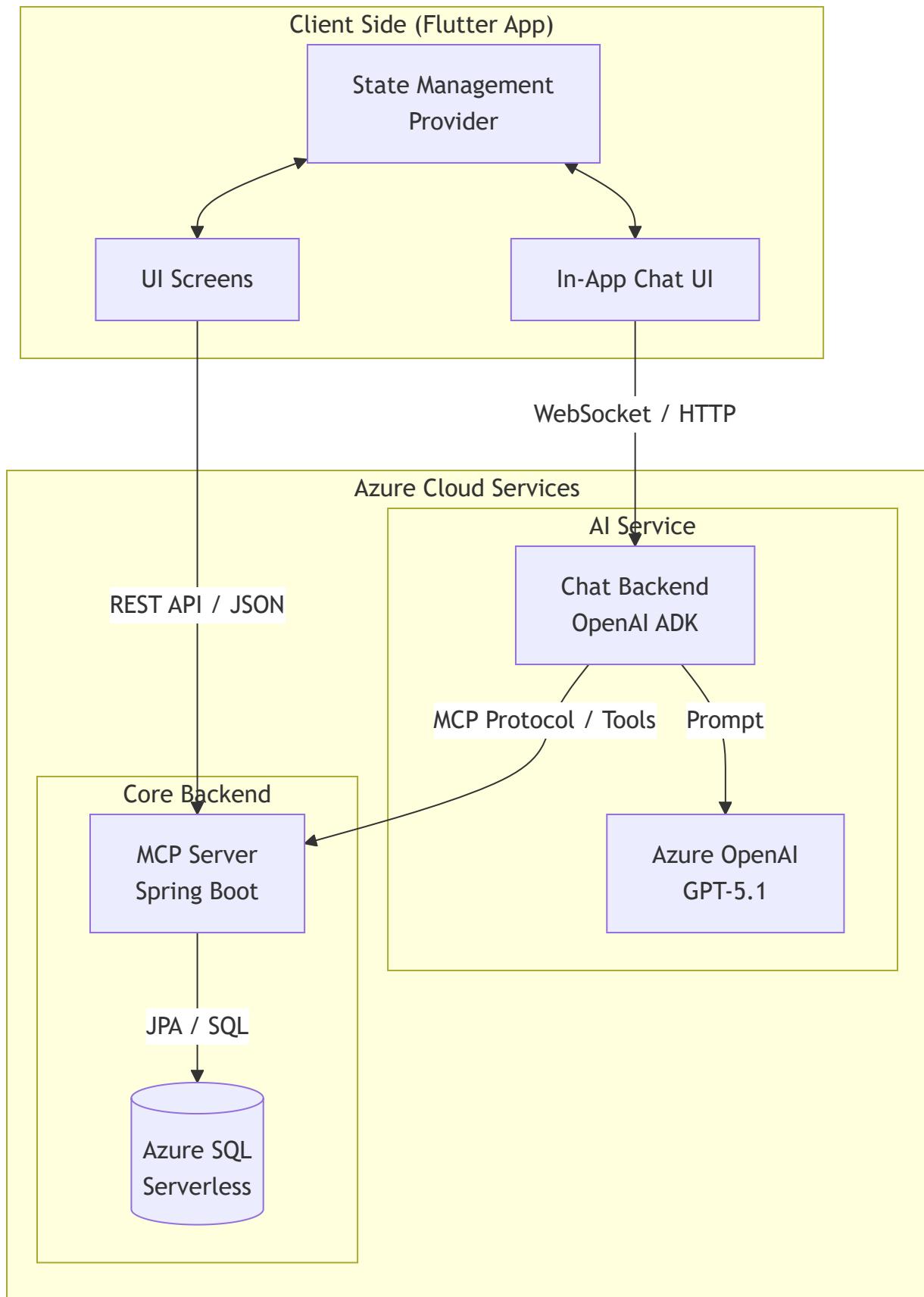
### 1.1 架構概述

本系統採用前後端分離架構，以 Azure 為核心雲端平台。

**核心組件:**

- Client Side:** 使用 Flutter 建構跨平台 App，包含標準 UI 操作介面與 In-App Chat 聊天介面
- AI Service:** 透過 OpenAI ADK 框架建構 Chat Backend，連接 Azure OpenAI GPT-5.1 模型，負責理解用戶自然語言指令並調度 MCP Server
- Core Backend:** Spring Boot MCP Server 作為核心資料服務，處理所有業務邏輯與資料庫存取
- Database:** 使用 Azure SQL Serverless，兼顧效能與成本效益

## 1.2 系統架構圖



## 2. 使用者註冊與身分選擇

### 2.1 註冊流程

系統支援多種身分註冊，確保不同角色的使用者能獲得對應的權限與功能。

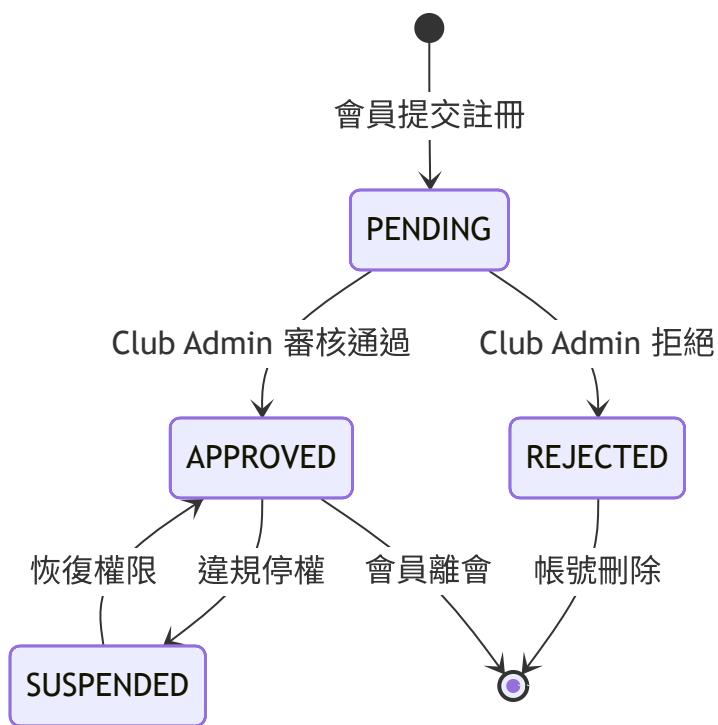
**註冊步驟：**

1. **註冊入口：**使用者下載 App 後，可選擇「註冊新帳號」
2. **身分選擇：**
  - **Guest (來賓)：**適用於非 Toastmasters 會員或參觀者
    - 僅需填寫基本姓名與 Email
    - 註冊後可瀏覽公開會議資訊
    - 無法報名角色
  - **Member (會員)：**適用於正式會員
    - 註冊時需選擇所屬分會 (Club)
    - 提交後狀態為 PENDING
    - 需等待分會管理員審核通過後才能啟用完整功能（如報名角色）
3. **平台管理員：**系統預設一組最高權限帳號，用於建立分會與指派初始管理員

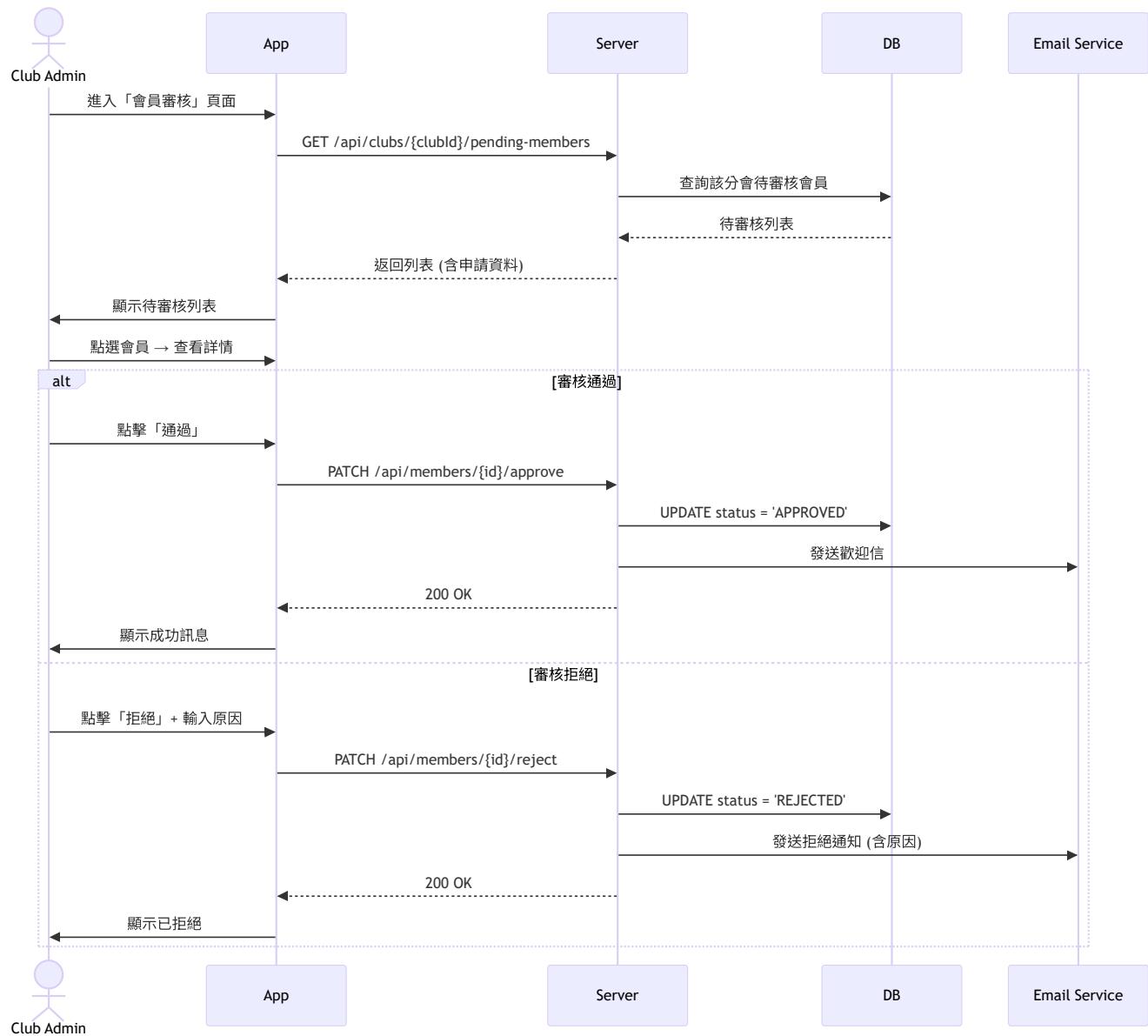
## 2.2 註冊流程圖



## 2.3 會員審核狀態機



## 2.4 審核操作流程



## 2.5 審核通知機制

事件	通知對象	通知方式	內容
新申請提交	Club Admin	App Push + Email	「有新會員申請加入」
審核通過	申請者	Email + App 通知	歡迎信 + 功能導覽連結
審核拒絕	申請者	Email	拒絕原因 + 重新申請引導
審核超時 (7天未處理)	Club Admin	App Push	提醒處理待審核申請

### 3. 角色權限設計

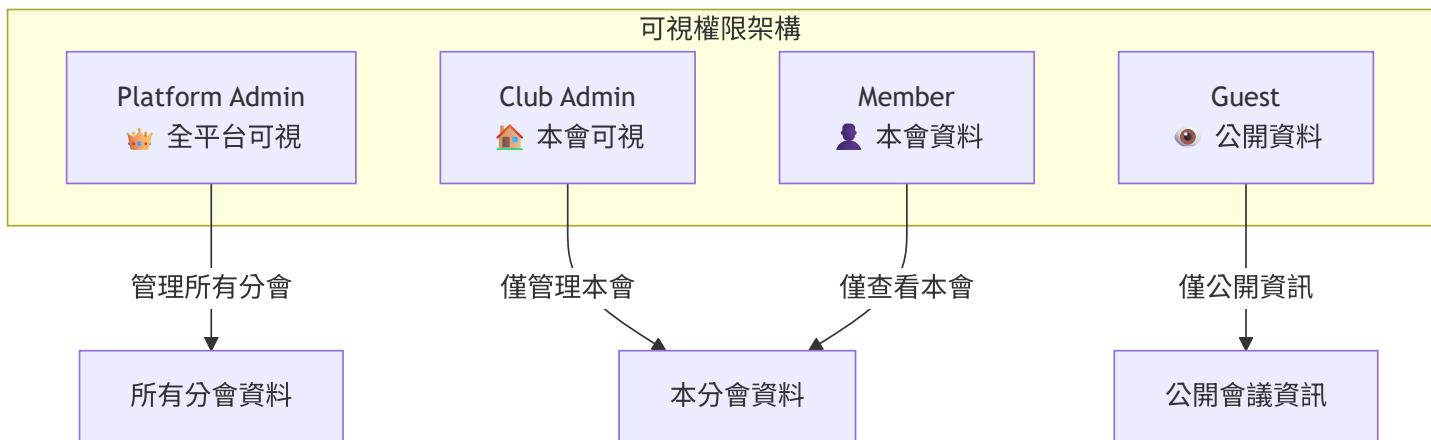
#### 3.1 權限階級

系統定義了四種權限階級，嚴格控管功能存取：

- **Platform Admin** (平台管理員)
- **Club Admin** (分會管理員)
- **Member** (會員)
- **Guest** (來賓)

#### 3.2 資料可視範圍原則

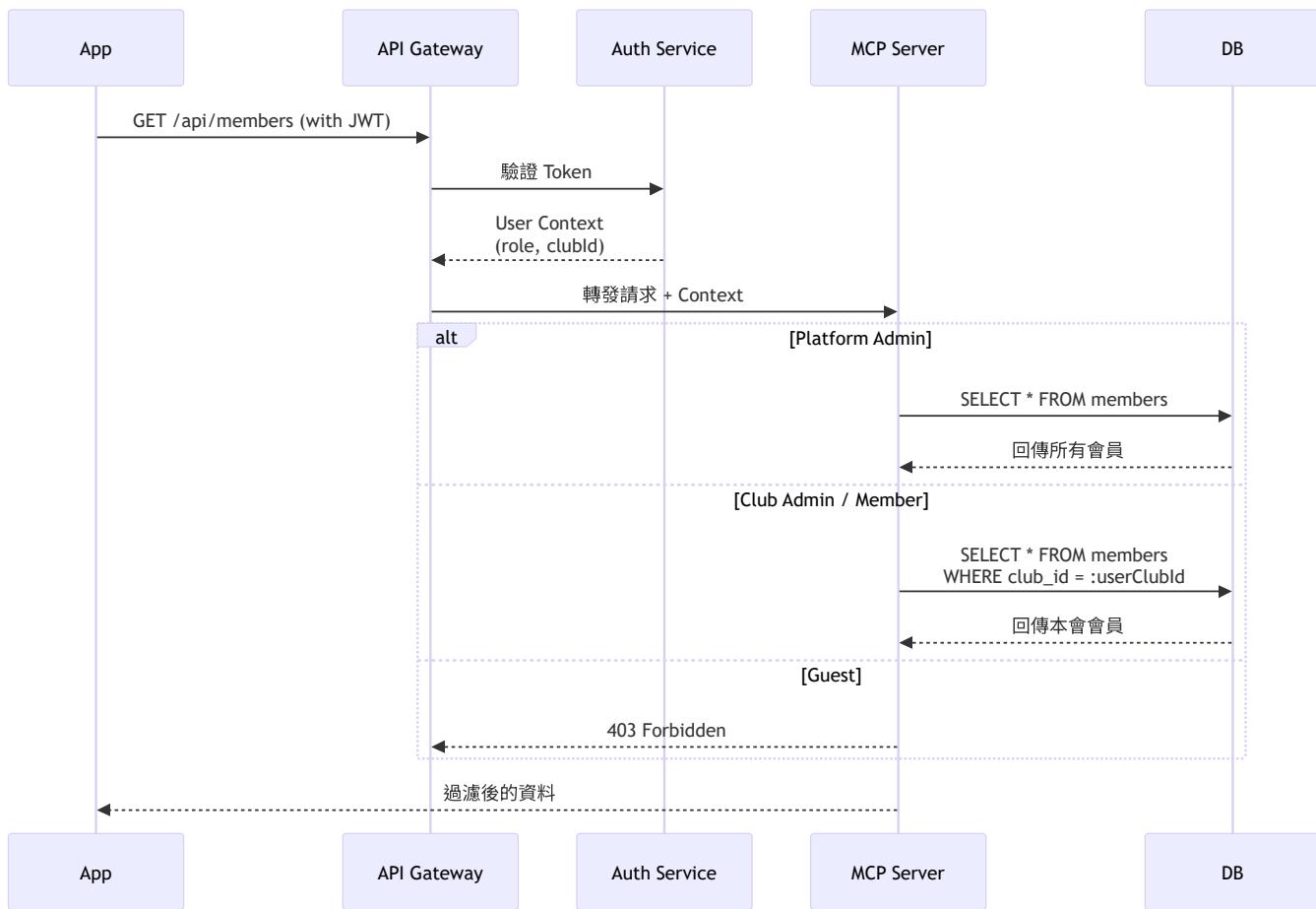
系統採用「分會隔離」原則，確保各分會資料互不可見。



#### 3.3 Club Admin 可視權限詳細矩陣

資料類型	本會資料	他會資料	說明
會員列表	<input checked="" type="checkbox"/> 完整資料	<input type="checkbox"/> 不可見	含姓名、Email、狀態、職位
待審核會員	<input checked="" type="checkbox"/> 完整資料	<input type="checkbox"/> 不可見	可執行審核操作
會議列表	<input checked="" type="checkbox"/> 完整	<input type="radio"/> 僅公開	他會僅見已發布的公開會議
角色報名狀態	<input checked="" type="checkbox"/> 含會員姓名	<input type="radio"/> 僅統計	他會僅見「已報名 X 人」
Agenda 模板	<input checked="" type="checkbox"/> 可編輯	<input type="checkbox"/> 不可見	模板屬於各分會私有
Agenda 文件	<input checked="" type="checkbox"/> 可編輯	<input type="radio"/> 僅已發布	他會僅見最終版 Agenda
分會設定	<input checked="" type="checkbox"/> 可編輯	<input type="radio"/> 僅基本資訊	名稱、聯絡方式等公開資訊

## 3.4 API 資料過濾機制



## 3.5 功能權限矩陣

功能模組	功能項目	Platform Admin	Club Admin	Member	Guest
系統管理	新增/刪除分會	✓	✗	✗	✗
	指定分會管理員	✓	✗	✗	✗
	維護角色定義	✓	✗	✗	✗
分會管理	審核會員註冊	✓	✓	✗	✗
	指派分會職位 (VPE等)	✓	✓	✗	✗
	編輯分會資訊	✓	✓	✗	✗
會議管理	建立/編輯會議	✓	✓	✗	✗
	產生/匯出 Agenda	✓	✓	✗	✗
	管理 Agenda 模板	✓	✓	✗	✗

功能模組	功能項目	Platform Admin	Club Admin	Member	Guest
角色報名	報名會議角色	✓	✓	✓	✗
	取消自己角色	✓	✓	✓	✗
	強制移除他人角色	✓	✓	✗	✗
	代理報名他人	✓	✓	✗	✗
資訊瀏覽	查看會議議程	✓	✓	✓	✓
	查看分會資訊	✓	✓	✓	✓

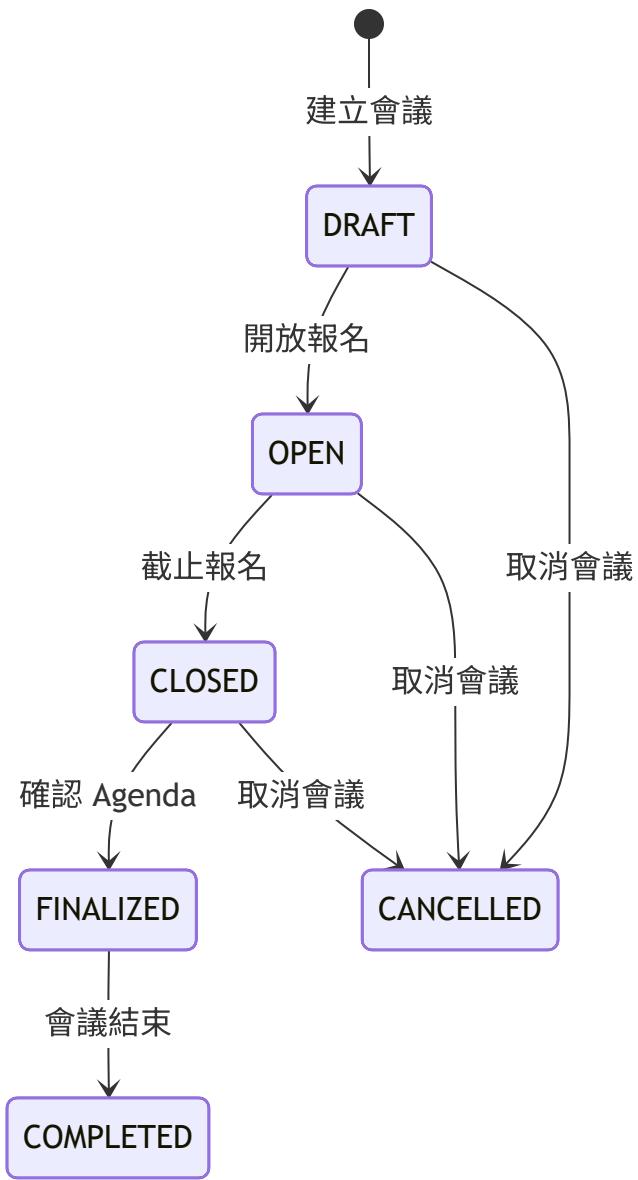
## 3.6 分會職位

分會管理員可將以下職位指派給會員（一職一人），這些職位在 App 中會有特殊標識，且 VPE 擁有產生 Agenda 的權限：

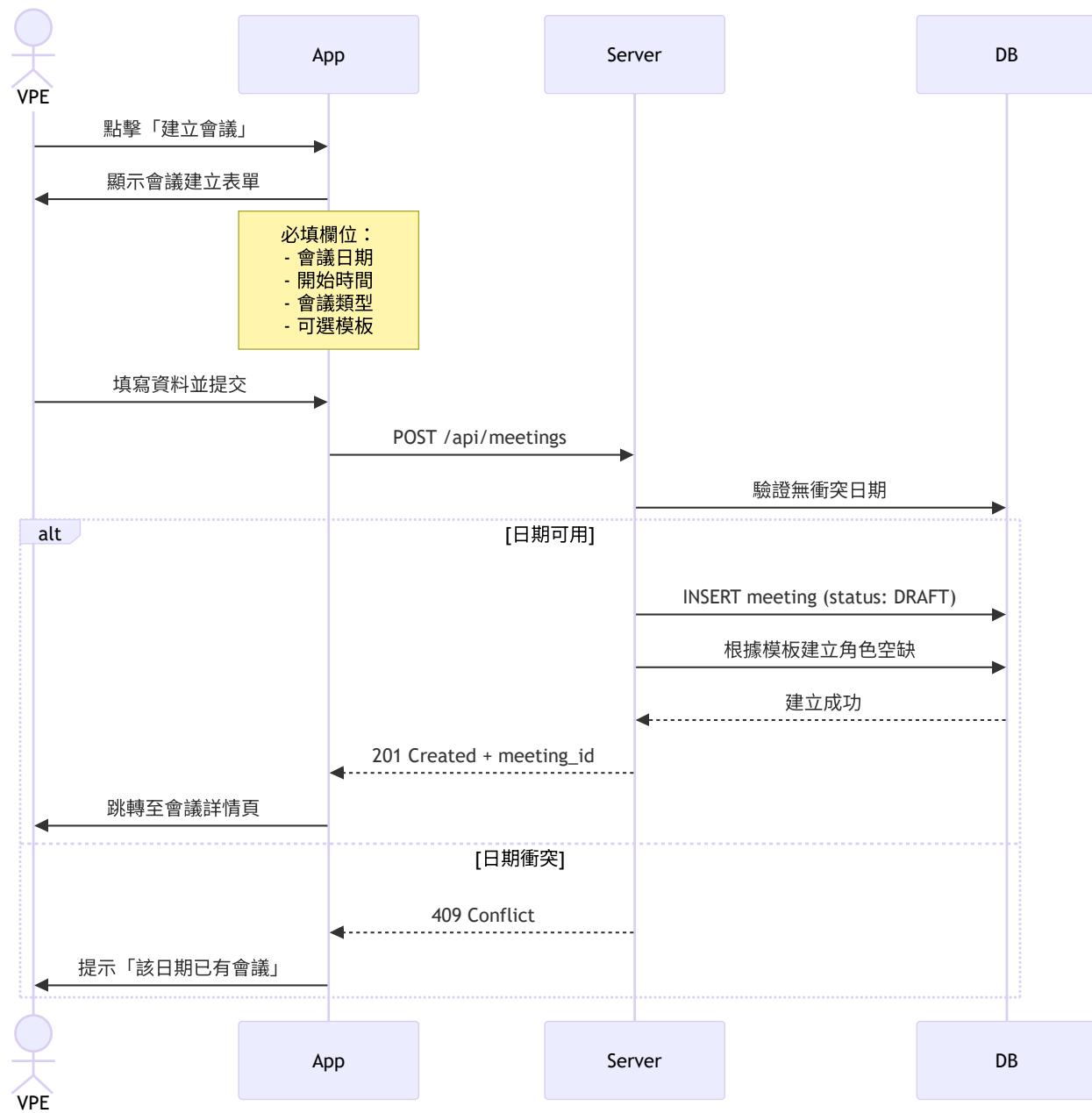
- **President** (會長)
- **VPE** (教育副會長) - 核心操作者
- **VPM** (會員副會長)
- **VPPR** (公關副會長)
- **Secretary** (秘書)
- **Treasurer** (財務長)
- **SAA** (場控)

## 4. 會議管理細部流程

### 4.1 會議生命週期



## 4.2 會議建立流程



## 4.3 會議編輯權限

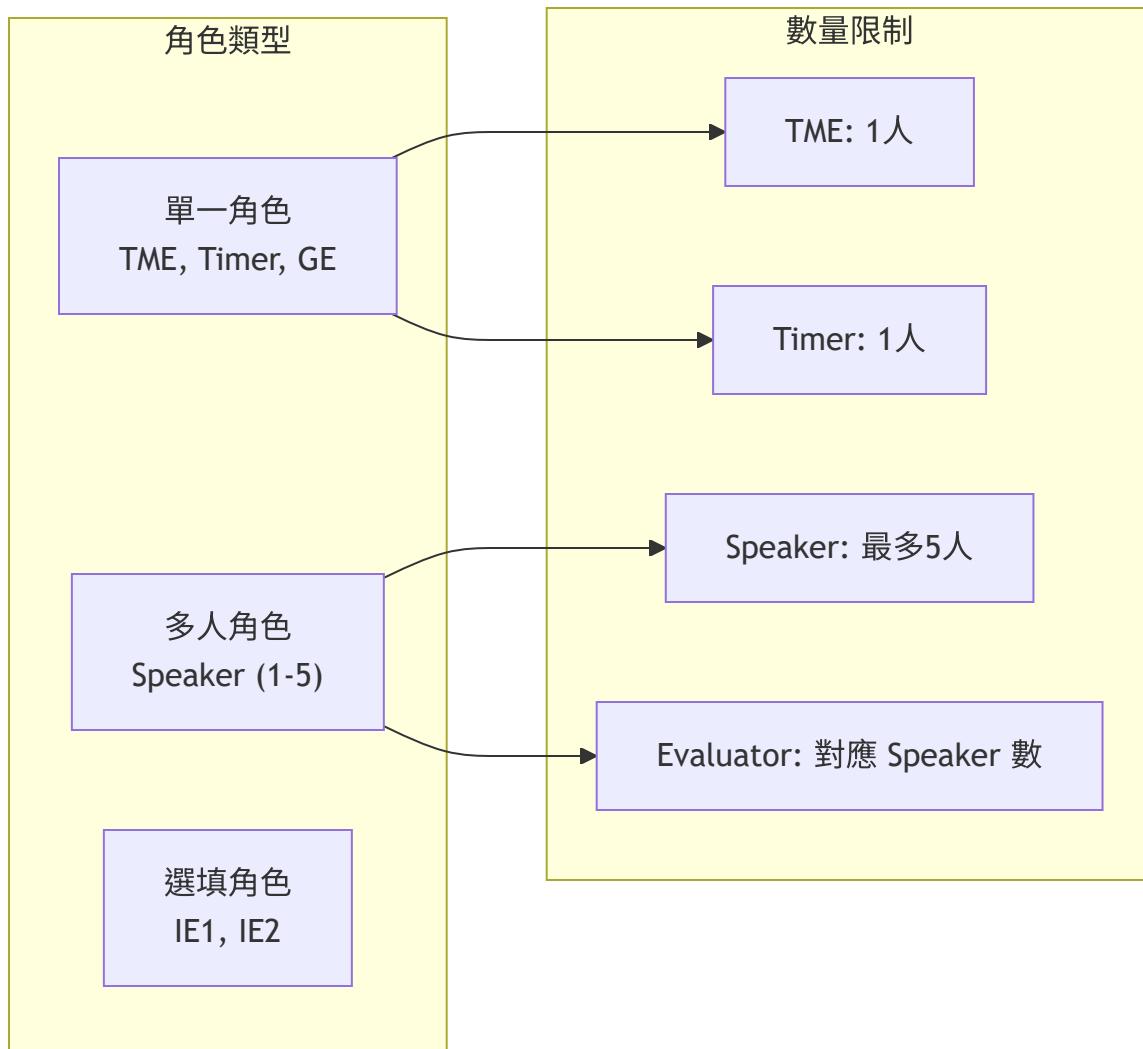
操作	DRAFT	OPEN	CLOSED	FINALIZED
修改日期/時間	✓	⚠ 需通知	✗	✗
修改會議類型	✓	✗	✗	✗
增減角色	✓	✓	⚠ 需確認	✗
開放報名	✓	-	-	-

操作	DRAFT	OPEN	CLOSED	FINALIZED
截止報名	-	✓	-	-
產生 Agenda	-	-	✓	✓ (微調)
取消會議	✓	✓ 需通知	✓ 需確認	✗

## 5. 會議角色註冊流程

提供「Chat 對話」與「UI 介面」兩種操作方式，資料即時同步。

### 5.1 角色類型與限制



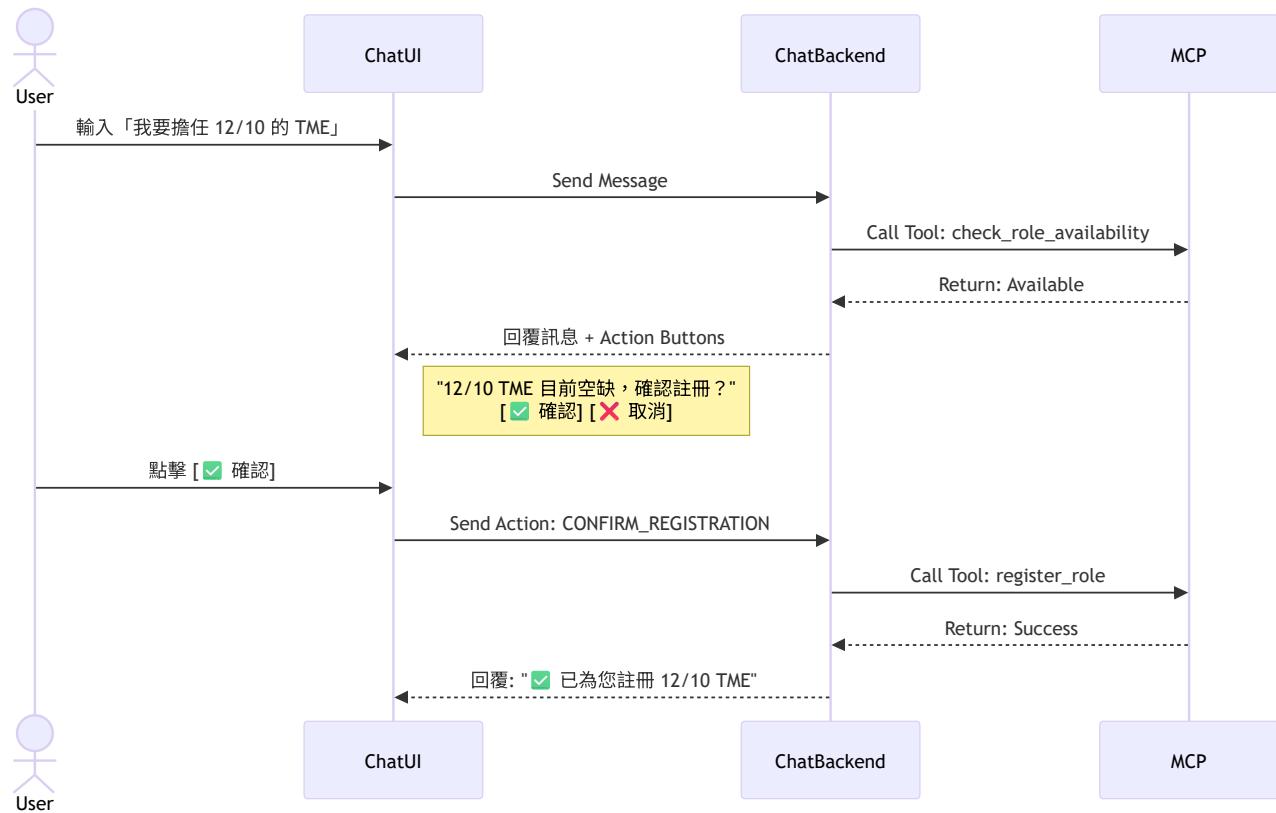
## 5.2 Chat 對話式註冊

透過自然語言與 AI 互動。AI 會先檢查角色空缺狀態，若有空缺則顯示互動式按鈕（Button）供用戶確認，避免誤操作。

### 功能特點：

- **指令範例：**「我要報名下週五的 Timer」、「取消我的 TME 角色」
- **防呆機制：**若角色已滿，AI 會建議其他空缺角色或候補

### 對話流程圖：



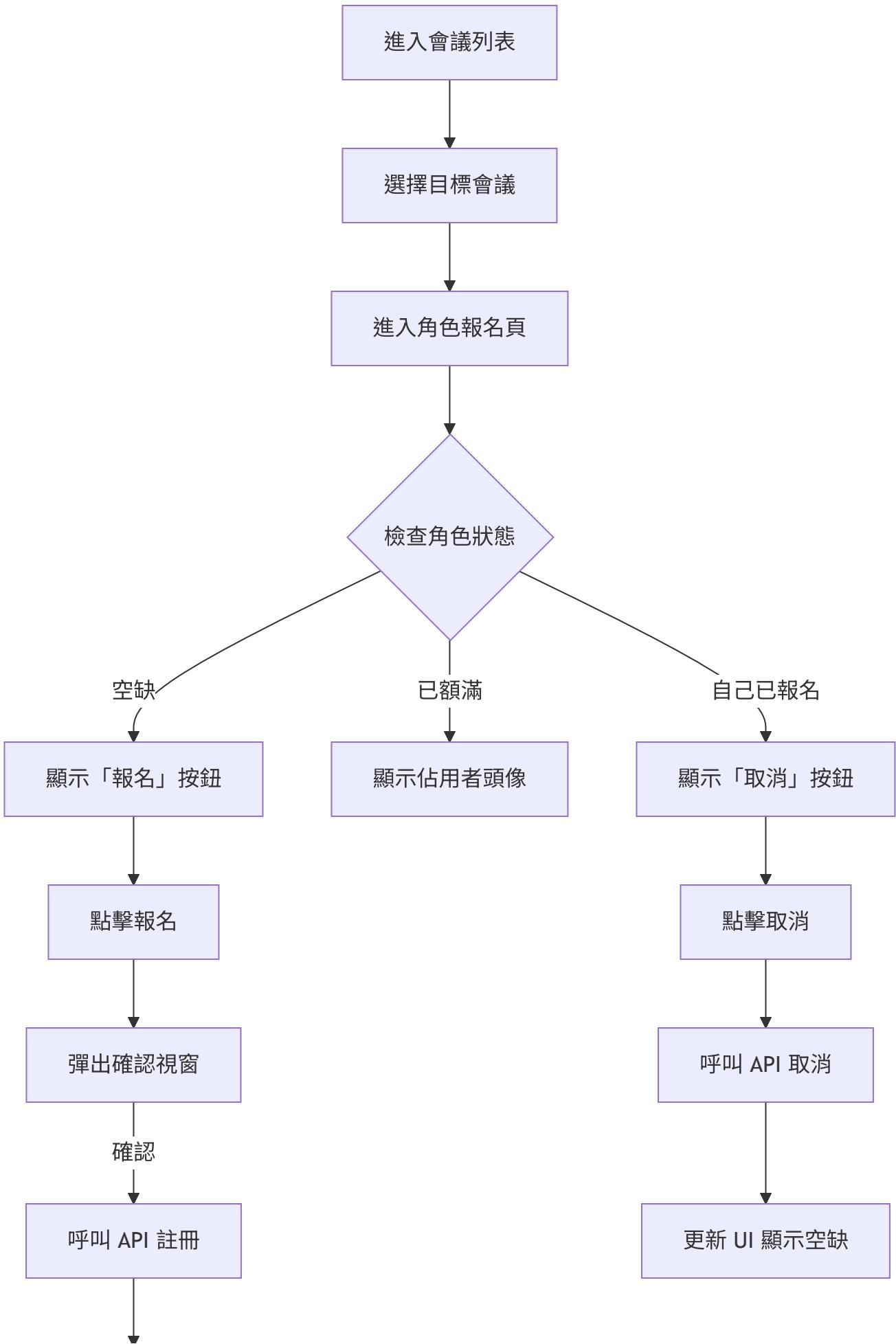
## 5.3 UI 介面註冊

視覺化的角色列表，提供直覺的操作體驗。

### 功能特點：

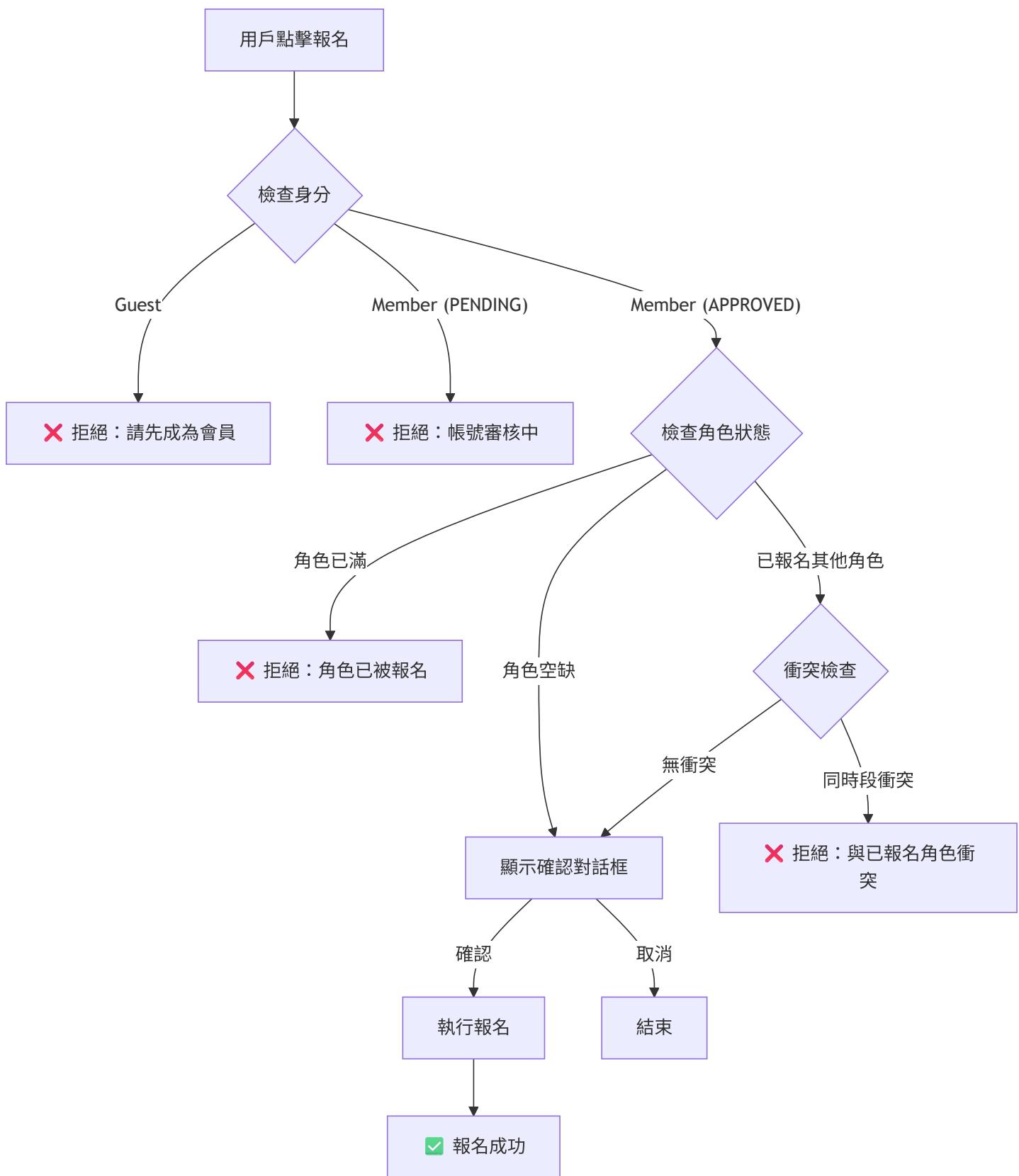
- **狀態顯示：**每個角色卡片會顯示「空缺（可報名）」、「已額滿（顯示頭像）」或「已報名（顯示取消按鈕）」
- **操作流程：**點擊空缺卡片 → 彈出確認窗 → 完成報名

### 操作流程圖：



更新 UI 顯示自己頭像

## 5.4 角色報名防呆流程

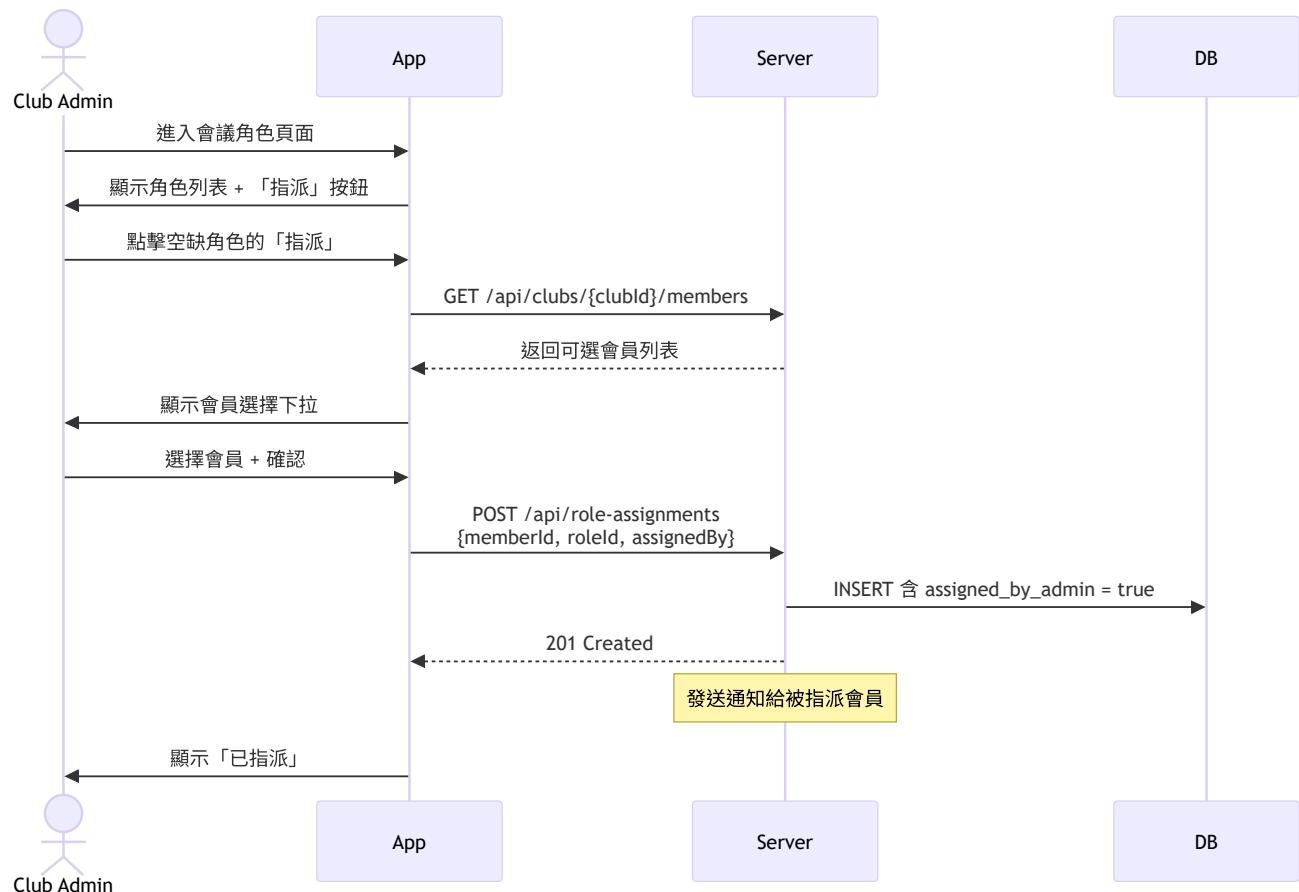


## 5.5 角色衝突規則

角色 A	可兼任	不可兼任
TME	-	Speaker, GE, Timer, Evaluator
Speaker	IE (非同場)	TME, Evaluator (同一人)
Timer	AH Counter, Grammarian	TME
GE	-	TME, Speaker, Evaluator
Evaluator	Timer, AH Counter	TME, GE, 對應 Speaker

## 5.6 管理員代理報名

Club Admin 可代替會員報名或取消報名：



# 6. Agenda 模板管理與產生

此功能專為 VPE (教育副會長) 設計，用於快速產生標準化的會議議程。

## 6.1 模板管理

功能流程：

1. **上傳**: 支援上傳 Excel 格式的議程範本
2. **解析與編輯**: 後端解析 Excel 後，VPE 可在 App 介面上調整時段順序、時間長度與負責職位
3. **儲存**: 將調整好的結構儲存為「標準例會」、「比賽」、「特別活動」等不同模板

## 6.2 模板結構設計

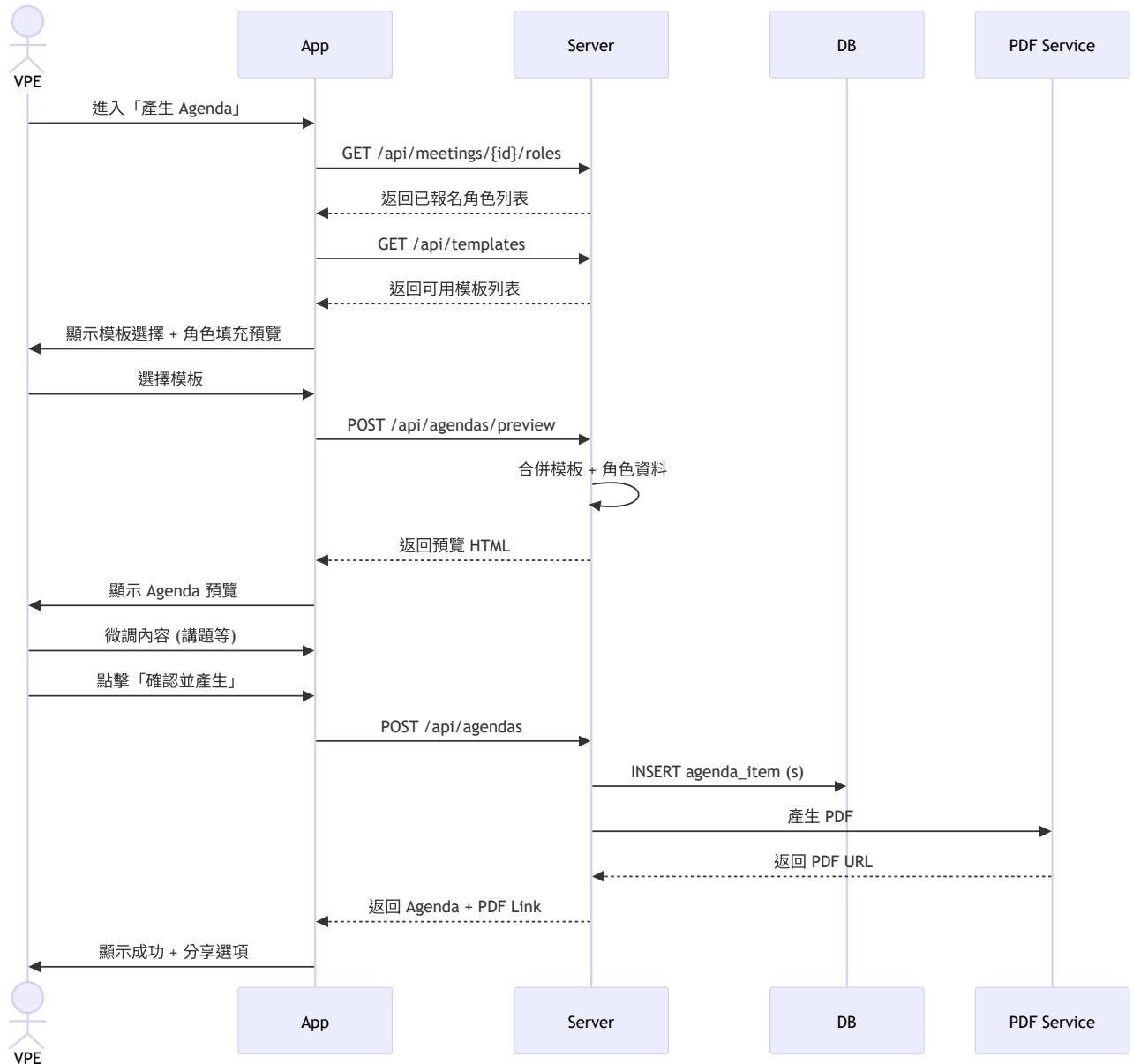
```
{
  "templateId": "standard-meeting-v1",
  "name": "標準例會",
  "sections": [
    {
      "order": 1,
      "name": "開場",
      "duration": 10,
      "items": [
        { "name": "Sergeant at Arms", "role": "SAA", "duration": 3 },
        { "name": "Opening", "role": "President", "duration": 2 },
        { "name": "TME Welcome", "role": "TME", "duration": 5 }
      ]
    },
    {
      "order": 2,
      "name": "準備演講",
      "duration": 35,
      "items": [
        { "name": "Speaker 1", "role": "Speaker", "duration": 7 },
        { "name": "Speaker 2", "role": "Speaker", "duration": 7 },
        { "name": "Speaker 3", "role": "Speaker", "duration": 7 }
      ]
    }
  ]
}
```

## 6.3 議程產生

功能流程：

1. **選擇**: 選擇會議日期與要套用的模板
2. **自動合併**: 系統自動將該次會議「已報名的角色」(如 TME, Speaker 1, Timer) 填入模板對應的欄位
3. **微調與發布**: VPE 可手動修改講題、調整臨時變動，確認無誤後匯出 PDF 或產生分享連結

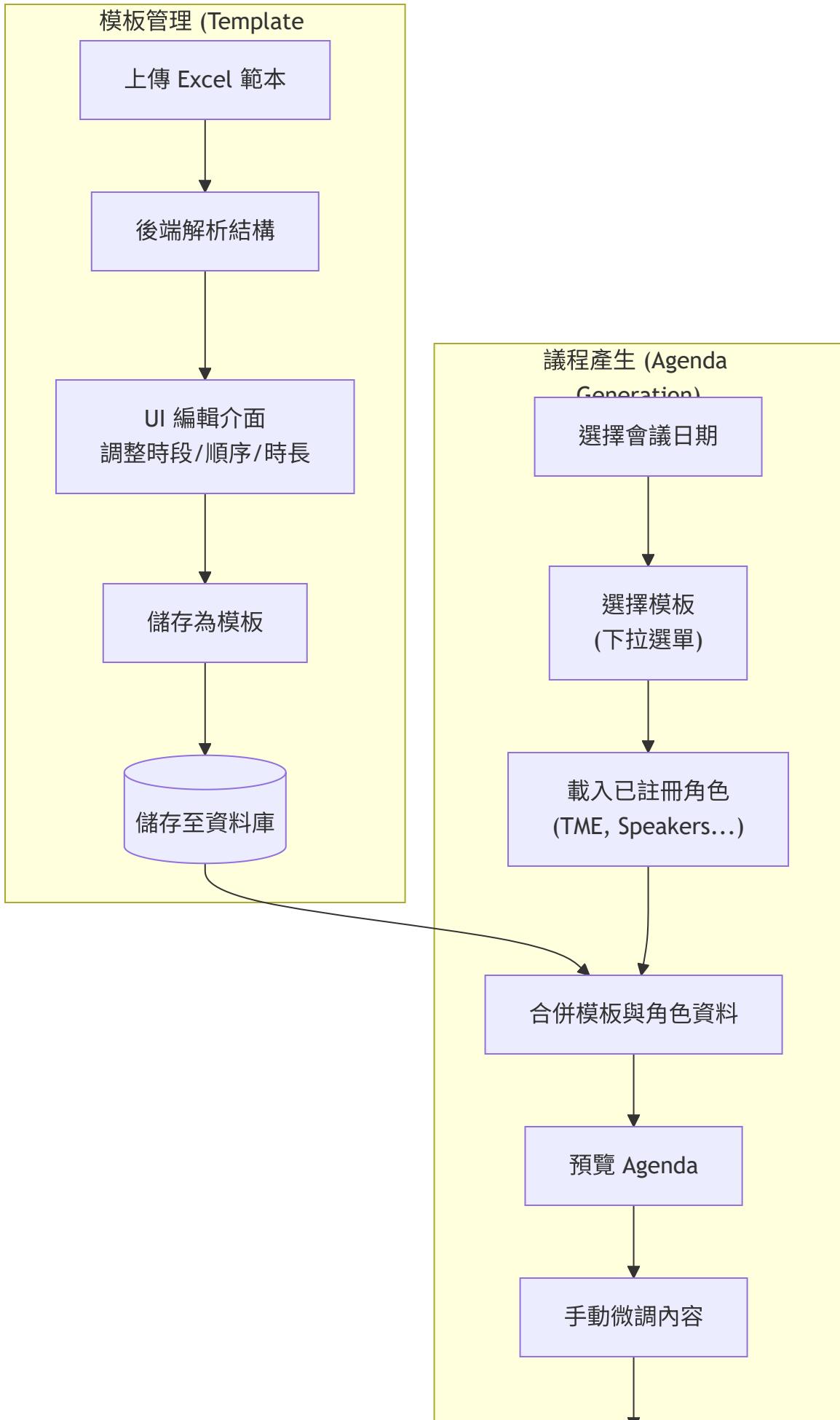
## 6.4 議程產生流程

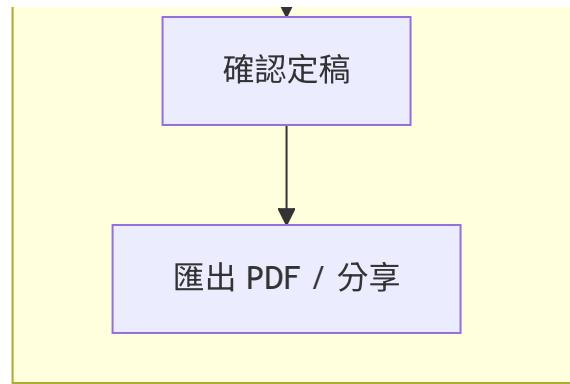


## 6.5 Agenda 狀態與權限

狀態	VPE 操作	Club Admin 操作	會員可見
DRAFT	編輯、刪除、預覽	查看、編輯	
PUBLISHED	小幅修正、發布更新	查看、修正	查看
ARCHIVED	查看	查看	查看

## 6.6 模板與產生流程圖

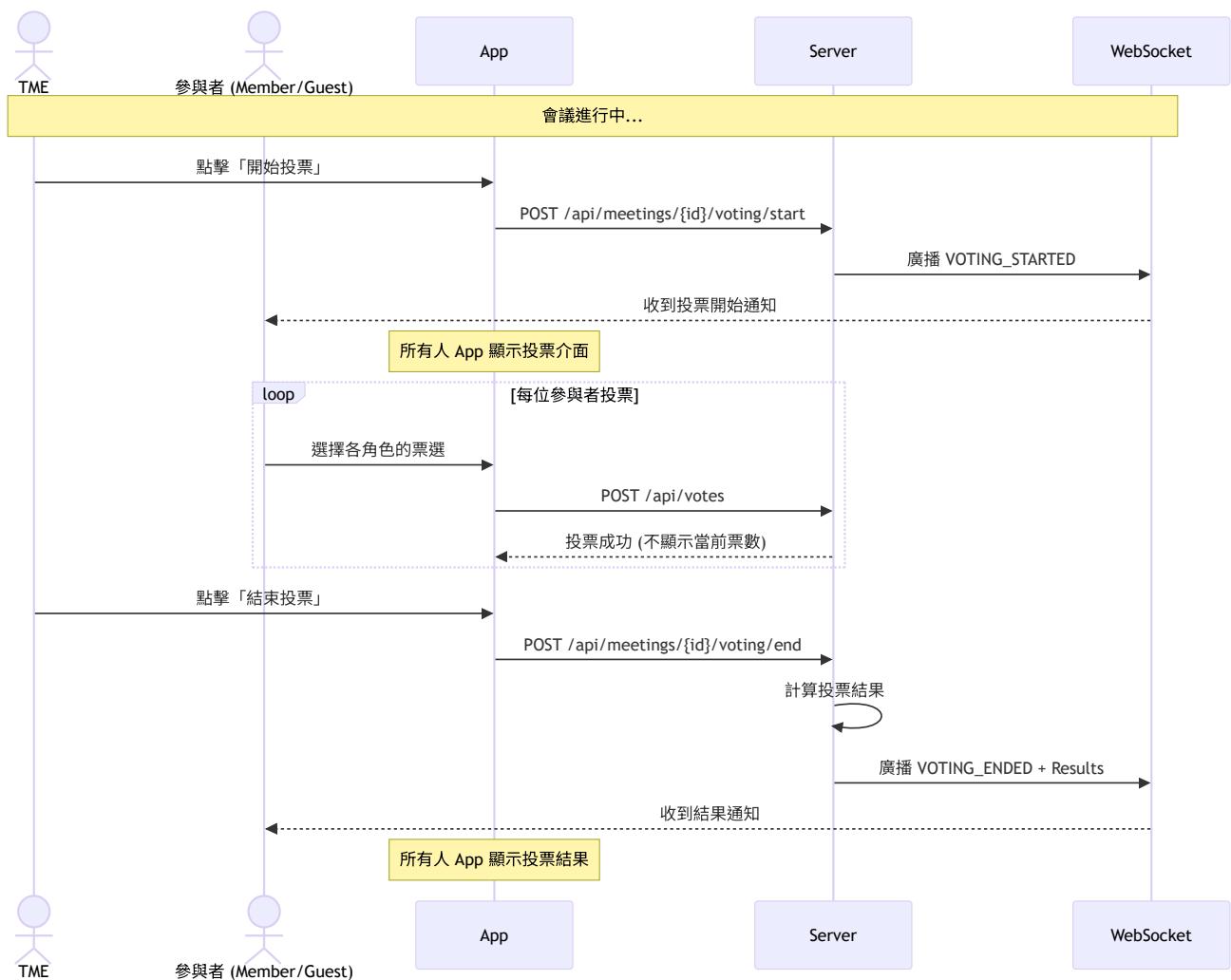




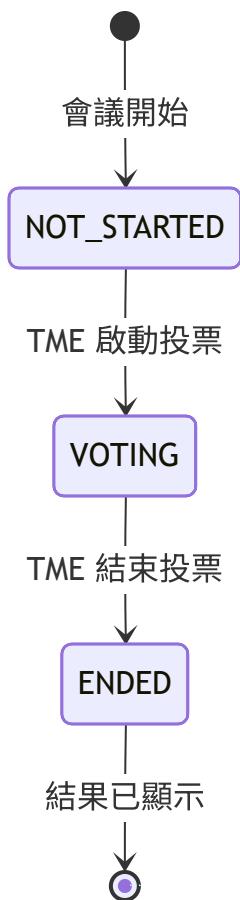
## 7. 會議投票機制 (Voting System)

會議進行中的即時投票功能，由 TME 控制投票流程，所有會議參與者皆可投票。

### 7.1 投票流程概述



## 7.2 投票狀態機



## 7.3 投票類別與獎項

投票類別	對象	說明
<b>Best Speaker</b>	所有 Speaker	最佳演講者
<b>Best Evaluator</b>	所有 Evaluator	最佳講評者
<b>Best Table Topic</b>	Table Topic 回答者	最佳即席演講
<b>Best Support Role</b>	Timer, AH Counter, Grammarian	最佳輔助角色

## 7.4 投票權限

角色	操作	說明
<b>TME</b>	<input checked="" type="checkbox"/> 啟動/結束投票	唯一控制者
<b>Member (APPROVED)</b>	<input checked="" type="checkbox"/> 投票	需為會議參與者

角色	操作	說明
Guest	<input checked="" type="checkbox"/> 投票	需為會議參與者
Role Taker	<input type="checkbox"/> 自己類別	不能投票給自己

## 7.5 投票界面流程

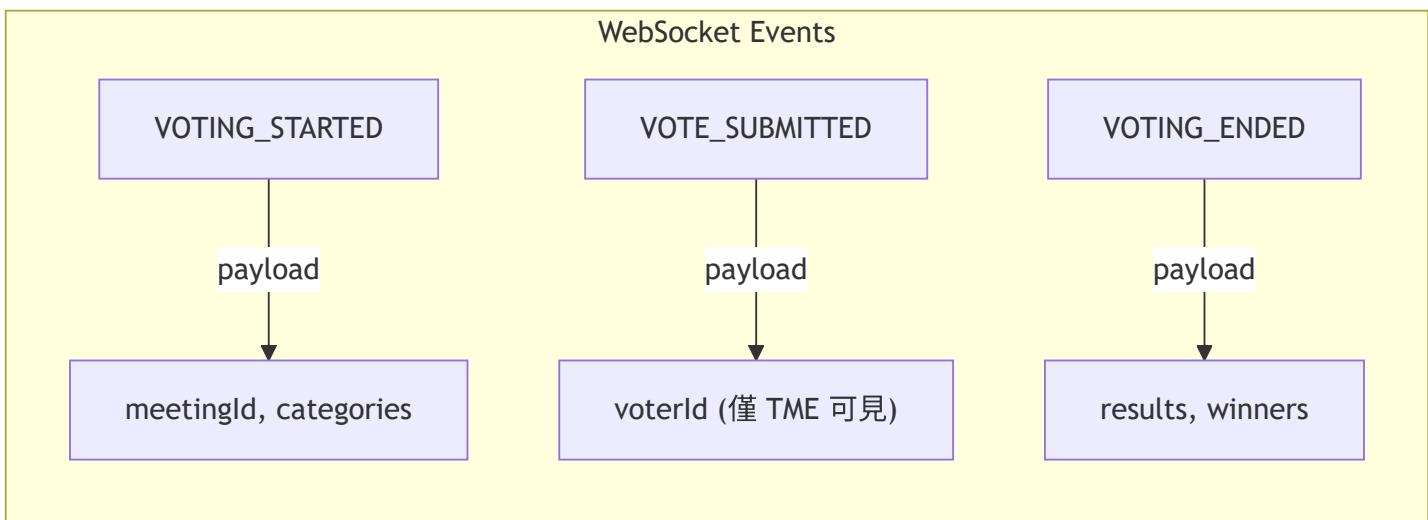


## 7.6 投票 API 設計

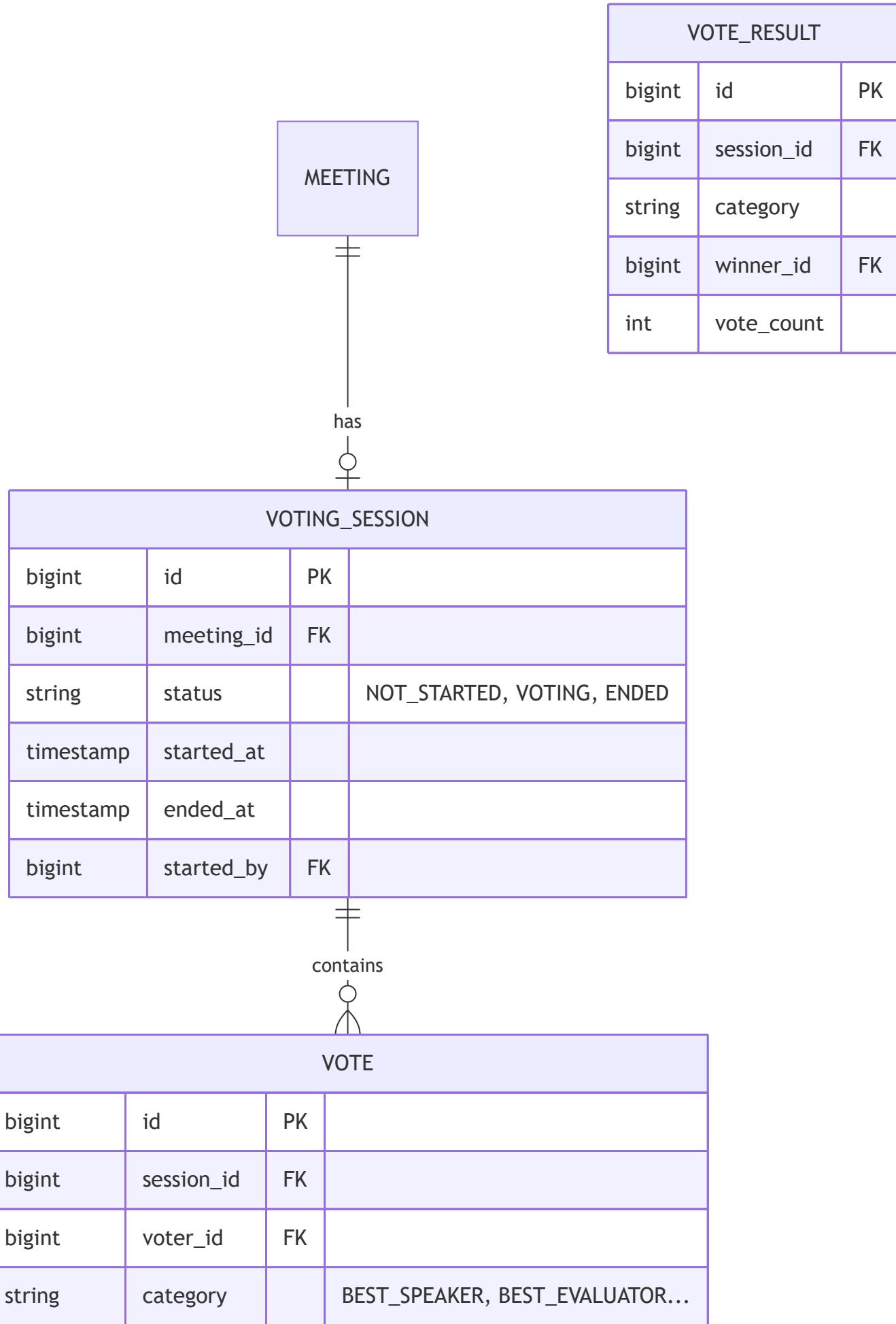
Method	Endpoint	說明	權限
POST	/api/meetings/{id}/voting/start	啟動投票	TME only
POST	/api/meetings/{id}/voting/end	結束投票	TME only
GET	/api/meetings/{id}/voting/status	查詢投票狀態	參與者
POST	/api/votes	提交投票	參與者
GET	/api/meetings/{id}/voting/results	查詢結果	投票結束後

## 7.7 即時通訊設計

使用 WebSocket 實現即時同步：



## 7.8 資料庫設計



bigint	voted_for	FK	
timestamp	created_at		

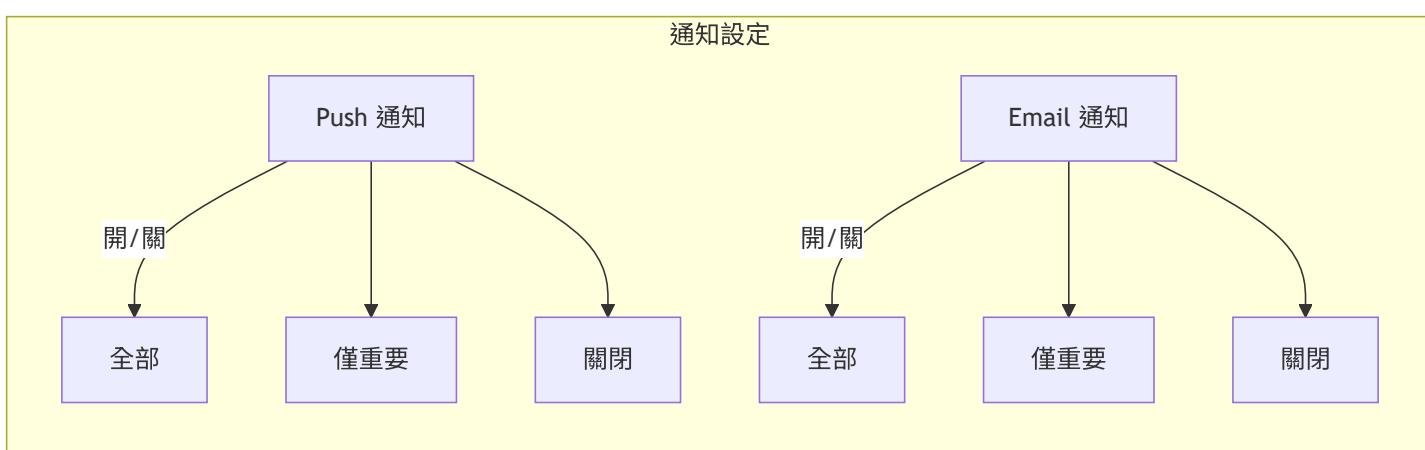
## 8. 通知系統設計

### 8.1 通知類型與觸發條件

類別	事件	通知對象	通知管道
會員	審核通過/拒絕	申請者	Push + Email
會議	新會議建立	全體會員	Push
	會議取消	已報名者	Push + Email
	報名截止提醒	未報名會員	Push
角色	被指派角色	被指派者	Push
	角色被移除	原報名者	Push + Email
Agenda	Agenda 發布	全體會員	Push
	Agenda 更新	全體會員	Push

### 8.2 通知偏好設定

用戶可在設定中調整通知偏好：

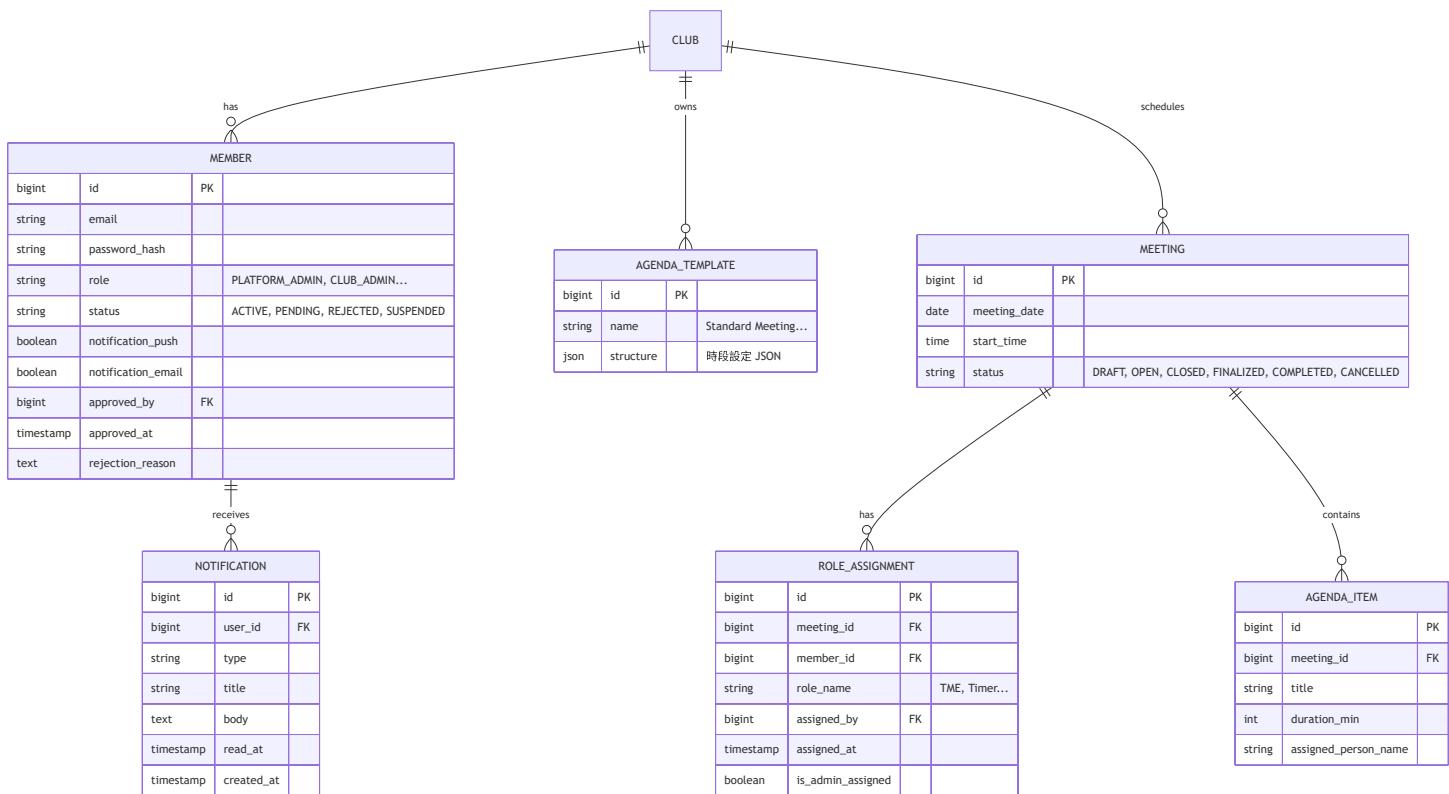


# 8. 資料庫設計

## 8.1 主要實體

- **CLUB**: 分會基本資料
- **MEMBER**: 會員資料，包含權限角色（Role）與狀態
- **MEETING**: 會議主檔，包含日期、主題
- **ROLE\_ASSIGNMENT**: 記錄誰在該次會議擔任什麼角色
- **AGENDA\_TEMPLATE**: 儲存議程結構的 JSON 定義
- **AGENDA\_ITEM**: 議程項目明細
- **NOTIFICATION**: 通知記錄

## 8.2 實體關係圖



# 9. 技術棧與部署

## 9.1 技術選型

組件	技術	Azure 服務	規格
Mobile App	Flutter 3.x, Provider, Dio	-	iOS / Android
MCP Server	Spring Boot 3.x, JPA	Container Apps	0.5 vCPU, 1GB
Chat Backend	OpenAI ADK (Python)	Container Apps	0.5 vCPU, 1GB
Database	Azure SQL	SQL Database	Serverless Tier
AI Model	GPT-5.1	AI Foundry	Standard

# 10. 功能雛型畫面

以下為 App 的核心功能畫面設計，採用宮崎駿風格的簡約可愛設計語言。

## 10.1 首頁

包含主要功能入口（Make Agenda）與快速導航。

# ToastLab<sup>+</sup>



Make Agenda



Club Info



Roles



Voting



Home



Chat



Settings

## 10.2 角色報名

視覺化的角色列表，清晰顯示空缺與已佔用狀態。

# Meeting Roles 12/10



## Timer

Available

Sign Up



## Speaker 1

Taken

Alex



## Ah Counter

Available

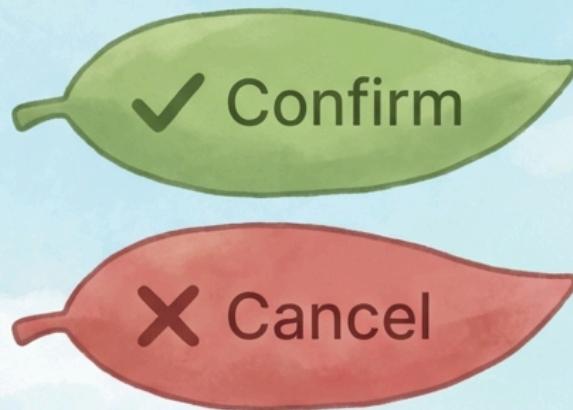
Sign Up

## 10.3 In-App Chat

對話式操作介面，支援互動式按鈕以簡化確認流程。



12/10 TME is available.  
Do you want to sign up?



## 10.4 議程產生

VPE 專用介面，支援模板選擇與預覽匯出。



## 附錄

## 版本歷史

- **v1.0** (2025-12-06): 初版完成，包含完整系統架構與功能設計
- **v1.1** (2025-12-06): 新增細部流程設計
  - 新增 Club Admin 可視權限詳細矩陣

- 新增 API 資料過濾機制
  - 新增會員審核狀態機與操作流程
  - 新增審核通知機制
  - 新增會議生命週期與編輯權限
  - 新增角色類型與衝突規則
  - 新增管理員代理報名流程
  - 新增模板結構設計 (JSON 格式)
  - 新增 Agenda 狀態與權限
  - 新增通知系統設計
  - 更新資料庫設計 (補充欄位)
- **v1.2 (2025-12-06): 新增會議投票機制**
    - 新增第 7 章 Voting System
    - 新增 TME 控制的投票流程
    - 新增投票狀態機與權限設計
    - 新增即時 WebSocket 通訊設計
    - 新增投票相關資料表 (VOTING\_SESSION, VOTE, VOTE\_RESULT)